

Министерство образования и науки Российской Федерации  
Федеральное агентство по образованию  
Южно-Российский государственный технический университет  
(Новочеркасский политехнический институт)

---

Шахтинский институт (филиал)



**Ю.Н. Попков, А.Ю. Прокопов, М.В. Проколова**

# **ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ГОРНОМ ДЕЛЕ**

*Допущено Учебно-методическим объединением вузов Российской Федерации  
по образованию в области горного дела в качестве учебного пособия для студентов  
вузов, обучающихся по специальности «Шахтное и подземное строительство»  
направления подготовки «Горное дело»*

Новочеркасск 2007

УДК 681.3.06  
ББК 38.11+33.01  
П 57

Рецензенты: д-р техн. наук, проф., директор НГЦ «Наука и практика»  
Ф.И. Ягодкин  
канд. техн. наук, доц. каф. «Информационные системы и  
радиотехника» ЮРГУЭС А.Н. Береза

**Попков Ю.Н., Прокопов А.Ю., Прокопова М.В.**

**П57 Информационные технологии в горном деле: Учеб. пособие/**  
Ю.Н. Попков, А.Ю. Прокопов, М.В. Прокопова/ Шахтинский ин-т  
(филиал) – Новочеркасск: ЮРГТУ, 2007. – 202 с.  
ISBN978-5-88998-739-0

Рассмотрены современные информационные технологии применительно к организационно-экономическим, инженерным и научно-техническим задачам горного производства. Даны примеры использования информационных технологий для составления баз данных, разработки компьютерных моделей, написания прикладных программ, создания объектов средствами компьютерной графики и др. Описаны основные возможности и правила работы в Internet.

Предназначено для студентов специальности 130604 – «Шахтное и подземное строительство» всех форм обучения.

ISBN978-5-88998-739-0

© Шахтинский институт ЮРГТУ, 2007  
© Попков Ю.Н., Прокопов А.Ю.,  
Прокопова М.В., 2007

# ОГЛАВЛЕНИЕ

<b>ВВЕДЕНИЕ .....</b>	<b>6</b>
<b>1. ОБЩИЕ СВЕДЕНИЯ ОБ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЯХ .....</b>	<b>7</b>
1.1. Технологические основы информатики .....	7
1.2. Информационные технологии в горном деле .....	10
<b>2. АДМИНИСТРИРОВАНИЕ СРЕДСТВ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ И СЕТЕЙ.....</b>	<b>14</b>
2.1. Аппаратное обеспечение.....	14
2.2. Операционная система.....	15
2.3. Автоматизированные и автоматические системы управления.....	17
2.4. Компьютерные сети.....	17
2.4.1. Основные понятия.....	17
2.4.2. Основные протоколы, применяемые в компьютерных сетях.....	19
2.4.3. Физическая установка сети.....	22
2.4.4. Рекомендации по конфигурированию сервера .....	23
2.4.5. Совместное использование Интернет .....	30
2.4.6. Совместное использование принтера.....	33
2.4.7. Совместное использование и доступ к файлам.....	33
2.4.8. Добавление сетевых дисков.....	34
2.5. Безопасность информационных систем .....	34
<b>3. ТЕКСТОВАЯ ИНФОРМАЦИЯ, ВЫЧИСЛЕНИЯ И ДЕЛОВАЯ ГРАФИКА .....</b>	<b>37</b>
3.1. Программное обеспечение для обработки информации.....	37
3.1.1. <i>Офисный программный пакет Microsoft Office 2000.....</i>	<i>37</i>
3.1.2. <i>Семейство приложений обработки информации Microsoft Office System .....</i>	<i>38</i>
3.1.3. <i>Перспективы развития Microsoft Office .....</i>	<i>40</i>
3.2. Обработка текстовой информации.....	42
3.2.1. <i>Основные функции текстовых редакторов.....</i>	<i>42</i>
3.2.2. <i>Различные форматы текстовых файлов.....</i>	<i>45</i>
3.2.3. <i>Гипертекст.....</i>	<i>45</i>
3.2.4. <i>Распознавание документов .....</i>	<i>46</i>
3.3. Вычисления и деловая графика .....	46
3.3.1. <i>Электронные таблицы .....</i>	<i>46</i>
3.3.2. <i>Построение диаграмм и графиков.....</i>	<i>50</i>
3.3.3. <i>Вероятность и статистика .....</i>	<i>52</i>
3.3.4. <i>Настройки в электронных таблицах.....</i>	<i>53</i>

<b>4. БАЗЫ ДАННЫХ.....</b>	<b>55</b>
4.1. Данные и персональный компьютер.....	55
4.2. Файлы и файловые системы.....	57
4.3. Концепция баз данных.....	59
4.4. Классификация баз данных.....	63
4.5. Структурные элементы и модели баз данных.....	63
4.5.1. Иерархическая модель данных.....	64
4.5.2. Сетевая модель данных.....	65
4.5.3. Реляционная модель данных.....	65
4.6. Обзор СУБД.....	67
4.7. Пример создания таблицы в MS Access.....	69
4.8. Пример создания формы для ввода данных в MS Access.....	70
4.9. Введение в язык SQL.....	72
4.10. Пример создания запроса в MS Access.....	75
4.11. Разработка баз данных.....	76
4.12. Перспективы развития баз данных.....	78
<b>5. ИСПОЛЬЗОВАНИЕ КОМПЬЮТЕРНОЙ ГРАФИКИ .....</b>	<b>81</b>
5.1. Задачи компьютерной графики.....	81
5.2. Приложения компьютерной графики.....	82
5.3. Технические средства компьютерной графики.....	82
5.4. Графические файлы и их форматы.....	84
5.5. Система автоматизированного проектирования AutoCAD.....	87
5.5.1. Модули AutoCAD.....	87
5.5.2. Общие принципы работы в AutoCAD.....	91
5.5.3. Создание изображений.....	94
5.5.4. Оформление чертежей.....	101
5.5.5. Служебные средства.....	105
5.5.6. Редактирование чертежей.....	107
5.5.7. Выход из графического редактора.....	109
5.6. Понятие геоинформационных систем (ГИС).....	110
<b>6. ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ ....</b>	<b>113</b>
6.1. Алгоритм и его свойства.....	113
6.2. Развитие языков программирования.....	113
6.3. Основы объектно-ориентированного программирования.....	114
6.3.1. Объекты: свойства, методы, события.....	114
6.3.2. Графический интерфейс и событийные процедуры.....	116
6.4. Интегрированная среда программирования Visual Basic.....	118
6.5. Пример разработки приложения в среде Visual Basic.....	122

<b>7. МАТЕРИАЛЬНОЕ И КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ..</b>	<b>157</b>
7.1. Понятие модели и моделирования. Сущность моделирования ..	157
7.2. Классификация моделей .....	158
7.3. Принципы и схема процесса моделирования.....	159
7.4. Методы материального моделирования в горном деле .....	159
7.5. Понятия компьютерного и имитационного моделирования .....	165
7.6. Компьютерное моделирование в горном деле .....	166
7.7. Назначение и структура программного комплекса «Ли́ра-Windows».....	166
7.8. Алгоритм построения компьютерных моделей в ПК «Ли́ра-Windows».....	171
7.9. Примеры построения моделей в ПК «Ли́ра-Windows» для геомеханических исследований .....	174
7.9.1. Исследование НДС массива и бетонной крепи вертикального ствола .....	174
7.9.2. Исследование НДС массива, вмещающего незакрепленную выработку арочной формы.....	177
7.9.3. Исследование НДС узлов крепления армировки ствола .....	178
<b>8. INTERNET-ТЕХНОЛОГИИ .....</b>	<b>182</b>
8.1. Краткая история создания сети Интернет .....	182
8.2. Адресация в Итернете.....	183
8.3. Функции Интернета.....	184
8.4. Подключение к сети Интернет .....	185
8.5. Поиск информации в Интернет .....	186
8.6. Навигация в World Wide Web.....	188
8.7. Сохранение Web-страниц .....	190
8.8. Электронная почта .....	190
8.9. Некоторые Интернет-ресурсы по горному делу .....	193
<b>ЗАКЛЮЧЕНИЕ.....</b>	<b>200</b>
<b>БИБЛИОГРАФИЧЕСКИЙ СПИСОК .....</b>	<b>201</b>

## ВВЕДЕНИЕ

---

В последнее десятилетие в нашу повседневную жизнь прочно вошло понятие информационных технологий. Без использования современных компьютеров, средств коммуникации, различных информационных сетей и каналов немислимы ни учебный процесс, ни производство, ни управление, ни социально-бытовая сфера.

Изучению современных информационных систем и технологий уделяется все большее внимание при составлении учебных планов подготовки специалистов различных направлений и специальностей. Не обходят стороной информационные технологии горное и строительное производства, поэтому неотъемлемой частью подготовки современных горных инженеров и инженеров-строителей является овладение как общими основами информатики, так и специальными знаниями по применению прикладных компьютерных программ, геоинформационных систем, специальной компьютерной графики, систем компьютерного моделирования и много другого.

Целью настоящего учебного пособия является знакомство студентов с основами современных информационных технологий, которые применяются или могут применяться в горном и строительном производстве. Пособие ориентировано на студентов ВУЗов специальности «Шахтное и подземное строительство», поэтому большинство примеров использования информационных технологий относятся к сооружению горных выработок и других подземных сооружений. Однако авторы надеются, что пособие будет полезно студентам любых горных, строительных и других инженерных специальностей, так как рассматривает общие аспекты применения средств вычислительной техники, компьютерной графики, баз данных, технологий программирования, компьютерного моделирования, Internet-технологий и др.

Каждый раздел пособия заканчивается контрольными вопросами или заданиями, выполнив которые студенты смогут закрепить на практике полученные теоретические знания.

Главы 1, 3 и 5 написаны инженером-маркшейдером, кандидатом технических наук М.В. Прокоповой, главы 2, 4 и 5 – инженером-механиком, кандидатом технических наук Ю.Н. Попковым, главы 6 – 8 – горным инженером-строителем, кандидатом технических наук, доцентом А.Ю. Прокоповым. В настоящее время все авторы являются сотрудниками кафедры «Подземное, промышленное, гражданское строительство и строительные материалы» Шахтинского института (филиала) Южно-Российского государственного технического университета (Новочеркасского политехнического института). Подробная информация об институте – на сайте <http://www.itsinpi.ru/>

Авторы будут благодарны и признательны всем приславшим свои отзывы и замечания по содержанию настоящего пособия, чтобы обязательно учесть их при работе над его последующими изданиями. Связь с авторами – по электронной почте [prokopov72@rambler.ru](mailto:prokopov72@rambler.ru).

# **1. ОБЩИЕ СВЕДЕНИЯ ОБ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЯХ**

---

## **1.1. Технологические основы информатики**

**Информатика** – это наука, изучающая все аспекты получения, хранения, преобразования, передачи и использования информации.

Информатика – практическая наука. Все её теоретические достижения и рекомендации проходят проверку практикой и принимаются лишь в тех случаях, когда обеспечивают повышение производительности труда и экономичности производства, то есть являются эффективными. Стремление повысить производительность технических систем и сократить затраты на их создание и применение проявляется в технологических аспектах информатики.

Под словом **технология** принято понимать **совокупность условий (режимов), приёмов и навыков использования предметов труда (инструментов и материалов) для получения результатов труда в определенном количестве с заданным качеством.** Инструментами труда в информатике являются аппаратные и программные средства вычислительной техники, а материалами – носители данных и структуры данных.

Разнообразная информационная деятельность человека складывается из осуществления трех основных видов информационных процессов: хранения, передачи и обработки информации.

Человек хранит информацию в собственной памяти (внутренняя, оперативная информация) и на внешних носителях: бумаге, магнитной ленте и пр. (внешняя информация).

Процесс передачи информации осуществляется от источника к приемнику по информационным каналам связи.

Процесс обработки информации связан с получением новой или изменением формы или структуры данной информации; осуществлением поиска информации на внешних носителях.

В информатике много технологических разделов – их принято называть **информационными технологиями.**

**Информационные технологии** – это процессы, использующие совокупность средств и методов сбора, обработки и передачи данных для получения информации нового качества о состоянии объекта, процесса или явления.

**Цель информационных технологий** – производство информации для ее анализа человеком и принятия на его основе решения по выполнению какого-либо действия.

Внедрение персонального компьютера в информационную сферу и применение телекоммуникационных средств связи определили новый этап развития информационных технологий – это информационные тех-

нологии с «дружественным» интерфейсом работы пользователя, использующие персональные компьютеры и телекоммуникационные средства.

Современные информационные технологии базируются на следующих основных принципах:

- интерактивный (диалоговый) режим работы с компьютером;
- интегрированность с другими программными продуктами;
- гибкость процесса изменения данных и постановок задач.

В качестве инструментария информационных технологий используются распространенные виды программных продуктов: текстовые процессоры, издательские системы, электронные таблицы, системы управления базами данных, системы автоматизированного проектирования, электронные календари, информационные системы функционального назначения.

Привести исчерпывающий список информационных технологий практически невозможно, потому что ежегодно появляются новые информационные технологии, да и действующие технологии непрерывно совершенствуются и изменяются. Разные информационные технологии взаимно обогащают друг друга. Вот краткий перечень наиболее важных, системообразующих технологий с описанием вопросов, рассматриваемых ими.

- **Технологии администрирования средств вычислительной техники.** В этом разделе информатики решаются вопросы настройки и наладки аппаратных и программных средств, обеспечения бесконфликтного взаимодействия между ними, эффективного распределения физических и логических ресурсов вычислительных систем между программными и аппаратными компонентами.

- **Технологии электронного документооборота.** Этот раздел информатики посвящён эффективному созданию и оформлению электронных документов разных категорий. С одной стороны, его практическая значимость для общества заключается в сокращении расходования невозполнимых природных ресурсов на цели, связанные с документооборотом. С другой стороны, значимость обуславливается повышением производительности труда на всех этапах создания, модификации и распространения документов, а также возможностью создания документов принципиально новых типов, например *мультимедийных*, *Web-документов* и других.

- **Технологии баз данных.** Этот раздел информатики занимается вопросами упорядоченного хранения информационных объектов и организации быстрого доступа к ним. Практическим результатом применения систем управления базами данных является многократное снижение затрат на обретение и отбор необходимой информации.

Виды баз данных: настольные, предназначенные для работы на ПК (ACCESS, FoxPro, Paradox), и серверные (SQL, Oracle, DB2, Informis).



Базы данных представляют собой ряд таблиц, в которых хранится и обрабатывается разнородная информация о большом количестве объектов. Примером может служить информация о движении материалов на складах, когда отслеживается их вид, цена, срок годности, количество, поставщик, реализация, и т.д. Удобны базы данных в управлении любыми структурами горного предприятия или организации, в т.ч. кадровой, бухгалтерской, планово-финансовой и др.

- **Коммуникационные технологии.** Слова *общество* и *общение* — однокоренные, причём не только в русском языке. Наличие общества обусловлено возможностью общения его членов. Если нет возможности общения, то нет общества. Принципы коммуникации с использованием средств вычислительной техники позволяют уже сегодня сделать доступным общение в режиме реального времени для всех людей планеты, независимо от их местонахождения. Более того, коммуникационные технологии позволяют автоматизировать и сам процесс общения. В настоящее время очень часто одной из сторон электронного общения является автоматический процесс, прошедший предварительную настройку.

- **Технологии программирования.** Компьютерная программа — это упорядоченная совокупность команд, реализующая алгоритм. Но реализуются эти команды только в момент исполнения программы. На этапе разработки программы её команды имеют ту же логическую природу, что и данные. Это даёт возможность применять компьютеры для автоматизации разработки новых программ. Один человек физически не может создать за год более 8-15 тысяч строк программного кода, а современные программные комплексы уже измеряются десятками миллионов строк кода. Их создание требует координированной деятельности тысяч программистов в течение нескольких лет. Без средств организации, автоматизации и автоматического контроля деятельность программистов была бы непродуктивной. Систематизация принципов создания и эффективного применения средств программной инженерии является одной из важнейших задач информатики. Первые языки программирования: Fortran, паскаль, Basic, затем появились P11, QBasic, Delphi, Visual Basic, C, C++, C++builder и др.

- **Технологии компьютерной графики.** Работа с изображениями традиционно считалась технологически очень трудной и неэффективной. Создание изображений требовало от исполнителя длительного обучения, а творческий характер работы долгое время препятствовал её автоматизации. Современные технологии компьютерной графики не только дали исполнителям эффективные инструменты для создания и обработки изображений, но и позволили автоматизировать работу художников, дизайнеров, мультипликаторов. В настоящее время широко используются технологии трехмерной графики, позволяющие создавать виртуальные сцены для кинематографии. Уже внедряются технологии, позволяющие заменить съём-

ку реальных актёров автоматической генерацией управляемых графических персонажей.

- **Технологии электронной коммерции.** Успешное развитие производственных технологий на протяжении XX века привело к существенному разрыву между автоматическим характером современного промышленного производства и устаревшим неавтоматизированным (в лучшем случае – механизированным) характером торговых операций. Торговые операции сопровождаются товарным документооборотом и финансовыми расчетами, которые автоматизируются очень трудно, что приводит к неуклонному росту доли накладных расходов, цены приобретаемого товара. Современные технологии электронной коммерции позволяют решить эту проблему. Формирование запасов на складах, контроль над сроками хранения скоропортящихся товаров, приём заказов, обработку сопроводительной документации, управление транспортными потоками, обслуживание платёжных систем — все эти и многие другие операции в режиме электронной коммерции выполняются автоматически.

- **Интернет-технологии.** В Интернете действует множество служб и сервисов. Службы Интернета – это программные комплексы, работающие согласно установленным протоколам. Сервисы Интернета — это услуги, предоставляемые пользователям в рамках действующих служб. Интернет-технологии позволяют реализовать потребности электронного документооборота, общения, электронной коммерции. Важными составляющими Интернет-технологий являются технологии поиска информации, хранящейся в обобщённых ресурсах всемирной Сети, а также технологии, связанные с обеспечением личной и общественной безопасности участников Сети.

## **1.2. Информационные технологии в горном деле**

Информационные технологии предназначены для решения различного вида задач, возникающих на каком-либо этапе горного производства, прежде всего, для информационного обслуживания всех работников предприятий, связанных с принятием управленческих решений. Здесь информация обычно представляется в виде регулярных или специальных управленческих отчетов и содержит сведения о прошлом, настоящем и возможном будущем предприятия. Автоматизация офиса предполагает организацию и поддержку коммуникационных процессов как внутри производства, так и с внешней средой на базе компьютерных сетей и других современных средств передачи и работы с информацией.

Широко используются информационные технологии при проектировании в виде компьютерной графики, моделирования процессов и инженерных расчетов.

Информационные технологии применяются и на уровне исполнительской деятельности персонала невысокой квалификации в целях автоматизации некоторых рутинных, постоянно повторяющихся операций.

**Управление** – это процесс целенаправленного воздействия на объект, организующий функционирование объекта по заданной программе. Информация, которая обеспечивает производство, распределение, обмен и потребление материальных благ и решение задач организационно-экономического управления, называется *управленческой*. В управленческой деятельности информация выступает как один из важнейших ресурсов наряду с энергетическими, материальными, трудовыми, финансовыми.

В горном деле современная система использования информационных технологий представляет собой комплекс со следующими основными подсистемами обеспечения:

- *информационное обеспечение* – система классификации информации, технологическая схема обработки данных, нормативно-справочная информация, система документооборота, создание различного вида документации;
- *организационное обеспечение* – совокупность мер и мероприятий, регламентирующих функционирование системы управления, наличие связи между структурами предприятия;
- *техническое обеспечение* – комплекс используемых в системе технических средств, включающий ЭВМ и средства связи;
- *математическое обеспечение* – совокупность методов, правил, математических моделей и алгоритмов решения задач;
- *программное обеспечение* – совокупность программ, необходимых на всех этапах деятельности предприятия.

В настоящее время существует множество программных продуктов, обеспечивающих информационные технологии обработки различного рода информации. К ним относятся текстовые процессоры, табличные процессоры, системы управления базами данных, системы автоматического проектирования, электронная почта и др.

Информационные подсистемы можно классифицировать по множеству аспектов. Среди инженерных информационных систем выделяются следующие:

- системы обработки данных (СОД);
- системы автоматизированного проектирования (САПР);
- автоматизированные системы управления (АСУ);
- информационно-поисковые системы (ИПС).

СОД производит информационное обслуживание специалистов органа управления объектом, принимающих управленческие решения. Решение, принятое на основе представленной информации, передается на управляемый объект, минуя СОД. Можно трактовать СОД как систему,

которая преобразует поток входной информации в поток выходной информации.

Если СОД способна выполнять выбор управленческих решений, то она становится автоматизированной системой управления. Принятие решений АСУ может производиться на основе экономико-математических методов или путем моделирования действий специалиста по принятию управленческого решения. Прикладные программы АСУ, формирующие управленческое решение, как правило, используют экономико-математические методы для выбора оптимальных решений. Исходные данные для оптимизационной задачи рассчитываются в режиме системы обработки данных. Моделирование принятия решений специалистом реализуется в так называемых экспертных системах, которые построены на принципах искусственного интеллекта и баз знаний.

Применение информационных технологий в горном производстве необходимо на любом этапе проектирования, строительства, эксплуатации и основано на подборе и формировании технического и информационного, математического, программного и организационно-правового обеспечения.

Подбор технического обеспечения должен быть таким, чтобы обеспечить своевременный сбор, регистрацию, передачу, хранение, наполнение и обработку информации.

Информационное обеспечение должно предусматривать создание и функционирование единого информационного фонда системы, представленного множеством информационных массивов, набором данных или базой данных.

Формирование математического обеспечения систем включает комплектацию методов и алгоритмов решения функции начальных задач. При формировании программного обеспечения систем особое внимание обращается на создание комплекса программ и инструкций пользователя и выбор эффективных программных продуктов.

Системы автоматизированного проектирования (САПР) давно и широко применялись в проектировании различных комплексов отраслей народного хозяйства, в том числе и в горном деле. В современном мире невозможно достичь высокого уровня конструирования без использования САПР, которые обеспечивают максимальную точность выполнения чертежей и экономят время на многочисленных рутинных операциях. Создаваемые с помощью САПР результаты можно передавать по технологической цепочке для выполнения последующих операций. Существует много графических редакторов и программ геометрического моделирования (AutoCad, SolidWorks, Компас и др.), а также программ производства трехмерной графики (3DStudio Max, Maya). Лидером среди систем автоматизированного проектирования можно считать систему AutoCad. Для моделирования различных процессов созданы программно-вычислительные комплексы (ПВК) «Зенит», Лира–Windows, StructureCAD и др.

Для осуществления различной деятельности используются текстовые процессоры, которые предназначены для создания и обработки текстовых документов. Подготовленные текстовые документы могут быть распечатаны, а также переданы по компьютерной сети.

Табличные процессоры позволяют выполнять многочисленные операции над данными, представленными в табличной форме. Пользователь имеет возможность вводить табличные данные, обрабатывать их, проводить необходимые вычисления, автоматически формировать итоги, выводить информацию в печатном виде и в виде импортируемых в другие системы файлов, качественно оформлять табличные данные, в том числе в виде графиков и диаграмм, проводить инженерные, финансовые, статистические расчеты, проводить математическое моделирование и т. д.

Системы управления базами данных предназначены для создания и поддержания в актуальном состоянии баз данных, содержащих различные сведения о системе управления и производственной деятельности фирмы. Электронная почта позволяет пользователю получать, хранить и отправлять сообщения своим партнерам по сети. Возможности, предоставляемые пользователю электронной почтой, различны и зависят от применяемого программного обеспечения.

Таким образом, можно сделать вывод о том, что все вышеперечисленные информационные технологии нашли применение в горном деле.

### *Контрольные вопросы*

1. Что изучает информатика?
2. Дайте определение технологии.
3. Какова цель информационных технологий?
4. Назовите современные информационные технологии.
5. Какие инженерные информационные системы вы знаете? Как они применяются в горном деле?

## 2. АДМИНИСТРИРОВАНИЕ СРЕДСТВ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ И СЕТЕЙ

---

### 2.1. Аппаратное обеспечение

Обязательный комплект, составляющий персональный компьютер : системный блок, клавиатура, монитор. В состав системного блока входят: микропроцессор, внутренняя память, дисководы, блок питания, контроллеры внешних устройств.

Все устройства ПК связаны между собой по многопроводной линии, которая называется информационной магистралью, или шиной. Каждое внешнее устройство имеет свой адрес (номер). Передаваемая к нему информация по шине данных сопровождается адресом устройства – по адресной шине.

Основными техническими характеристиками ПК являются: объем внутренней памяти, тактовая частота микропроцессора, разрядность микропроцессора.

Компьютер собрали на заводе из микросхем, проводов и плат. В память компьютера заносятся программы, необходимые для их работы.

Вся совокупность программ, хранящихся на всех устройствах долговременной памяти компьютера, составляет его **программное обеспечение** (ПО). Оно может пополняться, развиваться, изменяться точно так же, как знания человека. В программном обеспечении компьютера есть необходимая часть, без которой на нем просто ничего не сделать. Она называется **системным ПО**. Покупатель приобретает компьютер, оснащенный системным программным обеспечением, которое не менее важно для работы ЭВМ, чем память или процессор. Кроме системного в состав программного обеспечения входят: прикладное ПО и системы программирования.

**Прикладное ПО** – программы, с помощью которых пользователь может решать свои информационные задачи, не прибегая к программированию, называются прикладными программами.

Как правило, все пользователи предпочитают иметь набор прикладных программ, который нужен практически каждому. Их называют программами общего назначения. К их числу относятся

– *текстовые и графические редакторы*, с помощью которых можно готовить различные тексты, создавать рисунки, строить чертежи; проще говоря, писать, чертить, рисовать;

– *системы управления базами данных (СУБД)*, позволяющие превратить компьютер в справочник по любой теме;

– *табличные процессоры*, позволяющие организовывать очень распространенные на практике табличные расчеты;

– *коммуникационные (сетевые) программы*, предназначенные для обмена информацией с другими компьютерами, объединенными с данным в компьютерную сеть.

Кроме того, имеется большое количество *прикладных программ специального назначения* для профессиональной деятельности. Их часто называют пакетами прикладных программ. Например: бухгалтерские программы, производящие начисления заработной платы и другие расчеты, которые делаются в бухгалтериях; системы автоматизированного проектирования, которые помогают конструкторам разрабатывать проекты различных технических устройств; пакеты, позволяющие решать сложные математические задачи, не составляя программ; экспертные системы и многое другое.

Очень популярным видом прикладного программного обеспечения являются *компьютерные игры*.

Компьютер представляет собой единство двух составляющих: аппаратуры и программного обеспечения (ПО).

Программное обеспечение компьютера – это вся совокупность программ, хранящихся в его долговременной памяти. ПО компьютера может пополняться, изменяться.

## **2.2. Операционная система**

Системное программное обеспечение – это необходимая часть ПО, без которой не может работать компьютер, главной частью которого является **операционная система (ОС)**.

Любая операционная система организует взаимодействие компьютера с пользователем, управляет работой различных его устройств. Поэтому появление новых ОС тесно связано с развитием аппаратной части компьютера. Создание новых моделей микропроцессоров, новых типов носителей данных способствует разработке и развитию операционных систем.

Операционная система (ОС) представляет собой совокупность программ, выполняющих две основные функции: предоставление пользователю удобств виртуальной машины и повышение эффективности использования компьютера при рациональном управлении его ресурсами.

Виртуальная машина – это функциональный эквивалент воображаемого компьютера с заданной конфигурацией, моделируемый программно-аппаратными средствами реального компьютера. ОС скрывает от пользователя особенности физического расположения информации на дисках и осуществляет обработку прерываний (прекращение вычислительного процесса, вызванное требованиями на обслуживание других устройств), управление таймерами и оперативной памятью. В результате пользователю предоставляется виртуальная машина, реализующая работу на логическом уровне.

К современным операционным системам предъявляются следующие требования:

- *совместимость* – ОС должна включать средства для выполнения приложений, подготовленных для других ОС;
- *переносимость* – обеспечение возможности переноса ОС с одной аппаратной платформы на другую;
- *надежность и отказоустойчивость* – предполагает защиту ОС от внутренних и внешних ошибок, сбоев и отказов;
- *безопасность* – ОС должна содержать средства защиты ресурсов одних пользователей от других;
- *расширяемость* – ОС должна обеспечивать удобство внесения последующих изменений и дополнений;
- *производительность* – система должна обладать достаточным быстродействием.

По числу одновременно выполняемых задач выделяют ОС однозадачные (MS DOS, ранние версии PC DOS) и многозадачные (OS/2, UNIX, Windows).

Однозадачные ОС предоставляют пользователю виртуальную машину и включают средства управления файлами, периферийными устройствами и средства общения с пользователем. Многозадачные ОС дополнительно управляют разделением между задачами совместно используемых ресурсов. Многозадачность бывает не вытесняющая (NetWare, Windows 3/95/98) и вытесняющая (Windows NT, OS/2, UNIX). В первом случае активный процесс по окончании сам передает управление ОС для выбора из очереди другого процесса. Во втором – решение о переключении процессора с одного процесса на другой принимает ОС.

По числу одновременно работающих пользователей ОС делятся на однопользовательские (MS DOS, Windows, ранние версии OS/2) и многопользовательские (UNIX, WINDOWS NT). В многопользовательских системах присутствуют средства защиты информации пользователей от несанкционированного доступа.

В сетевой ОС присутствуют средства передачи данных между компьютерами по линиям связи и реализация протоколов передачи данных.

Кроме ОС, ориентированных на определенный тип аппаратной платформы, существуют мобильные ОС, легко переносимые на разные типы компьютеров (UNIX). В таких ОС аппаратно-зависимые места локализованы и при перекосе системы переписываются. Аппаратно-независимая часть реализуется на языке программирования высокого уровня, как правило, на языке Си, и перекомпилируется при переходе на другую платформу.

В настоящий момент около 90% компьютеров используют ОС Windows. Более широкий класс ОС ориентирован для использования на серверах. К этому классу ОС относятся семейство UNIX, разработки фирмы *Microsoft* (MS DOS и Windows), сетевые продукты Novell и корпорации IBM.



## **2.3. Автоматизированные и автоматические системы управления**

Компьютеры помогают решать задачи управления в самых разных масштабах: от управления станком или транспортным средством до управления производственным процессом на предприятии или даже целой отраслью экономики государства.

Конечно, поручать компьютеру полностью, без участия человека, руководить предприятием или отраслью экономики – сложно, да и небезопасно. Для управления в таком масштабе создаются компьютерные системы, которые называются *автоматизированными системами управления (АСУ)*. Такие системы работают вместе с человеком. АСУ помогает руководителю получить необходимую информацию для принятия управляющего решения, а также может предложить наиболее оптимальные варианты таких решений. Однако окончательное решение принимает человек.

В АСУ используются самые современные средства информационных технологий: базы данных и экспертные системы, методы математического моделирования, машинная графика и пр.

С распространением персональных компьютеров технической основой АСУ стали компьютерные сети. В рамках одного предприятия – это локальные компьютерные сети. Автоматизированные системы управления, работающие в масштабах отрасли, в государственных масштабах, используют глобальные сети ЭВМ.

## **2.4. Компьютерные сети**

### **2.4.1. Основные понятия**

*Сеть ЭВМ* – комплекс аппаратного и программного обеспечения, поддерживающий функции обмена информацией между отдельно расположенными (на расстояниях от нескольких метров до тысяч километров) компьютерами. *Сеть с централизованным управлением* содержит одну или более выделенных ЭВМ (серверов), управляющих обменом по сети (остальные ЭВМ в этом случае называются *рабочими станциями*), *одноранговая сеть* не содержит выделенных машин (функции управления сетью осуществляются рабочими станциями поочередно).

Соответственно *программное обеспечение компьютерных сетей* – комплекс программ, поддерживающий функции обмена информацией между отдельно расположенными ЭВМ. В настоящее время *программное обеспечение компьютерных сетей* обычно является (иногда опционально устанавливаемой) составной частью операционных систем.

*Локальная вычислительная сеть (ЛВС)* – система связи отдельно расположенных ЭВМ на относительно небольшом расстоянии (как прави-

ло, в пределах помещения и/или этажа здания); обычно объединяет до нескольких десятков (чаще однотипных) компьютеров, физическая линия связи – двухпроводной кабель или коаксиальный кабель.

*Корпоративная вычислительная сеть* – сеть, работающая по протоколу TCP/IP и не обязательно подключенная к Internet, но использующая коммуникационные стандарты Internet и сервисные приложения, обеспечивающие доставку данных пользователям сети; эксплуатируется в пределах (крупной) организации.

*Глобальная вычислительная сеть* объединяет множество локальных сетей и сотни тысяч – миллионы разнотипных ЭВМ по всему миру; физическая линия связи – оптокабель или космическая радиопередача.

*Рабочая группа (workgroup)* – набор компьютеров, объединенных для удобства при просмотре сетевых ресурсов одним именем.

*Домен (domain)* – определенная администратором сети совокупность компьютеров, использующих в операционной системе WINDOWS NT Server общую базу данных и систему защиты; каждый домен имеет уникальное имя.

*Узел (host)* – подключенное к сети устройство (обычно компьютер), идентифицируемое собственным *адресом* (например, в сети InterNet host-адресом является уникальное 32-разрядное двоичное число).

*Скорость передачи данных* по компьютерной сети измеряется в *битах в секунду* (bps - *bit per second*) или *бод* (*boud*).

*Трафик (traffic)* – поток сообщений в *разделяемой среде передачи данных*, часто используется для грубой оценки *уровня использования передающей среды* (тяжелый, средний, легкий трафик).

*Серверная ЭВМ* – компьютер (обычно обладающий высоким быстродействием и значительным объемом оперативной и дисковой памяти) и выполняющий запросы, поступающие с *клиентских ЭВМ*.

*Файл-сервер* – выделенная ЭВМ, выполняющая функции хранения данных и программ, используемых пользователями на клиентских ЭВМ.

*Серверное приложение* – выполняющееся ЭВМ приложение, могущее выполнять запросы, генерируемые другим (выполняющимся на данной или удаленной ЭВМ) *приложением-клиентом*.

*Клиентская ЭВМ* – пользовательский компьютер (обычно обладающий ограниченными ресурсами), выдающий запросы для исполнения серверу.

*Клиентское приложение* – приложение, обращающееся (с целью выполнения отдельных функций) к другому *приложению-серверу* (и обычно инициирующее начало его выполнения и завершение).

*Протокол (коммуникационный)* – набор правил и соглашений, согласно которому взаимодействуют два (или более) компьютера.

*Топология (topology) сети* – физическая конфигурация машин в сети.

*Временное уплотнение* при передаче данных – метод передачи данных по линии связи, основанный на последовательной (по времени) передаче пакетов (порций) данных, причем каждый пакет снабжен маркером (в

состав которого входит адрес, идентифицирующий машину-получателя пакета и некоторая дополнительная информация). Временное уплотнение является стандартом для *систем коллективного пользования*, при этом множество пользователей получают высокоскоростной канал, доступный в течение всего времени (но по отношению к каждому из них канал имеет очень низкий показатель использования).

*Маршрутизация* – процесс определения (оптимального) пути доступа к объектам (компьютерам) сети.

*Пакет (датаграмма)* – определенное количество байт, сгруппированное вместе и посылаемое одновременно (практически все сети коммуникаций передают данные небольшими частями – пакетами или датаграммами).

*Администрирование сети* – организация управления объектами сети через ее субъекты. К объектам сети относятся сетевые серверы, рабочие станции, маршрутизаторы и шлюзы, т. е. технические компоненты сети. Управление этими объектами осуществляется через субъекты сети, к которым относятся администратор сети, менеджер рабочей группы, регулярный пользователь сети, оператор сети. Администрирование сети может осуществляться непосредственно через администратора (малые и средние сети) или через организацию специальных структурных подразделений (большие сети).

#### **2.4.2. Основные протоколы, применяемые в компьютерных сетях**

*Протокол* – совокупность правил, регламентирующих процедуру коммуникации. Различают *протоколы физического уровня* (Ethernet, Token-Ring, ARCnet и т.д.), определяющие именно физические правила сетевых соединений (уровень, полярность, длительность сигналов и т.п.) и реализуемые сетевыми картами и иной сетевой аппаратурой (сетевым ПО), и *протоколы высоких* (примыкающих к транспортному и выше) *уровней* (XNS, IP/TCP и т.д.), определяющие логическую структуру сообщений и реализуемые в основном программным путем.

Microsoft Corp. штатно предоставляет следующие сетевые транспортные протоколы:

- транспорт NetBEUI (*NetBios Extended User Interface transport*) – транспортный протокол локальной сети, созданный для работы совместно с сетевым интерфейсом NetBIOS фирмы Microsoft Corp.;
- транспорт TCP/IP (*Transmission Control Protocol / Internet Protocol transport*). Разработанный для Министерства обороны США протокол, предназначенный для соединения разнородных систем через глобальные сети. TCP/IP широко распространен в сетях UNIX и позволяет WINDOWS'NT взаимодействовать с различными сервисами на UNIX-машинах.

Протокол TCP/IP фактически представляет собой два различных протокола, работающих совместно, – не гарантирующий доставку пакетов данных по сети протокол IP (*Internet Protocol*) и гарантирующий доставку

пакетов в правильной последовательности протокол TCP (*Transmission Control Protocol*). В свою очередь, протокол TCP/IP может служить носителем ('оберткой') для других протоколов (протоколов IPX, NetBIOS, служебных протоколов адресации ARP (*Address Resolution Protocol*) и протокола межсетевых управляющих сообщений ICMP (*Internet Control Message Protocol*). В локальной сети TCP/IP-пакеты упаковываются в 'обертку' пакетов Ethernet, сами же TCP/IP-пакеты являются 'оберткой' для HTTP.

Протокол SLIP (*Serial Line Internet Protocol*) позволяет изолированным компьютерам связываться с TCP/IP через телефонную сеть. Этот протокол определяет метод разбиения датаграмм на фреймы при передаче их по последовательному каналу и указывает конец одной и начало другой датаграммы. Хотя протокол SLIP вполне подходит для установления связи с дисковым набором, но недостатки в адресации, идентификации типа и сжатии данных делают его негибким, медленным и трудным в конфигурации.

Межузловой протокол PPP (*Point-to-Point Protocol*) был разработан для устранения недостатков SLIP; для PPP разработано несколько расширений, таких как опция предоставления имен серверов (DNS, *Domain Names Service - служба доменных имен*), обеспечение безопасной идентификации пользователя и объединение многочисленных соединений в одно логическое соединение с повышенной полосой пропускания [9].

Среди прочих существующих или находящихся в стадии разработки протоколов разработки фирмы Microsoft Corp. и другими фирмами можно назвать:

- IPX/SPX (*Internet Packet eXchange / Sequest Packet eXchange*) – набор транспортных протоколов, используемых программным обеспечением NetWare фирмы Novell Corp [10, 11];
- DECnet – используемый фирмой Digital Equipment Corp. транспортный протокол, предназначенный для связи систем Windows'NT с сетями DECnet;
- AppleTalk – разработанный фирмой Apple Corp., Inc протокол для взаимодействия WINDOWS'NT с компьютерами Apple Macintosh;
- XNS (Xerox Network Systems) – транспортный протокол, разработанный фирмой Xerox Corp. и использовавшийся в первых сетях Ethernet.

В Internet часто применяются следующие протоколы:

- \* TIME – наиболее простой протокол, с помощью которого извлекаются данные времени из соответствующего сервера;
- \* Протокол FTP (*File Transfer Protocol*), мало изменившийся с начала 70-х годов, работает на всех компьютерах (от платформ PC DOS до суперкомпьютеров); данный протокол считается устаревшим, т.к. не может взять на себя приоритет управления трафиком и возобновлять несостоявшиеся (аварийно завершённые) передачи данных;
- \* Протокол SMTP (*Simple Mail Transfer Protocol*) известен с 1980 года и был рассчитан на обмен почтой между «большими» ЭВМ (*mainframe*

- *мэйнфрейм*), которые имеют постоянное соединение (но не на имеющиеся случайное, непостоянное соединение ПЭВМ);
- \* NNTP (*Network News Transfer Protocol*) – относительно сложный протокол, служит для передачи новостей между серверами новостей и от сервера к клиенту;
- \* Протокол POP (*Post Office Protocol*) снял в 1984 году ограничения протокола SMTP путем добавления двух новых функций – восстановления всех сообщений (в случае аварии) и удаления их с сервера (при успехе передачи). Текущая версия, POP3, добавляет несколько новых характеристик, сохраняя многое из структуры первоначальной версии. Однако и SMTP и POP поддерживают только поток текста ASCII и не стандартизируют обмен данными современных форматов;
- \* Формат MIME (*Multipurpose Internet Mail Extension*) появился в 1992 году и снял ограничения SMTP и POP в области передачи двоичных файлов (графика, мультимедиа и др.); промежуточные (взятые из ОС UNIX) преобразования «формат ASCII ↔ двоичный формат» UUEncode и UUDecode в былое время широко применялись в FidoNet;
- \* HTTP (*HiperText Transfer Protocol*) – протокол передачи гипертекста в InterNet; информацию о самой последней версии HTTP можно получить непосредственно от рабочей группы HTTP по адресу [www.ics.uci.edu/pub/ietf/http](http://www.ics.uci.edu/pub/ietf/http).

Не следует считать, что существующие протоколы исчерпывают все возможности сетевого обмена данными. Любой (подготовленный) разработчик ПО может предложить собственный (позволяющий успешно выполнять некоторые специфические действия) протокол; в случае действительного востребования этой разработки она будет признана стандартом (и включена в список документов RFC).

Например, в настоящее время используется версия 1.1 протокола HTTP. Ее поддерживают все основные браузеры и WEB-серверы. Протокол HTTP 1.1 описан в RFC-2068 и превосходит предыдущую версию HTTP 1.0 – прежде всего, по производительности. Однако есть и другие отличия, описанные ниже:

- \* Постоянные соединения. Протокол HTTP 1.1 устанавливает меньше ТСР-соединений, чем HTTP 1.0. Версия 1.0 устанавливает и разрывает ТСР-соединение для каждого HTML-запроса, а HTTP 1.1 создает ТСР-соединение, сохраняющееся на протяжении многих запросов;
- \* Протокол HTTP 1.1 поддерживает сжатие данных. Это означает, что файлы между клиентом и сервером могут передаваться сжатыми, что снижает нагрузку на сеть;
- \* Протокол HTTP 1.1 поддерживает многие языки сетевого программирования;
- \* Создание виртуальных хостов. Протокол HTTP 1.1 позволяет одному WEB-серверу иметь несколько доменных имен. В настоящее время

эта ситуация распространена широко (например, когда поставщик услуг InterNet'а поддерживает несколько доменов);

Консорциум W3C работает над протоколом HTTP-NG (*Next Generation*), который, как предполагается, заменит HTTP. К HTTP-NG предъявляются следующие требования:

- \* Простота – протокол HTTP-NG должен быть прост для реализации и обслуживания;
- \* Расширяемость – на случай ситуации, не предусмотренной в процессе разработки;
- \* Масштабируемость – вне зависимости от того, используется ли HTTP-NG в маленькой локальной сети или в сети InterNet;
- \* Эффективность – ожидается, что протокол HTTP-NG будет намного эффективнее HTTP. Последний плохо работает в сетях с большим временем задержки. Причина в том, что HTTP – протокол одиночных запросов и ответов. Кроме того, он перегружен информацией. Протокол HTTP-NG призван устранить эти и другие недостатки.

В последнее десятилетие наметилась тенденция разработки формальных методов описания протоколов, значительно упрощающих создание и тестирование новых протоколов передачи данных, требования обеспечения конфиденциальности передаваемой по сетям информации инициировали разработку новых протоколов обеспечения секретности.

### **2.4.3. Физическая установка сети**

Для упрощения, предположим, что мы используем в сети три компьютера – два «клиентских компьютера» и один «сервер». Для организации, даже такой маленькой сети, нам понадобятся сетевые карты, которые устанавливаются в каждый компьютер, свич или хаб, а так же специальный сетевой кабель, называемый витая пара.

*Выбор сетевой карты.* В прошлом сетевая карта представляла собой отдельную ISA (это старый стандарт слота расширения) или PCI плату и была относительно дорогостоящей. Сегодня сетевая карта стала настолько доступной, что частенько ее интегрируют на системную плату. В случае, если плата не имеет интегрированного сетевого контроллера, то используют внешнюю PCI плату. Карта должна соответствовать стандарту Realtek 10/100. Сетевые карты могут иметь множество дополнительных функций, обеспечивающих большую стабильность при передаче данных.

При выборе сетевой карты необходимо обратить внимание на максимальную поддерживаемую скорость передачи 10/100. Это означает, что карта может передавать данные на скорости 10 mbps и 100 mbps в зависимости от сетевой архитектуры. Дополнительно необходимо, чтобы карточ-

ка имела разъем RJ-45 (современный стандарт CAT5), поддерживающий скорость передачи 100 mbps и обратно совместимый с 10 mbps стандартом. Самый последний стандарт «CAT6» (пока находится в разработке) будет поддерживать скорости от 300 mbps до 1 gbps. Это означает, что Вы сможете копировать файлы в сети со скоростью 125 МВ в секунду. Это быстрее скорости современных жестких дисков. CAT5-E или категория 5 UTP Enhanced так же обратно совместима с 10, 100 и 1000Mbit Ethernet. Использование RJ-45 предпочтительнее, чем RJ-58, или больше известные как BNC. Этот разъем предназначен для подключения на более низких скоростях (ограничена 10 mbps) коаксиальным кабелем.

*Хаб против свича.* Для управления всеми транзакциями (передача блоков информации) в сети используется устройство, называемое *хаб* или *свич*. В чем отличие между этими двумя устройствами? Во время передачи пакета данных хаб отправляет их сразу на все компьютеры, что значительно уменьшает пропускную способность канала. Свич имеет встроенную память, в которой хранится информация о том, к какому порту подключен какой компьютер. Поэтому во время передачи пакета он отправляется на определенный порт. Кроме того, свич позволяет использовать в сети контроллеры с разной скоростью передачи, при этом общая пропускная способность не будет опускаться до уровня контроллера с минимальной скоростью. На сегодняшний день выпускаются свичи с 5, 8, 16, 24 или 32 портами. Цена устройства напрямую зависит от количества портов.

*802.11a и 802.11b.* Говоря об организации сети, необходимо затронуть вопрос беспроводных сетей, т.е. передающих информацию по радиоканалу. Такие сети становятся все популярнее, т.к. позволяют сделать размещение компьютеров более гибким, а пользователям, использующим карманные компьютеры или ноутбуки, получить доступ к сети в любой точке офиса и даже за пределами. На сегодняшний день существует два стандарта 802.11a и 802.11b. Для реализации такого подключения необходимо использовать специальные хабы и сетевые карты.

#### ***2.4.4. Рекомендации по конфигурированию сервера***

Если сервер не планируется использовать для игр, нет необходимости использовать самый последний процессор Pentium 4 или Athlon XP с большим объемом оперативной памяти и емким жестким диском, а также с самой последней графической картой GeForce 4. Главное, чего необходимо добиться, это стабильной круглосуточной работы. Так как сервер будет использовать Windows XP, он должен быть оснащен минимум 128 MB RAM.

В идеале, можно использовать процессор Pentium 2, III или Athlon 500 MHz, при этом система будет достаточно хорошо работать.

*Конфигурация сети в Windows XP.* Существует три способа запуска мастера настройки сети Network Setup Wizard – запуск приложения с ло-

кального жесткого диска, CD или флоппи-диска, содержащего файлы установки (создаваемые при запуске Мастера с локального жесткого диска машины под управлением Windows XP). Рабочие станции под управлением Windows XP поддерживают все три варианта, в то время как Windows 98 и Windows Millennium – только установку с CD и флоппи-диска.

Для запуска Network Setup Wizard с жесткого диска под Windows XP нужно выполнить следующие шаги:

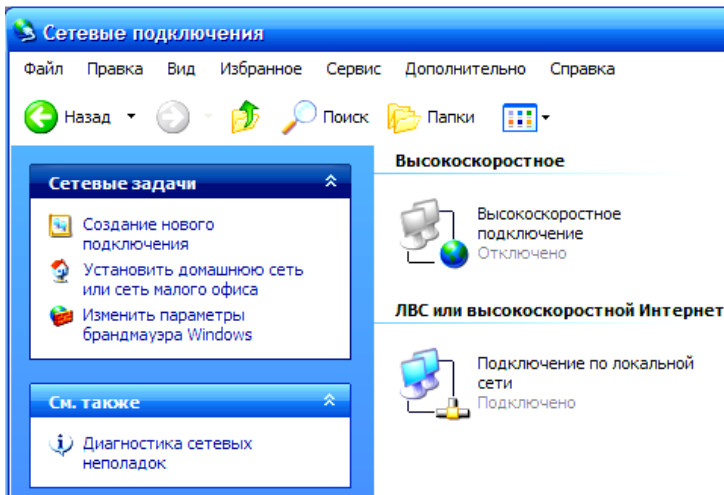


Рис. 2.1. Запуск Network Setup Wizard

1. Откройте папку Network Connections (Сетевые подключения) в Control Panel (Панель управления). Выберите **Set up a home or small office network** (Установить домашнюю сеть или сеть малого офиса) в разделе **Network Tasks** (Сетевые задачи), как показано на рис. 2.1.

В качестве альтернативы можно нажать кнопку **Start** (Пуск), затем **All Programs** (Все программы) или **Programs** (Программы), если Вы используете классический стиль меню. Затем выберите пункт меню **Accessories** → **Communications** → **Network Setup Wizard** (Стандартные → Связь → Мастер настройки сети).

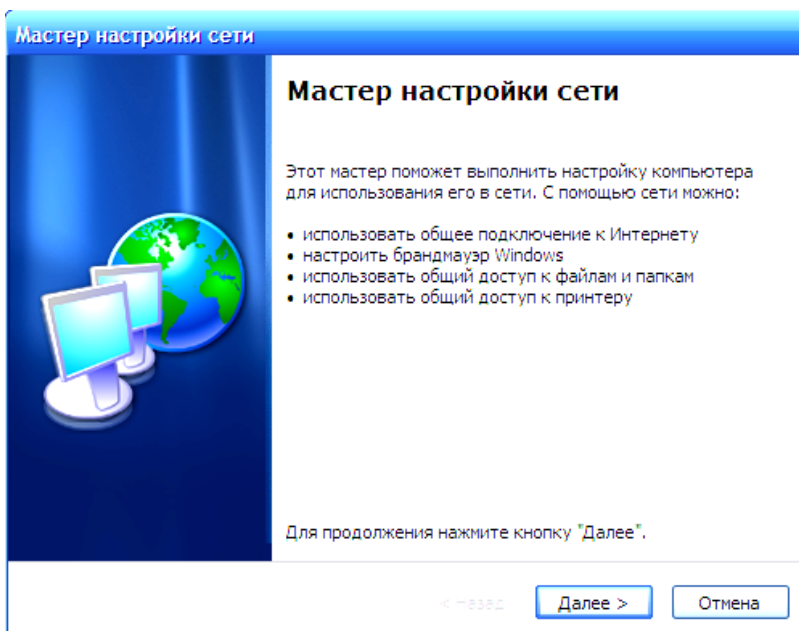


Рис. 2.2. Окно приветствия мастера настройки сети (шаг 1)

2. Появится окно приветствия Мастера, показанное на рис. 2.2. Нажмите кнопку **Next** (Далее) для продолжения.



Мастер отобразит список предварительных этапов, показанный на рис. 2.3.

3. Перед тем как продолжить настройку, воспользуйтесь ссылкой **check list for creating network** (Контрольный список: установка сети).

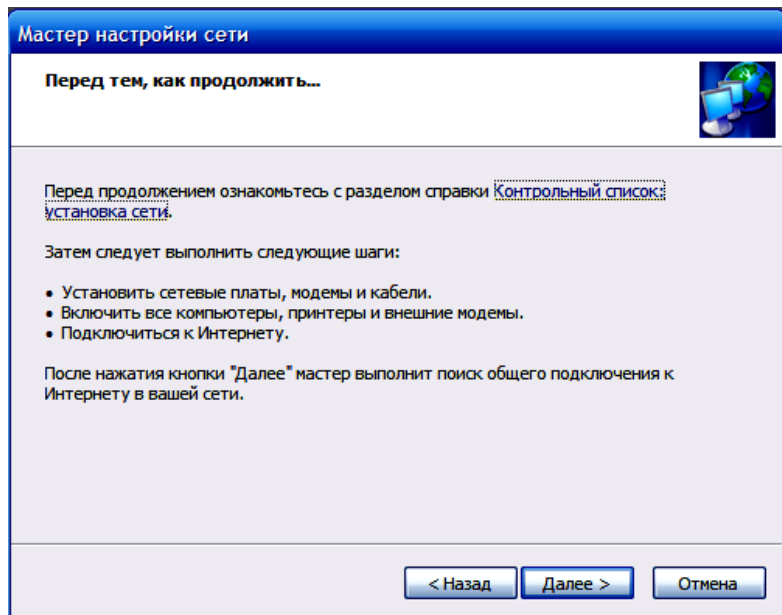


Рис. 2.3. Окно работы мастера настройки сети (шаг 2)

4. Windows XP отобразит для вас список действий, необходимых для подготовки к настройке сети, изображенный на рис. 2.4.

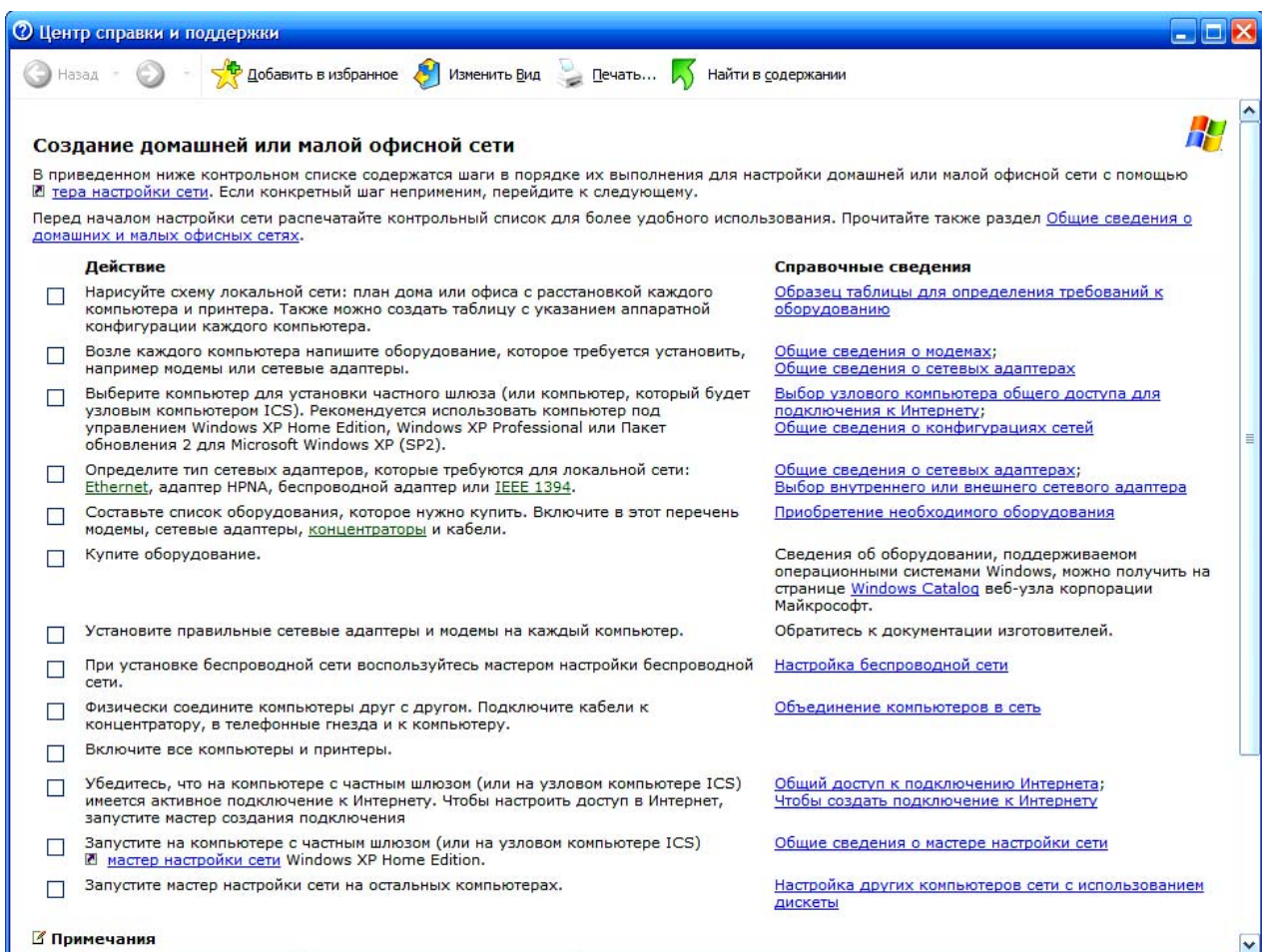


Рис. 2.4. Окно отображения последовательности настройки сети

Если в данном пункте Вы все выполнили, нажмите кнопку **Next** (Далее).

5. Если после просмотра Вы обнаружили, что решили не все вопросы в списке, щелкните по кнопке **Cancel** (Отмена) для выхода из Мастера. Когда Вы выполните все требования по подготовке, перезапустите Мастер.

6. Выберите метод подключения.

Для этого придется ответить на несколько вопросов по сетевому сценарию. Продолжите работу Мастера, предоставляя ответы, соответствующие вашей сети. Когда Вы ответите на все вопросы, то получите напоминание просмотреть функции конфигурации, выбранные вами, перед тем как активировать эти изменения в конфигурации.

Когда изменения в конфигурации произведены, Мастер напомнит вам о необходимости запустить Мастер на всех остальных компьютерах в вашей сети. Выберите подходящую для вас функцию и нажмите кнопку **Next** (Далее).

7. Следуйте появляющимся на экране инструкциям в соответствии с вашим выбором. Щелкните по кнопке **Close** (Закреть) в последнем окне Мастера.

*Мастер сетевой идентификации.* Теперь необходимо установить имя Вашего компьютера в рабочей группе, к которой он принадлежит. Для этого нажмите **Start** → **Settings** → **Control Panel** → **System** (*Пуск*→*Панель управления*→*Система*) и выберите закладку **Computer Name** (*Имя компьютера*), рис. 2.5.

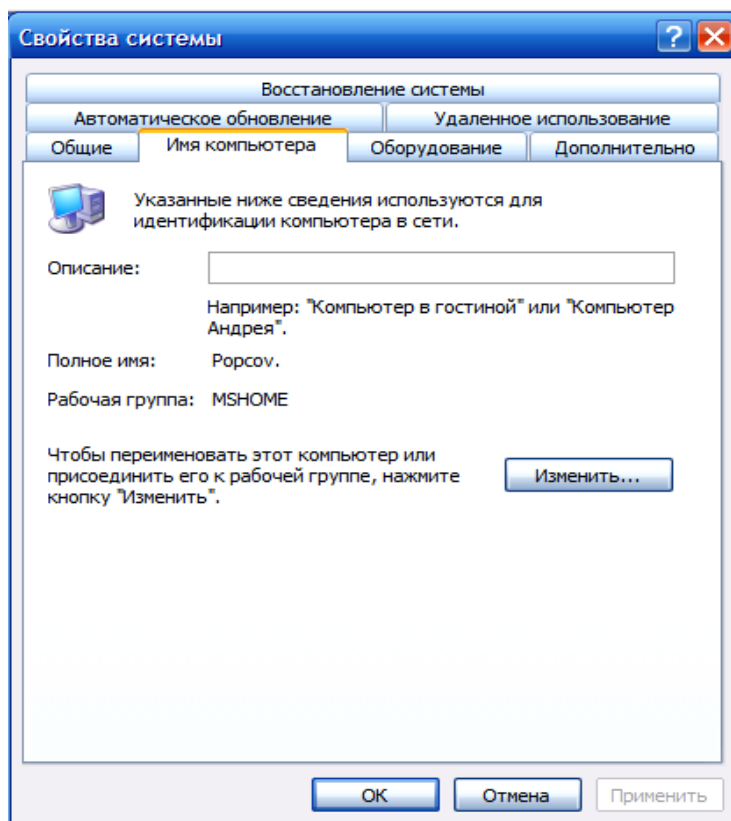


Рис. 2.5. Окно установки имени компьютера

Сначала нажмите «**Network ID**», что позволит активизировать мастер сетевой идентификации (Network Identification Wizard), рис. 2.6.

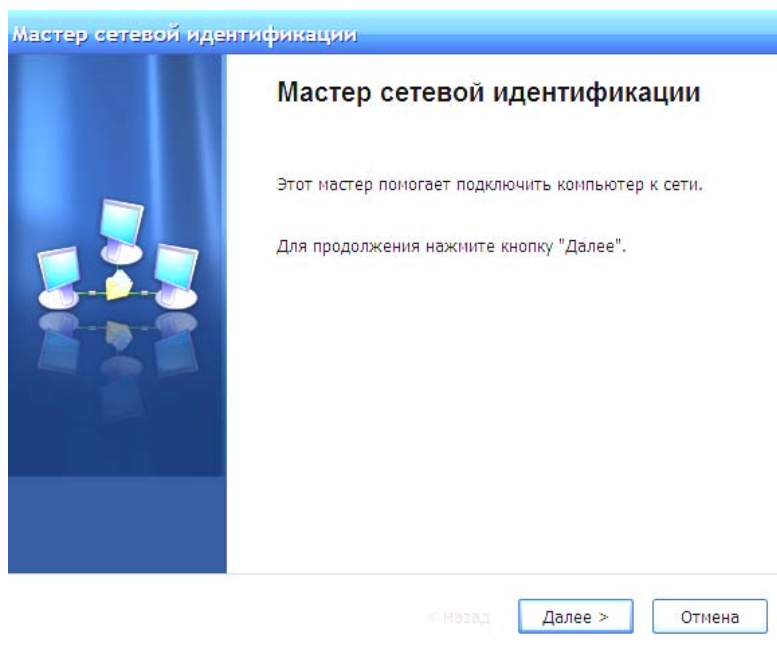


Рис. 2.6. Начало работы мастера сетевой идентификации

На первом экране просто нажмите **Next (Далее)**.

На следующем экране (рис. 2.7) нужно выбрать первую опцию (**This computer is part of a business network, and I use it to connect to other computers at work**) (*Компьютер входит в корпоративную сеть, и во время работы я использую его для соединения с другими компьютерами*).

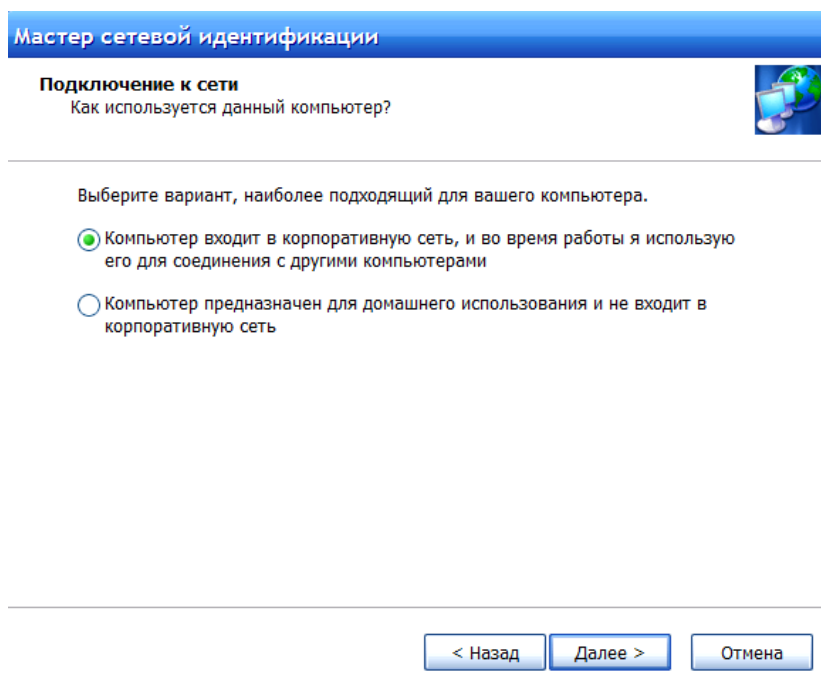
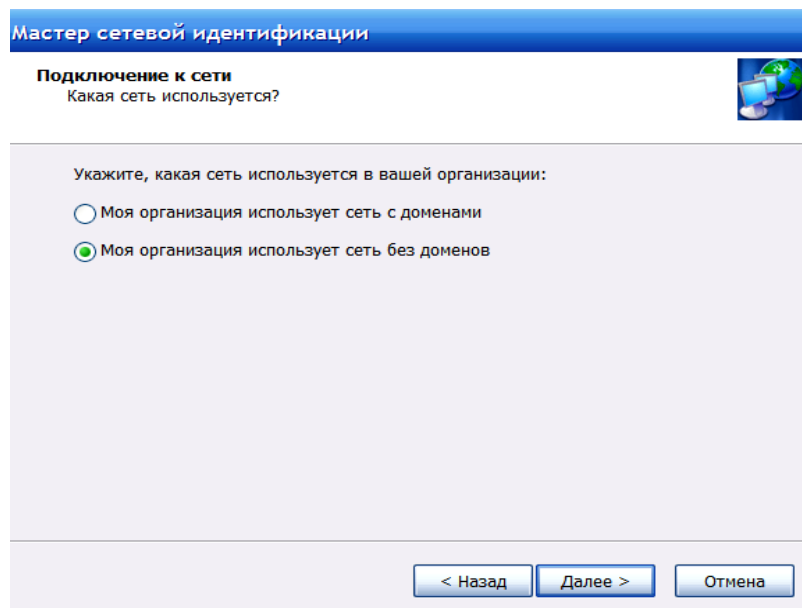


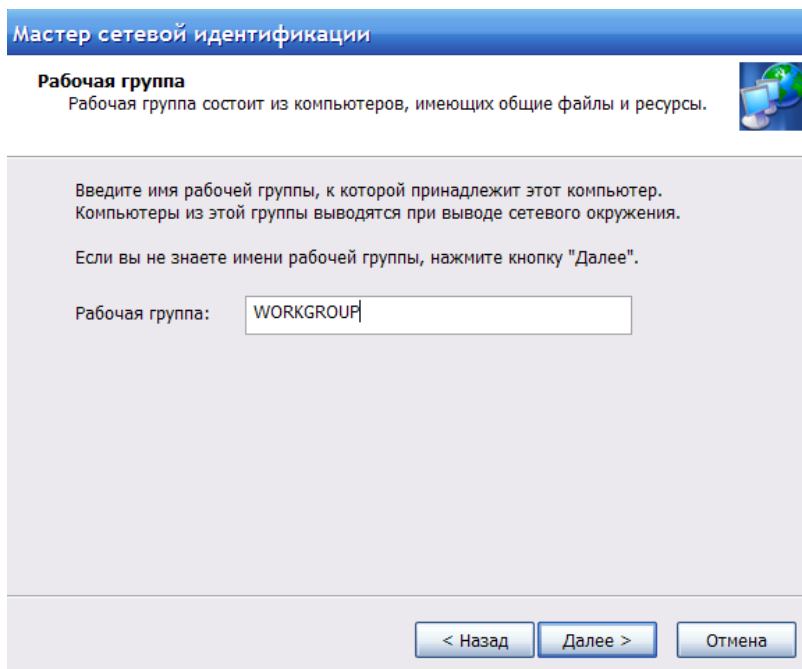
Рис. 2.7. Окно выбора варианта подключения к сети

На следующем экране (рис. 2.8) выберете вторую опцию (**My company uses a network without a domain**)(*Моя организация использует сеть без доменов*).



**Рис. 2.8. Окно выбора сети**

Это приведет Вас к следующему экрану (рис. 2.9):



**Рис. 2.9. Окно ввода имени рабочей группы**

Здесь Вы должны установить название рабочей группы. Эти действия Вы должны повторить на всех компьютерах Вашей сети.

Некоторые провайдеры используют свою собственную рабочую группу. В этом случае они должны Вас проинструктировать об использовании имени рабочей группы.

*IP адресация.* Прежде всего, Вы должны идентифицировать каждый компьютер в сети. Для этого служит так называемая IP (Internet Protocol) адресация. IP адрес – это уникальный номер Вашего компьютера в Вашей сети. IP адрес может быть “статическим” или “динамическим”. В своей внутренней сети Вы можете использовать IP адреса класса С, т.е. в диапазоне 192.168.0.1 до 192.168.0.254. Другими словами, в одной рабочей группе может работать до 254 компьютеров. Обычно серверу назначают адрес 192.168.0.1. Когда Вы активизируете совместный доступ к сети (Internet Connection Sharing), по умолчанию Вашему серверу будет автоматически назначен этот адрес.

*Конфигурация сервера.* Под XP и Windows NT для установки сетевых параметров Вы должны войти с правами администратора.

Установив на все компьютеры сетевые карты, Вы должны назначить каждому компьютеру IP адрес. Для этого нажмите Start → Settings → Network *Connections (Сетевые подключения)*. Теперь кликните правой кнопкой мышки на “Local Area Connection”(«Подключение по локальной сети») и выберите меню Properties (*Свойства*), рис. 2.10.

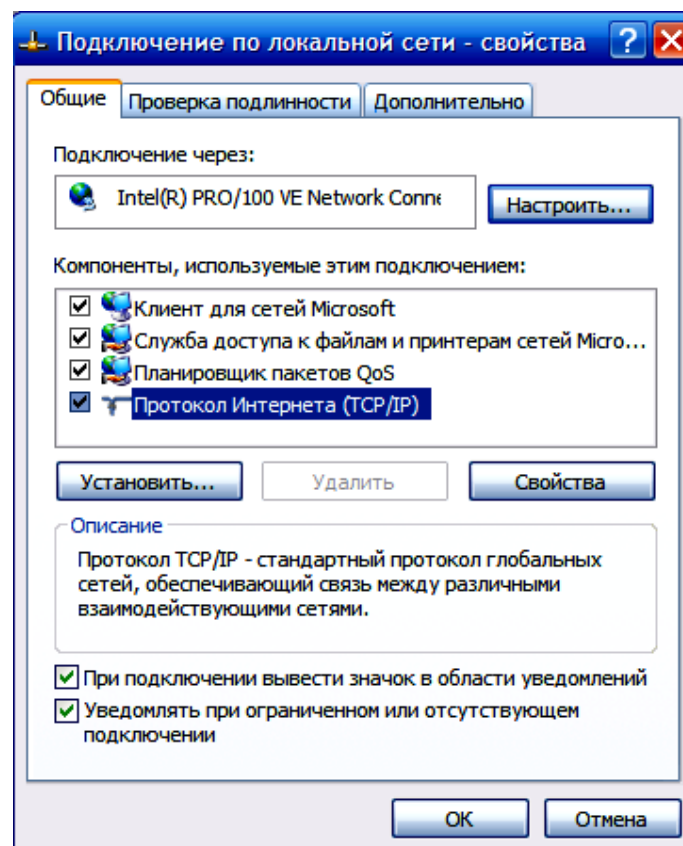


Рис. 2.10. Окно свойств подключения по локальной сети

Затем укажите на протокол TCP/IP и нажмите Properties (*Свойства*). Перед Вами откроется окно, позволяющее установить все необходимые сетевые параметры (рис. 2.11).

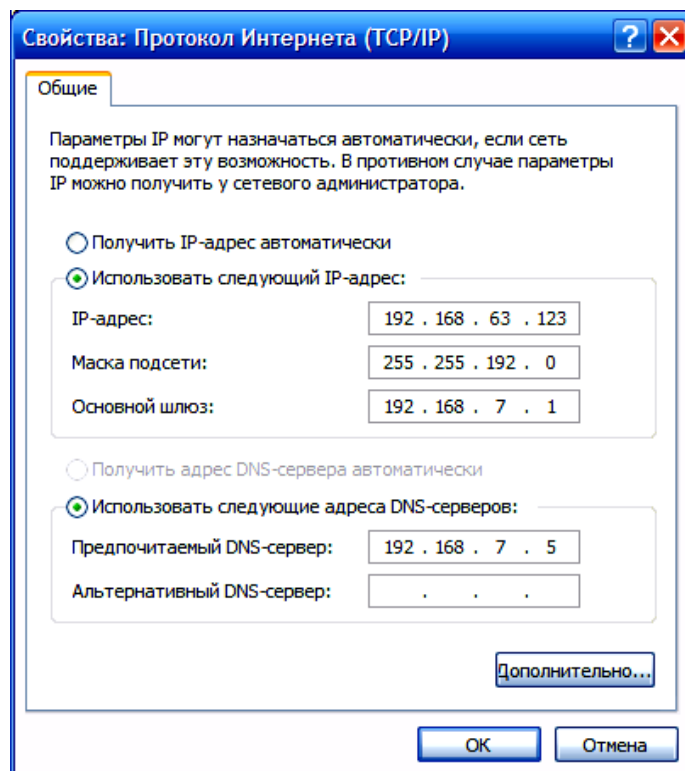


Рис. 2.11. Окно установки сетевых параметров

*Конфигурирование клиентских машин.* Например, одному из клиентских компьютеров установлен IP адрес 192.168.0.5. Маска подсети устанавливается автоматически, по этому поводу Вам волноваться не стоит. Обратите внимание, что IP адрес должен быть уникальным, и не может повторяться внутри одной сети. В случае повторного использования IP адреса, Windows сообщит о возникновении проблемы с повторяющимся IP адресом.

#### 2.4.5. Совместное использование Интернет

Современная сеть должна решать три основные задачи: совместный доступ к Интернету, совместное использование принтера и совместное использование файлов и папок. В принципе, все три задачи уже, так или иначе, решены в некоторых предыдущих версиях Windows. Однако в Windows XP они решены лучше и проще. Итак, в первую очередь давайте рассмотрим реализацию совместного доступа в Интернет. В нашем примере настройки клиентского компьютера мы устанавливаем обращение к серверу с IP адресом 192.168.0.1. Этот адрес указан как адрес шлюза, т.е. компьютера, через который все остальные будут обращаться в Интернет. Тот же адрес мы указываем в качестве первичного DNS (DNS – это сервис, который позволяет по символическому имени узла определить его физический IP адрес). Теперь, нажмите ОК, что вернет Вас в первоначальное меню свойств сетевого соединения. Здесь откройте закладку **Advanced (Дополнительно)**, и проверьте опцию **Internet Connection Firewall (Брандмауэр Windows)**, рис. 2.12.

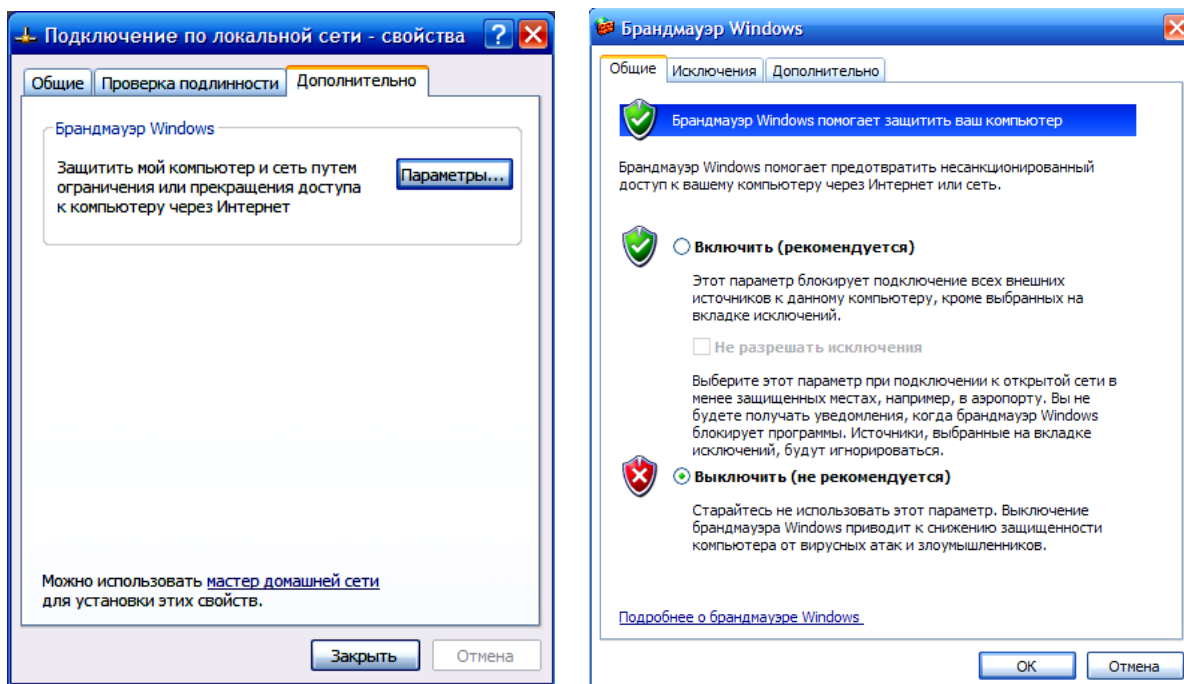


Рис. 2.12. Окна настройки опции Брандмауэр Windows

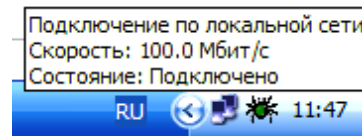
На клиентских машинах эта опция должна быть отключена.

После завершения настройки всех клиентских машин Вам необходимо активизировать доступ в Интернет на серверной машине. Для этого нужно войти в меню сетевых соединений (**Start** → **Settings** → **Network Connections**) (*Пуск* → *Панель управления* → *Сетевые подключения*), выбрать иконку, через которую подключаетесь к провайдеру (по модему или быстрому каналу). В меню свойств учетной записи необходимо выбрать закладку *Advanced* (*Дополнительно*), где Вы найдете все опции, необходимые для организации совместного доступа.

В первую очередь обратите внимание на включение Internet Connection Firewall для Internet соединения. Firewall – это система защиты, которая работает как защитный щит между внутренней сетью и внешним миром. Internet Connection Firewall (ICF) – это программный продукт, который используется для установки ограничений передачи информации от и к вашей рабочей группе. Мы рекомендуем разрешить эту опцию. В принципе, Windows XP не требует перезагрузки при изменении сетевых настроек, однако настоятельно рекомендуется это сделать не только на сервере, но и на всех клиентских компьютерах. Теперь, Вы можете попробовать войти в Интернет с любого компьютера в Вашей сети, причем в случае с Windows XP Вы можете спокойно, без необходимости делать дополнительные настройки, использовать программы, подобные ICQ.

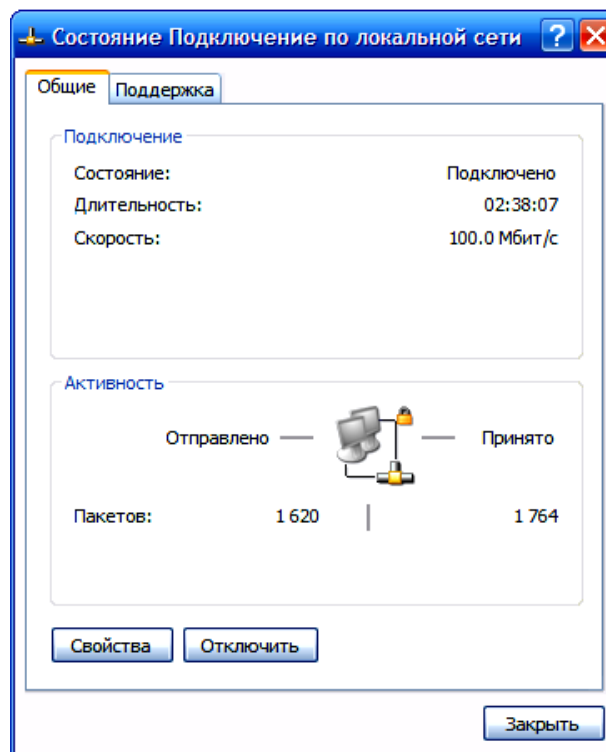
В случае, если доступа к сети нет, попробуйте сначала проверить правильность подключения всех кабелей и включение концентратора или свича. Если все сделано правильно, то при подключении сетевого кабеля к Вашей сетевой карте на экране в *tray* области (рядом с часами), появится сообщение о подключении сетевого кабеля на скорости 100 или 10Mb), рис. 2.13.

**Рис. 2.13. Сообщение о подключении сетевого кабеля**



В случае, если этого не происходит, необходимо проверить включение опции отображения индикатора в сетевых настройках и проверить правильность и целостность кабеля. Далее попробуйте в окне DOS набрать команду *ping 192.168.0.1*, которая проверит соединение с серверной машиной. Если пинг не прошел, необходимо проверить работоспособность и настройку сетевых карт и свича. Кроме того, одной из причин имеющихся проблем может быть установка Windows XP поверх другой операционной системы, имеющей настроенную сеть, либо компьютер использовал собственное подключение к Интернету.

Теперь необходимо коснуться вопроса контроля состояние подключения к Интернету. Особенно это касается случая использования операционных систем, отличных от Windows XP. Для доступа к этой опции нажмите Start → Network Connections (*Пуск → Панель управления → Сетевые подключения*), рис. 2.14.



**Рис. 2.14. Окно контроля состояния подключения к сети**

Как Вы можете видеть, Windows XP достаточно подробно показывает параметры соединения, такие как скорость соединения и трафик на Вашей клиентской машине. Кроме того, здесь Вы можете отключить Интернет-соединение.



### 2.4.6. Совместное использование принтера

Теперь, когда Вы установили ICS, пришло время настроить принтер для совместного использования в рабочей группе. Эта возможность очень полезна для небольших компаний или отделов. Установив совместный доступ к одному или нескольким принтерам, Вы сможете значительно сэкономить не только на принтерах, но и на времени. Если на Вашем сервере еще не установлен принтер, сделать это можно с помощью меню Printers and Faxes, открыть которое можно так: Start → Settings → Printers and Faxes (**Пуск** → **Панель управления** → **Принтеры и факсы**). Как только принтер будет установлен, кликните по его иконке правой клавишей мышки и выберите пункт Sharing. В открывшемся окне просто нажмите «Share this printer» и наберите имя, под которым принтер будет виден в сети.

### 2.4.7. Совместное использование и доступ к файлам

Совместное использование файлов и папок позволит создавать на предприятии библиотеку документов, шаблонов и т.д., открывать и ограничивать доступ к документам различных сотрудников. На самом деле для этого необходимо использовать специальные продукты, способные индексировать и архивировать документы, осуществлять быстрый поиск и т.д. Однако Windows XP частично позволяет решить эти задачи и сэкономить на приобретении программ третьих разработчиков.

Для того, чтобы разрешить сетевой доступ к файлам и папкам, нужно в My Computer выбрать диск, к которому нужно открыть доступ, и правой кнопкой войти в свойства диска, где выберите закладку Sharing (**Доступ**), как показано на рис. 2.15.

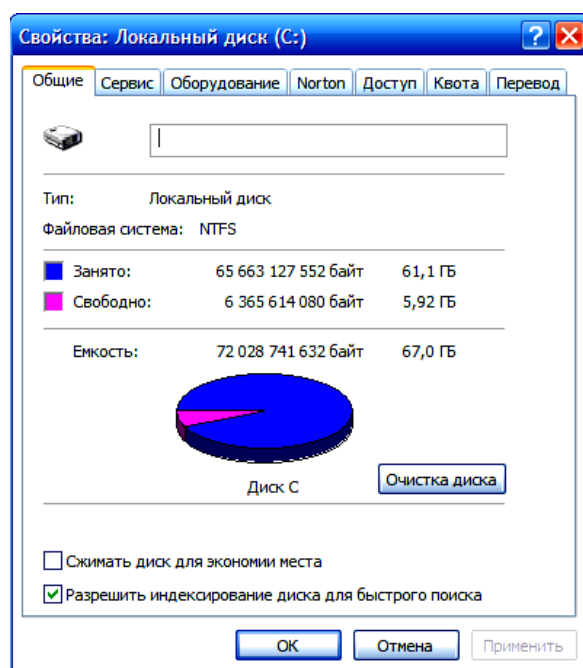


Рис. 2.15. Окно свойств диска

В целом здесь все понятно, однако необходимо обратить внимание на то, что в случае включения опции «Allow network users to change my files» Вы даете другим пользователям полный доступ к своим файлам, т.е. не только редактирование, но и удаление. В случае совместной работы с Windows 2000 и NT с точки зрения защиты Вы можете отключить простой режим совместного использования файлов через меню View → Folder Options → [X]Use Simple File Sharing.

#### 2.4.8. Добавление сетевых дисков

Для упрощения доступа к часто используемым сетевым дискам можно добавить их в свое окно My Computer и использовать как обычный диск на Вашем компьютере. Для этого Вам необходимо правой кнопкой мышки кликнуть на сетевое окружение, и в открывшемся контекстном меню выбрать «Map Network Drive...» («Подключить сетевой диск»).

Здесь выберите букву для нового диска, и укажите путь к нему. Если Вы не знаете точного пути, то можете воспользоваться кнопкой «Browse...» («Обзор»), рис. 2.16.

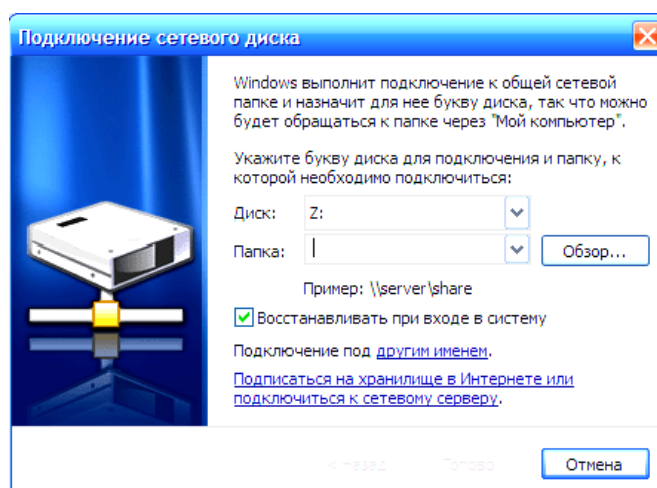


Рис. 2.16. Окно подключения сетевого диска

## 2.5. Безопасность информационных систем

Безопасность информационной системы – свойство, заключающееся в способности системы обеспечить конфиденциальность и целостность информации, т. е. защиту информации от несанкционированного доступа с целью ее раскрытия, изменения или разрушения.

Информационную безопасность часто называют среди основных информационных проблем XXI в. Действительно, вопросы хищения информации, ее сознательного искажения и уничтожения часто приводят к трагическим для пострадавшей стороны последствиям, ведущим к разорению и банкротству фирм.

Все угрозы информационным системам можно объединить в обобщающие их три группы:

- угроза раскрытия – возможность того, что информация станет известна тому, кому не следовало бы ее знать;
- угроза целостности – умышленное несанкционированное изменение (модификация или удаление) данных, хранящихся в вычислительной системе или передаваемых из одной системы в другую;
- угроза отказа в обслуживании – возможность появления блокировки доступа к некоторому ресурсу вычислительной системы.

Средства обеспечения информационной безопасности в зависимости от способа их реализации можно разделить на следующие классы методов:

- организационные методы – имеют в виду рациональное конфигурирование, организацию и администрирование системы. В первую очередь это касается сетевых информационных систем, их операционных систем, полномочий сетевого администратора, набора обязательных инструкций, определяющих порядок доступа и работы в сети пользователей;
- технологические методы, включающие в себя технологии выполнения сетевого администрирования, мониторинга и аудита безопасности информационных ресурсов, ведения электронных журналов регистрации пользователей, фильтрации и антивирусной обработки поступающей информации;
- аппаратные методы, реализующие физическую защиту системы от несанкционированного доступа, аппаратные функции идентификации периферийных терминалов системы и пользователей, режимы подключения сетевых компонент и др.;
- программные методы – это самые распространенные методы защиты информации (например, программы идентификации пользователей, парольной защиты и проверки полномочий, брандмауэры, криптопротоколы и др.).

Без использования программной составляющей практически невыполнимы никакие, в том числе и первые три группы методов (т. е. в чистом виде организационные, технологические и аппаратные методы защиты, как правило, реализованы быть не могут – все они содержат программную компоненту).

При этом следует иметь в виду, вопреки иному распространенному мнению, что стоимость реализации многих программных системных решений по защите информации существенно превосходит по затратам аппаратные, технологические и тем более организационные решения (конечно, если использовать лицензионные, а не «пиратские» программы). Наибольшее внимание со стороны разработчиков и потребителей в настоящее время вызывают следующие направления защиты информации и соответствующие им программно – технические средства:

- защита от несанкционированного доступа информационных ресурсов автономно работающих и сетевых компьютеров. Наиболее остро эта проблема стоит для серверов и пользователей сети Интернет. Эта функция реализуется многочисленными программными, программно-аппаратными и аппаратными средствами;

- защита секретной, конфиденциальной и личной информации от чтения посторонними лицами и целенаправленного ее искажения. Эта функция обеспечивается как средствами защиты от несанкционированного доступа, так и с помощью криптографических средств, традиционно выделяемых в отдельный класс;

- защита информационных систем от многочисленных компьютерных вирусов, способных не только разрушить информацию, но иногда и повредить технические компоненты системы: Flash - BIOS, винчестеры и др.

Активно развиваются также средства защиты от утечки информации по цепям питания, каналам электромагнитного излучения компьютера или монитора (применяется экранирование помещений, использование генераторов шумовых излучений, специальный подбор мониторов, комплектующих компьютера, обладающих наименьшим излучением), средства защиты от электронных «жучков», устанавливаемых непосредственно в комплектующие компьютера, и т.д.

### *Контрольные вопросы*

1. Из чего состоит программное обеспечение компьютера?
2. Что такое операционная система? Какие требования предъявляют к операционным системам?
3. Что такое компьютерная сеть?
4. Какие типы сетей Вы знаете?
5. Что такое администрирование сети?
6. Для чего в компьютерных сетях используются протоколы?
7. Какие протоколы физического уровня Вы знаете?
8. Какие протоколы определяют логическую структуру сообщений?
9. Назовите протоколы, применяемые в Internet?
10. Как произвести выбор сетевой карты?
11. Что такое хаб?
12. Как установить сетевые параметры?
13. Для чего служит IP- адресация?
14. Какие задачи должна осуществлять современная сеть?
15. Что Вы знаете о безопасности информационных систем?

### **3. ТЕКСТОВАЯ ИНФОРМАЦИЯ, ВЫЧИСЛЕНИЯ И ДЕЛОВАЯ ГРАФИКА**

---

#### **3.1. Программное обеспечение и технологии обработки информации**

Для выполнения основных задач компьютерной обработки данных в современных офисах целесообразно использовать не отдельные программы, а интегрированные пакеты офисного обслуживания, так как в них реализуется не просто объединение больших автономных программ в пакеты, а их интеграция в прикладные программные комплексы, означающая их полную унификацию. Программы в них имеют общий пользовательский интерфейс и единообразные подходы к решению типовых задач по управлению файлами, форматированию, печати, работе с электронной почтой и т.д.

##### ***3.1.1. Офисный программный пакет Microsoft Office 2000***

В настоящее время на рынке прикладных офисных программных продуктов доминируют пакеты фирмы Microsoft Office 2000 и 2003. Последние версии этих офисных комплексов содержат средства коллективной работы, более тесной *интеграции компонентов*, а также *средства взаимодействия с Интернетом*.

Microsoft (MS) Office 2000 может работать под управлением операционных систем Windows 95, Windows 98, Windows NT, Windows 2000. Необходимая для эффективной работы конфигурация компьютера зависит от используемой операционной системы, но минимально достаточный набор включает:

- микропроцессор Pentium любого типа;
- оперативную память емкостью не менее 32 Мбайт;
- объем свободного дискового пространства 300 Мбайт;
- видеоадаптер с памятью не менее 4 Мбайт.

Комплект Microsoft Office 2000 поставляется в нескольких вариантах, включающих разное количество основных приложений:

- Office 2000 Standard – предназначен для пользователей, которым нужен базовый набор средств для делопроизводства, расчетов и анализа информации, а также для создания и публикации документов в сети Интернет;
- Office 2000 Professional – дополнительно обеспечивает средства работы с большими базами данных в настольных издательских системах, а также для управления малым бизнесом;
- Office 2000 Premium – самый полный универсальный вариант поставки;
- Office 2000 Small Business – ориентирован в основном на решение задач малого бизнеса.

### ***3.1.2. Семейство приложений обработки информации Microsoft Office System***

С выпуском новой версии **Microsoft Office 2003** компания Microsoft не только представила обновленный вариант наиболее популярного комплекта инструментов для обработки деловой информации, но и заложила основу целого семейства приложений обработки информации, получившего название **Microsoft Office System**. В состав этого семейства помимо приложений, доступных в разных вариантах поставки MS Office, вошли и другие популярные продукты, серверы и службы, разработанные для совместной работы по решению широкого спектра бизнес-задач. Сегодня Microsoft Office System включает следующие продукты:

**Microsoft Office 2003.** Пакет Microsoft Office 2003, преемник Microsoft Office XP, является ключевым компонентом семейства Microsoft Office System. Новые версии популярных продуктов существенно расширяют возможности обработки корпоративных данных, предлагая более быстрые и эффективные инструменты редактирования, хранения и представления информации и организации совместной работы сотрудников с данными. В различные выпуски Office 2003 могут входить следующие приложения системы Microsoft Office System:

- **Microsoft Office Outlook 2003.** Персональный менеджер Microsoft Outlook 2003 выполняет функции планировщика рабочего времени и одновременно коммуникационного центра системы Microsoft Office System, управляя электронной почтой, календарями, контактами и другими личными сведениями и данными группы. Новые возможности Outlook 2003 – кэширование, автоматическая группировка сообщений, функции борьбы со спамом и улучшенные средства просмотра сообщений - повышают эффективность работы пользователей. В Microsoft Outlook 2003 изменен способ организации сообщений, а также то, как почта может быть прочитана, расположена по приоритетам, зарегистрирована и прочее. Например, пользователи теперь могут регистрировать одно и то же сообщение в нескольких папках, в противовес сохранению сообщения только в одной папке. Система мобильной электронной почты использует кэшируемую почтовую систему, за счет чего сообщения доступны более широкому кругу сетевых и информационных соединений.
- **Microsoft Office Word 2003.** Текстовый процессор Microsoft Word 2003 давно стал стандартом де-факто в создании и редактировании документов. Новые функции Word 2003 направлены на улучшение возможностей коллективной работы с информацией – обновлены механизмы рецензирования и разметки, включая методы отслеживания исправлений и управления примечаниями; включены возможности создания, редактирования и сохранения документов в формате XML.
- **Microsoft Office Excel 2003.** Среди улучшений Excel 2003 – ряд новых функций анализа и статистической обработки данных, дополнительные возможности коллективной обработки документов, более широкая поддержка формата XML, интеграция с Microsoft Windows SharePoint Server, расширенные возможности смарт-тегов.

- **Microsoft Office PowerPoint 2003.** В новой версии системы подготовки презентаций PowerPoint 2003 улучшен интерфейс, введена поддержка смарт-тегов в PowerPoint 2003, расширены возможности работы с мультимедиа – теперь при показе слайдов можно воспроизводить потоковое аудио и видео. Добавлена опция записи презентаций на компакт-диск.
- **Microsoft Producer for PowerPoint 2003.** Бесплатное расширение к PowerPoint 2003 позволяет добавлять аудио- и видеоконтент к презентациям PowerPoint, поддерживает технологию Windows Media 9 Series, а также платформы Mac и Netscape. Продукт предлагает расширенную возможность захвата и импорта презентаций непосредственно в Producer 2003 и включает новый мастер Presentation Wizard. Как отмечают в Microsoft, Producer 2003 является мощным приложением к PowerPoint 2003, позволяющим пользователям выпускать презентации для широкой аудитории. Получить более подробную информацию по продукту, ознакомиться с демо-версией и загрузить саму программу можно с сайта компании [www.microsoft.com/office/powerpoint/producer/](http://www.microsoft.com/office/powerpoint/producer/).
- **Microsoft Office Access 2003.** Новая версия базы данных Access 2003 включает улучшенный режим импорта и экспорта файлов в формате XML, а также возможность непосредственной работы с XML файлами. Среди других нововведений - режим обнаружения часто встречающихся ошибок (помечаемых определенным образом) с вариантами их исправления, а также новая функция определения зависимости объектов.
- **Microsoft Office FrontPage 2003.** HTML редактор FrontPage 2003 обеспечивает быструю разработку интерактивных Web-узлов и генерацию кода. Новая версия поддерживает формат XML и подключение данных из внешних источников.
- **Microsoft Office InfoPath 2003.** Новый продукт InfoPath 2003 предназначен для сбора и управления данными, позволяя создавать динамические формы, заполняемые пользователем, и управлять движением заполненных форм при создании комплексных систем обработки информации на платформе XML. InfoPath 2003 обеспечивает высокоэффективный и гибкий способ сбора данных с возможностью совместного использования этих данных сотрудниками организации.
- **Microsoft Office Publisher 2003.** Система подготовки публикаций Publisher 2003 позволяет пользователям любого уровня подготовки быстро создавать и публиковать информационные материалы профессионального качества. Общий интерфейс с другими приложениями Microsoft Office System облегчает изучение и работу с продуктом.
- **Microsoft Office OneNote 2003.** Редактор заметок OneNote 2003 - новый продукт семейства Microsoft Office System, предлагающий возможность записи, хранения и многократного использования данных различной формы, включая вводимые с помощью клавиатуры и рукописные заметки, нарисованные от руки схемы, звукозаписи, фотографии и рисунки из Internet, а также данные из других приложений. Microsoft OneNote разработан для заполнения промежутка между записями данных на бумаге и вводом текста в ПК и позволяет пользователям делать заметки более естественным путем, чем с помощью текстового процессора. Пользователи могут печатать в любой части экрана и помечать свои записки. Таблицы помогают организовать записные книжки, в то время как индексная панель облегчает в них поиск сделанных заметок. Файлы постоянно сохраняются и могут быть опубликованы как html, например, в Weblogs. Заметки могут сопровождаться цифровой записью голоса.

- **Microsoft Office Visio 2003.** Специализированный продукт Visio 2003 предоставляет широкие возможности для работы с экономическими и техническими диаграммами. Усовершенствованные функции Visio 2003 помогают более четко отображать сложные иерархические взаимосвязи; предлагают больше инструментов наглядного представления сложных бизнес-процессов; реализуют новые механизмы связывания данных.
- **Microsoft Office Project 2003.** Семейство продуктов Microsoft Office Project 2003 включает Project Standard 2003, Project Professional 2003, Project Server 2003 и Project Web Access. Используемые в сочетании, эти продукты обеспечивают эффективное управление корпоративными проектами и позволяют эффективно планировать и управлять распределением ресурсов, обеспечивая достижение заданных целей в намеченные сроки.

Минимальные системные требования к компьютеру при использовании продуктов Microsoft Office 2003:

- Pentium 233 МГц и выше, Pentium III рекомендуется;
- Microsoft Windows 2000 Service Pack 3 и более поздним или Microsoft Windows XP;
- 128 МБ оперативной памяти;
- 400 МБ дискового пространства.

### *3.1.3. Перспективы развития Microsoft Office*

В июне 2005 года Microsoft объявила о направлениях дальнейшего развития семейства продуктов Office. Новый Office (Office 12), построенный на XML, претендует на роль полноценного бизнес-пакета с возможностями совместной работы и бизнес-аналитики и будет отличаться расширенным функционалом и измененным интерфейсом приложений. В продукте появятся возможности совместной работы через Web-портал, Communication Server и решения Groove. Кроме того, в Office 12 будут расширены функции поиска, бизнес-аналитики и управления контентом.

**Изменения форматов данных.** Начало процессу упорядочивания документов было положено в Office 2003, который позволяет определять структуру, расположение и правила работы с документами. Новая версия, как обещают в Microsoft, будет обладать еще более широким функционалом в этой области. Форматы следующей версии будут открытыми и бесплатными. Изменению подвергнутся файлы Word, Excel и PowerPoint. Их структура будет полностью документирована. Документы Word будут сохраняться в новом расширении — .DOX. При этом вместо одного файла появляется структура папок, в которых стили, картинки, таблицы и др. информация хранятся отдельно друг от друга. Теперь, чтобы удалить картинку, достаточно удалить соответствующий файл.

Новые документы будут сжиматься встроенным алгоритмом архивации ZIP, что позволит сэкономить 50% места и даст возможность архива-



торам распознавать содержимое. Также документы будут обладать модульной структурой, что повысит эффективность работы системы автоматического восстановления, которая в случае необходимости отделит поврежденные части для последующего «ремонта». Модульность позволит локализовать мета-данные и макросы, и «лечить» их от вирусов, не вмешиваясь в другое содержимое документа. Microsoft надеется на создание специальных утилит для проверки XML.

За счет использования XML увеличится интероперабельность (взаимная совместимость с другими приложениями). Ранее несовместимые СУБД и приложения смогут работать с содержанием документов и электронными таблицами, получая оттуда данные и внося изменения. Эта возможность была названа Биллом Гейтсом «фундаментальной основой ИТ», когда он описывал преимущества XML на саммите Microsoft для исполнительных директоров. Интероперабельность будет распространяться на любое ПО и системы, использующие XML – например, СУБД Oracle SQL Plus и MySQL. Те, кто не захочет пользоваться новым форматом, смогут настроить Office 12 на сохранение документов в старых форматах. При желании старые документы можно будет преобразовать в новые при помощи опции массовой конвертации.

**Поддержка PDF.** В Office 12 появится возможность просматривать PDF-файлы, наподобие функционала программы Acrobat Reader от Adobe, и сохранять документы в формате PDF в режиме «только для чтения». Решение было принято по многочисленным просьбам пользователей.

Кроме того, Microsoft планирует включить в Office 12 поддержку документов формата XPS (бумажная спецификация XML), конкурирующего аналога Adobe PDF. В отличие от документов DOC и подобных форматов, XPS и PDF не требуют дополнительных компонентов системы, в частности, шрифтов, и отображаются на всех компьютерах одинаково. В формат PDF также включена поддержка защиты авторских прав, например, ограничение срока отображения.

XPS будет поддерживаться ОС Windows Vista, в систему встроит программу чтения XPS, которая необходима для пользования документами этого формата. По утверждению Microsoft, XPS предоставит лучшее качество печати и большее соответствие цветов экранного и распечатанного изображений. Несмотря на конкуренцию форматов, PDF также будет поддерживаться в Office 12.

**Изменения интерфейса и функционала.** У Microsoft Word, PowerPoint, Excel и Access полностью поменяется интерфейс. Теперь результат многих функций форматирования будет демонстрироваться при наведении курсора на кнопку, а не при ее нажатии. Пропадут традиционные выпадающие меню. Вместо них появятся закладки с командами для стандартных операций над документами. Команды на закладках будут отображены в виде иконок и текста.

Сейчас в Office насчитывается около 1,5 тыс. команд (в момент появления было 100). Поэтому проблема «захламления» рабочего пространства пользователя выходит на первое место. В новой версии Office все команды будет разделены на 2 логических области. Наверху экрана – главные инструменты для работы с документами, ниже – вспомогательные, например, для перехода в другие режимы, управления окнами и т.д. В подсказках к инструментам есть и названия, и картинки результатов.

Для удобства работы будут предложены так называемые галереи – наборы готовых шаблонов, отображаемых при наведении курсора, например, шаблонов цвета, шрифта и ширины таблиц. Благодаря этому неопытные пользователи смогут избежать использования пугающих их диалоговых окон. Для продвинутых пользователей останется возможность форматирования через несколько команд. Шаблоны можно применять не только к целому документу, но и к его части.

В новом Excel число строк увеличится с 65 тыс. до более 1 млн, а столбцов – до 16 тыс. В режиме просмотра страниц Excel перед печатью появится возможность вносить правку, работая как в обычном режиме. Упростится создание списков и сводных таблиц. Усовершенствуются инструменты диаграмм – их станет больше, как и цветов диаграмм. Появится инструмент автозапоминания формул. Теперь окрашенные ячейки с максимальными и минимальными числами в массивах данных смогут иметь соответственно более и менее интенсивный цвет.

В Outlook улучшится поиск писем и их содержания. В календаре появится список задач, причем невыполненные задачи программа автоматически перенесет на следующий день. Появится специальная панель, отражающая расписание выполнения поставленных задач. Пользователь сможет использовать сразу несколько календарей, а также делиться своим календарем задач с партнерами и коллегами, отправив его в теле письма или приложении. Изменения в списке задач Outlook теперь можно будет видеть на общем портале через SharePoint.

В PowerPoint появится новый динамический инструмент создания диаграмм и новые их форматы. При работе PowerPoint в связке с SharePoint можно в режиме командной работы создавать библиотеки слайдов.

Первая бета-версия Office 12 поступила пользователям в середине ноября 2005 г.

## **3.2. Обработка текстовой информации**

### ***3.2.1. Основные функции текстовых редакторов***

Одной из основных функций перечисленных выше программ – работа с текстами. Любой из вас согласится с тем, как много сил и времени затрачивается на многочисленные записи, которые приходится делать в разных ситуациях. При ручной записи или даже при использовании пишущей машинки часто неприятной проблемой становится необходимость

исправлять ошибки или вносить какие-то изменения в текст. При этом приходится зачеркивать, стирать, заклеивать, что портит вид текста. Необходимость переписывать текст ведет к лишнему расходу бумаги.

Имея компьютер, можно создавать тексты, не тратя на это бумаги. Носителем текста становится память ЭВМ. Конечно, для длительного его сохранения это должна быть внешняя память – магнитные или оптические диски. С помощью компьютера можно создавать текстовые документы и хранить их на носителях внешней памяти в виде файлов.

Преимущества файлового хранения текстов: экономия бумаги; компактное размещение; возможность многократного использования магнитного носителя для хранения разных документов; возможность быстрого копирования на другие магнитные носители; возможность передачи текста по линиям компьютерной связи.

Для хранения текста на диске очень часто используется специальный формат файла, который называется текстовым файлом.

Текстовый файл – простейшая форма хранения текстовой информации. Его содержимое может выводиться на печать или на экран только в единообразном шрифте. Но с помощью компьютера возможно создание текстовых документов, в которых используются разнообразные шрифты, нестандартные символы (например, математические знаки), рисуются таблицы, схемы. Организация таких файлов более сложная, чем текстовых.

Для работы с разнообразными по содержанию и форме текстовыми документами существуют прикладные программы, которые называются текстовыми редакторами. *Текстовый редактор* – это прикладная программа, позволяющая создавать текстовые документы, редактировать их, просматривать содержимое документа на экране, распечатывать документ, изменять формат документа. По отношению к развитым текстовым редакторам с широкими возможностями по форматированию текста, включению графики, проверке правописания часто применяется название «*текстовый процессор*».

Существует множество разнообразных текстовых редакторов – от простейших учебных до мощных издательских систем, с помощью которых делают книги, газеты, журналы.

Для любого текстового редактора характерно присутствие на экране *меню команд* управления редактором. Это команды изменения режимов работы, файловых операций, печати, форматирования текста, обращения за справкой и другие. Меню может иметь как текстовую, так и пиктографическую форму.

Характерные режимы работы различных текстовых редакторов:

- ввод-редактирование;
- поиск по контексту и замена;
- орфографический контроль;
- работа с файлами;
- печать;
- помощь.

*Ввод-редактирование* – это основной режим работы текстового редактора. В текстовом документе, созданном на компьютере с помощью текстового редактора, могут использоваться разнообразные шрифты. Современные текстовые редакторы имеют наборы шрифтов. У каждого шрифта есть свое название. Например: Arial, Times New Roman, Serif и др. Буквы одного шрифта могут иметь разные начертания. Различаются обычное (прямое) начертание, курсив, полужирное начертание. Кроме того, предоставляется возможность подчеркивания текста.

Текстовые редакторы, работающие в среде Windows, позволяют управлять размером символов.

Практически все редакторы, распространенные в нашей стране, позволяют использовать как русский, так и английский алфавиты.

*Форматирование текста.* Под форматом печатного текста понимается расположение строк (длина строки, межстрочное расстояние, выравнивание текста по краю или по середине строки); размеры полей, страниц. Параметры формата (размер строки, межстрочное расстояние) устанавливаются пользователем перед вводом текста и в дальнейшем автоматически выдерживаются текстовым редактором. Пользователю остается только набирать текст. Некоторые текстовые редакторы производят автоматический перенос слов, соблюдая правила переноса. Если вы ввели текст в определенном формате, а потом решили изменить формат, то с помощью текстового редактора это легко сделать. Достаточно установить новые параметры формата и отдать команду «Переформатировать текст» (весь текст, или абзац, или выделенный фрагмент текста). Большинство текстовых редакторов позволяют выделять в тексте куски, которые называют фрагментами или блоками. Чаще всего блок на экране отмечается изменением цвета фона и символов. С выделенным блоком могут быть выполнены следующие действия: переформатирование, изменение шрифта, удаление, перенос, копирование. Три последние операции связаны с использованием специальной области памяти, которую называют буфером для копирования.

При работе с текстом, когда необходимо разместить различные перечни удобно использовать команду *Списки*. Существуют списки различных типов: нумерованные, маркированные и др.

Возможно в текстовом редакторе создание таблиц.

Работая с текстовым редактором, пользователь должен иметь возможность выполнять основные файловые операции:

- создать новый файл;
- сохранить текст в файле;
- открыть файл (загрузить текст из файла в оперативную память).

Тексты, создаваемые с помощью текстового редактора, в конечном итоге распечатываются на бумаге. Для этого предусмотрен режим печати.

Одно из главных условий «дружественности» программного обеспечения – наличие помощи пользователю. Это делается в форме подсказки,

справочника, учебника, хранимого во внешней памяти компьютера. Обычно обращение к режиму помощи происходит по команде СПРАВКА, ПОМОЩЬ или «?». Получив справку, пользователь выходит из режима помощи и возвращается к тому этапу работы, который был прерван.

### **3.2.2. Различные форматы текстовых файлов**

Формат файла определяет способ хранения текста в файле. Существуют универсальные форматы текстовых файлов, которые могут быть прочитаны большинством текстовых редакторов, и оригинальные форматы, которые используются отдельными текстовыми редакторами. Для преобразования текстового файла из одного формата в другой используются специальные программы – программы-конверторы. Наиболее распространенные форматы текстовых файлов:

- Text Only (ТХТ). Наиболее универсальный формат. Сохраняет текст без форматирования, в текст вставляются только управляющие символы конца абзаца. Применяют этот формат для хранения документов, которые должны быть прочитаны в приложениях, работающих в различных операционных системах;

- текст в формате RTF. Универсальный формат, который сохраняет все форматирование. Преобразует управляющие коды в команды, которые могут быть прочитаны и интерпретированы многими приложениями. В результате информационный объем файла существенно возрастает;

- документ Word (DOC). Формат, который используют версии редактора Word;

- Works для Windows (WPS). Оригинальный формат интегрированной системы. Works. При преобразовании из формата DOC форматирование сохраняется не полностью;

- HTML – документ (НТМ, HTML, XML). Форматы хранения Web-страниц. Содержит управляющие теги (коды) языка разметки гипертекста;

- Лексикон (LX). Оригинальный формат отечественного текстового редактора Лексикон.

Выбор требуемого формата текстового документа или его преобразование производится в процессе сохранения файла. С помощью диалоговой панели *Сохранение документа* присвоить документу имя и требуемый формат.

### **3.2.3. Гипертекст**

Для отображения в тексте смысловых связей между основными разделами или понятиями используют *гипертекст*. Гипертекст позволяет структурировать документ путем выделения в нем слов-ссылок (гиперссылок). При активизации гиперссылки происходит переход на фрагмент текста, заданный в ссылке.

Гиперссылка состоит из двух частей: указателя ссылки и адресной части ссылки. Указатель ссылки – это объект (фрагмент текста или рисунок), который визуально выделяется в документе. Адресная часть гиперссылки представляет собой название закладки в документе, на который указывает ссылка. Закладка – это элемент документа, которому присвоено уникальное имя.

Гипертекстовые структуры могут распространяться на документы различных типов.

#### **3.2.4. Распознавание документов**

При переходе предприятий к электронному документообороту и необходимости редактирования различных документов используются системы оптического распознавания символов. С помощью сканера достаточно просто получить изображение текста в графическом файле. Для получения документа в текстовом формате необходимо провести распознавание текста, т.е. преобразовать элементы графического изображения в последовательность текстовых символов.

Сначала необходимо распознать структуру размещения текста на странице: выделить колонки, таблицы, изображения. Далее выделенные текстовые фрагменты графического изображения необходимо преобразовать в текст. Наиболее распространенная система оптического распознавания *FineReader*.

### **3.3. Вычисления и деловая графика**

#### **3.3.1. Электронные таблицы**

В повседневной деятельности инженера приходится решать ряд научных и производственных задач, в том числе и расчетов. Обычно результатами расчетов являются длинные колонки чисел. Обработать большие массивы числовых данных, например результаты экспериментов, статистические данные и т.д., позволяют электронные таблицы. Наибольшее распространение получили таблицы Microsoft Excel.

Электронная таблица – это работающее в диалоговом режиме приложение, хранящее и обрабатывающее данные в прямоугольных таблицах.

Электронная таблица состоит из столбцов и строк. Заголовки столбцов обозначаются буквами или сочетаниями букв (А, В, С, АС), заголовки строк – числами (1, 2, 3 и т.д.). Ячейка – место пересечения столбца и строки. Адрес ячейки состоит из заголовка столбца и заголовка строки (А1, АС3). Ячейка, с которой производятся какие-либо действия, выделяется рамкой и называется активной.

Электронные таблицы, с которыми работает пользователь, называются рабочими листами. Документы электронных таблиц могут включать несколько рабочих листов и называются рабочими книгами.

В работе с электронными таблицами можно выделить три основных типа данных: число, текст, формула. В зависимости от решаемой задачи возникает необходимость применения различных форматов представления данных. В каждом конкретном случае можно выбрать наиболее подходящий формат.

В формулах используются ссылки на адреса ячеек. Существуют два основных типа ссылок: относительные и абсолютные.

Относительные ссылки в формулах используются для указания адреса ячейки, вычисляемого относительно ячейки, в которой находится формула. При перемещении или копировании формулы из активной ячейки относительные ссылки автоматически обновляются в зависимости от нового положения формулы. Относительные ссылки имеют следующий вид: A1, P3. При копировании формулы, содержащей только относительные ссылки, из ячейки в ячейку, обозначения в формуле меняются.

Абсолютные ссылки в формулах используются для указания фиксированного адреса ячейки. При копировании или перемещении формулы, абсолютные ссылки не изменяются. В абсолютных ссылках перед неизменяемым значением адреса ячейки ставится знак доллара (\$A\$1).

Для производства вычислений используют формат формул. Формула должна начинаться со знака равенства и может включать числа, имена ячеек, функции и знаки математических операций. При вводе формулы в ячейке отображается результат вычислений. При изменении исходных данных, результат пересчитывается.

Таблицы Microsoft Excel предоставляют большую библиотеку встроенных функций следующих категорий:

- математические;
- статистические;
- финансовые;
- текстовые;
- логические;
- дата и время;
- ссылки и массивы;
- инженерные, и др.

Решение математической задачи в электронных таблицах – это способ получения результата за меньшее время, чем при обычном аналитическом решении, причем возможно решение задач следующих разделов математики: элементарная математика, высшая математика, теория вероятности и математическая статистика, численные методы. Excel предоставляет обширную библиотеку встроенных функций, которые можно найти на панели меню под названием «*Мастер функций*». После этого в диалоговом окне необходимо выбрать категорию, затем необходимую функцию (рис.3.1).

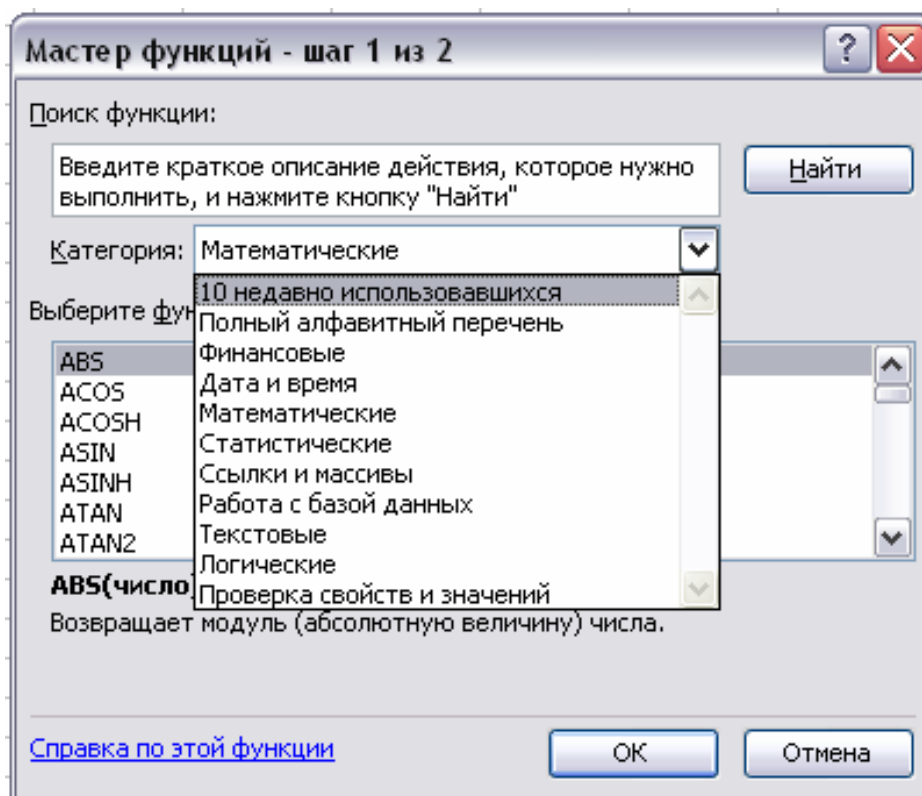


Рис. 3.1. Диалоговое окно выбора функции

Вычислить значение заданной функции можно непосредственно в ячейке. Ввод в ячейку аналитической заданной функции начинается со знака равно =. После него вводится аналитическое выражение заданной функции, в котором переменные заменяются именами каких-либо ячеек, отличными от ячейки ввода, в которых находятся значения переменных. Нажатие на кнопку «Enter» завершает ввод функции. Автоматически в ячейке ввода появляется результат вычисления.

Вот один из примеров применения встроенных математических функций для решения задач горного производства (рис. 3.2.):

Вычислить относительную метанообильность шахты

	A	B	C	D	E	F
1	Исходные данные					
2	Месяцы работы в году	1	2	3	4	5
3	Расход газа, <i>l</i> , м. куб./мин	6.5	6.8	7.1	7.6	7.2
4	Число отработанных дней, <i>N</i>	26	26	25	26	26
5	Добыча, <i>A</i> , тыс.т.	57.2	54.3	61	52.9	41
6						
7	Результат: Относительная метанообильность шахты					<b>4.91</b>

Рис. 3.2. Пример использования электронной таблицы при производственных расчетах



Электронные таблицы позволяют осуществлять сортировку данных, т.е. производить их упорядочение. Данные можно сортировать по возрастанию и убыванию. Также, при помощи фильтров, возможно производить поиск данных в соответствии с заданными условиями.

Электронные таблицы полезны и для научных целей. С их помощью можно строить компьютерные математические модели, проводить вычислительные эксперименты. Математической моделью называется информационная модель объекта или процесса, выраженная математическими средствами (формулами, уравнениями и т. п.).

Табличный процессор может применяться в качестве инструмента для математического моделирования.

Полученную математическую модель можно использовать для проведения вычислительного эксперимента. Вычислительный эксперимент – это математические расчеты с целью прогноза поведения какой-то системы, с целью выяснения вопроса о том, как изменение одних характеристик системы отражается на других.

При решении определенных задач в Excel нет необходимых встроенных функций. В таких случаях, при знании методов решения той или иной поставленной задачи, это можно сделать без проблем. К примеру, необходимо решить систему линейных алгебраических уравнений.

$$\sum_{j=1}^n a_{ij}x_j = b_i, \quad i = 1, 2, \dots, n, \det A \neq 0.$$

Встроенной функции для решения систем линейных алгебраических уравнений нет. По методу Крамера решение системы имеет вид

$$x_i = \frac{\Delta_i}{\Delta},$$

где  $\Delta = \det A$ ,  $\Delta_i$  – вспомогательный определитель, равный определителю матрицы системы, в которой  $i$ -й столбец заменен столбцом свободных членов.

Найти значения матриц можно при помощи встроенной функции «МОПРЕД».

Вычисление пределов и производных возможно, используя циклические ссылки. Расчеты по циклическим ссылкам проводятся через диалоговое окно, открываемое командами Сервис → Параметры → Вычисления (рис. 3.3).

Напротив параметра «итерации» ставится галочка, напротив параметра «Предельное число итераций» задается число повторений циклов, по умолчанию равное 100.

Например, вычислим первый замечательный предел

$$\lim_{x \rightarrow 0} \frac{\sin x}{x}$$

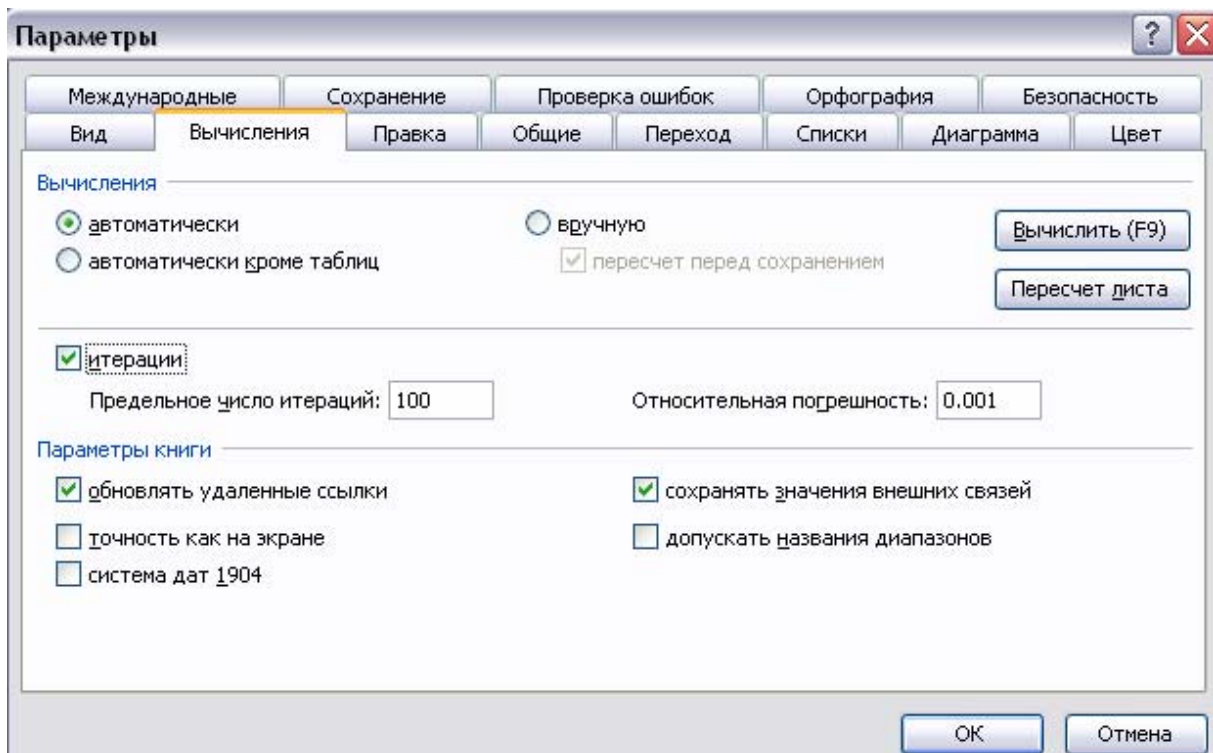


Рис. 3.3. Диалоговое окно (параметры) вычисления

Пример расчета предела показан на рис. 3.4.

В ячейке A1 задаем формулу  $=A1+1$ . В ячейке A2 задаем формулу двустороннего приближения к предельной точке:  $=(-0,1)^{A1}$ .

Значение функции вычисляем по формуле  $=\sin(A2)/A2$  в ячейке A3.

Командами Сервис → Параметры → Вычисления открываем диалоговое окно, устанавливаем предельное число итераций 10. Нажимаем ОК. В ячейке A3 появляется результат.

	A	B	C	D
1	10			
2	1E-10			
3	1			
4				
5				

Рис. 3.4. Пример расчета предела с помощью электронной таблицы

### 3.3.2. Построение диаграмм и графиков

При производстве расчетов горный инженер в большинстве случаев получает результат в виде ряда чисел. Для того чтобы лучше понять полученные результаты, необходимо построить графики, диаграммы, чертежи рассчитанных конструкций. Иначе говоря, произвести графическую

обработку результатов вычислений. В графическом виде такие результаты становятся более наглядными и понятными. Научная графика предназначена для наглядного изображения объектов научных исследований, графической обработки результатов расчетов, проведения вычислительных экспериментов с наглядным представлением их результатов. Деловая графика – область компьютерной графики, предназначена для создания иллюстраций плановых показателей, отчетной документации, статистических сводок и т.п. Чаще всего это графики, круговые и столбчатые диаграммы. Программные средства научной и деловой графики обычно включаются в состав табличных процессоров (электронных таблиц).


Диаграммы и графики могут быть различных типов и соответственно представлять данные в различной форме. Для каждого набора данных важно правильно подобрать тип создаваемой диаграммы. Для наглядного сравнения различных величин, отображения изменения величин в зависимости от каких либо значений и построения графиков функций используются линейные графики. Например, с помощью столбчатой диаграммы можно отобразить добычу полезного ископаемого по месяцам на горнодобывающем предприятии. С помощью линейного графика можно представить аэродинамическую характеристику вентиляционного трубопровода.

Для отображения величин частей от целого применяется круговая диаграмма. Она позволяет, например, наглядно показать доли производительности добычных участков по предприятию.

Примеры построения диаграмм показаны на рис. 3.5.



**Рис.3.5. Линейная и круговая диаграммы**

Для создания графиков и диаграмм используется *Мастер диаграмм* , который имеет диалоговое окно, показанное на рис. 3.6.

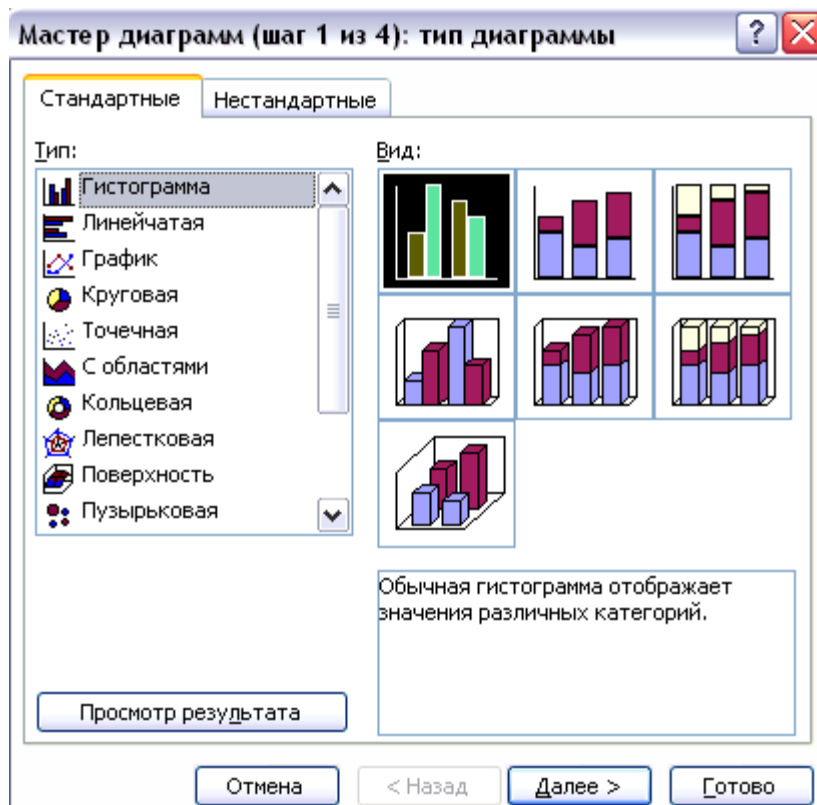


Рис. 3.6. Окно мастера диаграмм

Диаграммы могут располагаться как на листе с данными, так и на отдельных листах. Диаграммы связаны с исходными данными на рабочем листе и обновляются при обновлении исходных данных.

### 3.3.3. Вероятность и статистика

Статистика – наиболее содержательный раздел Excel. Он имеет обширный список встроенных функций, входящих в категорию статистических функций. Также прилагается пакет «Анализ данных», который находится в меню «Сервис»→»Настройка» и имеет список инструментов, показанный на рис. 3.7.

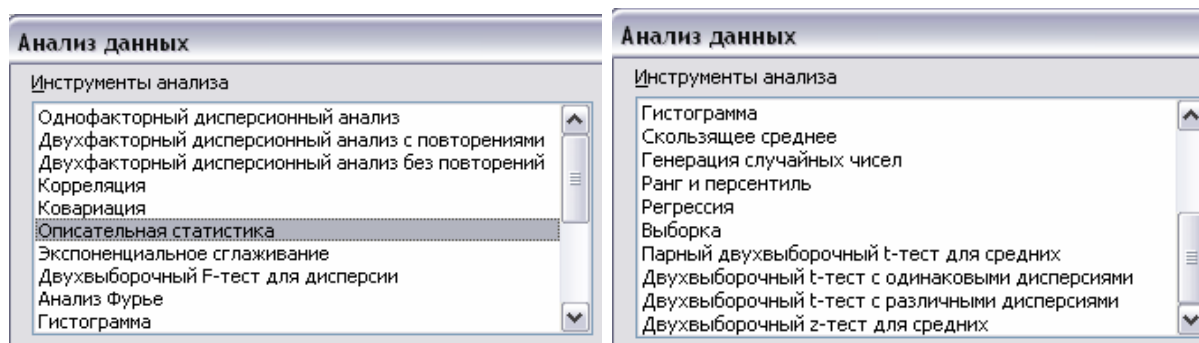


Рис. 3.7. Окно выбора функции анализа данных

### ***3.3.4. Надстройки в электронных таблицах***

Возможности электронных таблиц не ограничиваются вычислениями по формулам и построением диаграмм и графиков. С помощью надстроек электронных таблиц можно строить геоинформационные модели, решать уравнения методом «подбор параметра», решать задачи оптимизационного моделирования методом «поиск решения» и т.д.

Установить надстройки можно с помощью команды меню *Сервис – Надстройки*.

«Поиск решения» позволяет решать задачи оптимизационного моделирования, т.е. позволяет найти оптимальное значение формулы, содержащейся в ячейке, которая называется целевой. Эта процедура работает с группой ячеек, прямо или косвенно связанных с формулой в целевой ячейке. Чтобы получить по формуле, содержащейся в целевой ячейке, искомый результат, процедура изменяет значения во влияющих ячейках. Чтобы сузить множество значений, используемых в модели, применяются ограничения. Эти ограничения могут содержать ссылки на другие влияющие ячейки.

«Подбор параметра» используется при поиске значения аргумента функции, который обеспечивает требуемое значение функции. При подборе параметра изменяется значение в ячейке аргумента функции до тех пор, пока значение в ячейке самой функции не будет возвращать нужный результат.

Наиболее часто применяются надстройки при научных исследованиях и при решении задач проектирования.

### ***Контрольные вопросы и задания***

1. В чем преимущества хранения электронных документов перед бумажным способом?
2. Что такое гипертекст?
3. Какие редакторы используются для создания электронных документов?
4. Какие характерные режимы работы различных текстовых редакторов Вы знаете?
5. Что такое форматирование текста?
6. Перечислите основные файловые операции.
7. Какие текстовые форматы существуют?
8. Создайте текстовый файл в редакторе Word и сохраните его в таком формате, чтобы его можно было прочесть в приложении WordPad.
9. Для чего необходимы системы оптического распознавания документов?
10. Отсканируйте текст и переведите в текстовый формат.
11. Как ввести текст в ячейку электронной таблицы?

12. Определите при помощи встроенных функций значения среднего арифметического и среднеквадратического отклонения для приведенного ряда измерений: 28, 25, 19, 20, 29, 23, 19, 26, 24, 21, 25, 25, 24, 28, 25, 28, 29.

13. Пользуясь данными, приведенными в таблице, постройте график зависимости:

$x$	8	12	15	19	21	30
$y$	5	1	-2	-3	0	6

14. В каких случаях используются надстройки в электронных таблицах?

15. На каких этапах работы горного предприятия используются информационными технологиями обработки документов?

16. При проведении эксперимента для каждого значения  $X_i$  определены значения  $Y_{ij}$ . Провести обработку результатов эксперимента. Подобрать зависимость для полученных результатов с высоким корреляционным отношением. Оценить надежность полученных результатов.

$X_i$	$Y_{ij}$									
1	1.1	1.2	1.1	1.3	1.25	1.22	1.3	1.8	1.1	
1.2	2.2	2.4	2.6	2.25	2.23	2.2	2.5	2.4	2.3	
1.4	2.9	2.7	2.4	2.7						
1.6	3.4	3.6	3.5	3.4	4.8	3.2	3.5			
1.8	2.8	2.8	2.8	2.9	3	2.8	2.9	3	3.9	2.9
2	2.4	2.4	2.5	2.1	2.4	2.3	2.1	2.4	2.2	2.3
2.2	1.5	1.3	1.2	1.5	1.7	1.1	1.2	1.2	1.3	
2.4	0.9	0.8	1	0.9	0.9	0.1	0.9	0.9	0.8	

17. Исследовать и построить вариационный ряд по следующим результатам измерения величины продвижения забоя выработки за цикл, м:

1,95 1,80 1,95 2,00 1,95 2,00 2,00 1,90 2,05 2,00 1,95 1,90 1,90 1,95  
 2,00 1,95 1,95 1,90 1,90 2,00 1,95 1,95 1,95 1,95 1,95 2,10 1,95 1,95  
 1,95 2,00 2,00 1,95 1,95 1,95 1,95 1,95 1,95 1,95 1,95 2,00 1,95 1,95  
 1,95 1,95 1,95.

18. Исследовать и построить вариационный ряд по следующим результатам измерения напряжений в кровле выработки, кПа:

19,1 18,2 19,3 20,4 19,5 20,5 20,4 19,3 20,2 20,1 19,0 19,0 19,9 19,8  
 20,7 19,6 19,6 19,5 19,4 20,4 19,4 19,4 19,4 19,4 19,4 21,5 19,9 19,8  
 19,1 20,1 20,8 19,7 19,1 21,1 19,2 19,4 21,2 19,9 19,8 19,4 19,2 20,6  
 19,1 19,6 19,1.

## 4. БАЗЫ ДАННЫХ И СУБД

---

### 4.1. Данные и персональный компьютер

**Информацией** называют любые сведения о каких-либо явлениях, событиях, процессах, являющиеся объектами восприятия, передачи, преобразования, хранения и использования. На основе информации углубляются познания законов развития материального мира, взаимосвязываются и координируются работы, контролируются процессы и принимаются решения [12].

**Предметной областью** называют определенную часть реального мира, представляющую интерес для конкретного исследования или планируемых действий и соответственно для использования и отображения в информационной системе (в банке данных или знаний).

Реальный мир воспринимается как последовательность разных, хотя иногда и взаимосвязанных, явлений. Люди всегда пытались описать эти часто непонятные для них явления. Такое описание называют данными, т.е. информация, фиксируемая в определенной форме и пригодная для последующей обработки, хранения и передачи, называется *данными*. Данные характеризуют некоторую предметную область и являются информационной моделью некоторой части реального мира.

Процесс восприятия состояния системы в виде данных, описывающих состояние системы, называется *фиксацией данных*.

Обычно для фиксации данных применяется конкретное средство общения (естественный язык, изображения) на конкретном носителе (на камне, доске, бумаге, а в современных условиях – на магнитном или оптическом носителе).

Естественный язык достаточно гибок для представления данных и их интерпретации (семантики), поэтому обычно и то, и другое фиксируется совместно. Примером может служить утверждение «Предельная прочность разрушаемых пород 100». Здесь «100» – данное, а «Предельная прочность разрушаемых пород» – его семантика.

Семантика, в широком смысле слова, – анализ отношения между языковыми выражениями и миром, реальным или воображаемым, а также само это отношение и совокупность таких отношений. Данное отношение состоит в том, что языковые выражения (слова, словосочетания, предложения, тексты) обозначают то, что есть в мире.

К данным можно отнести факты, явления, события, идеи, предметы, качества (или свойства), действия, способы совершения действий, отношения, ситуации и их последовательности.

Данные и интерпретация обычно разделены. Например, в верхней части табл. 4.1. «Характеристики проходческих комбайнов» отдельно от данных приводится их интерпретация. Это затрудняет работу с данными.

Таблица 4.1

## Характеристики проходческих комбайнов

Наименование параметров					
Тип комбайна, страна-производитель	Предельная прочность разрушаемых пород	Суммарная номинальная мощность электродвигателей	Номинальная мощность электродвигателя исполнительного органа	Высота по корпусу	Масса
Единица измерения					
	МПа	кВт	кВт	мм	т
КПД Украина	100	285	110	1500	35
КСП22 Украина	70	165	75	1600	28,4
П110 Украина	100	195	55	1400	39
4ПП2М КПЗ Украина	80	225	110	2100	45
АМ50 Австрия	80	155	100	1645	24
РН25 Англия	70	165	82	1920	25,4
1ГПКС Россия	70	110	55	1500	21
КП20Б Россия	80	180	90	1300	25

Еще большее разделение данных и интерпретации является следствием применения персональных компьютеров для ведения и обработки данных.

Персональный компьютер имеет дело главным образом с данными как таковыми. Персональный компьютер «не знает», является ли «80» предельной прочностью разрушаемых пород или номинальной мощностью исполнительного органа комбайна. Большая часть интерпретирующей информации вообще не фиксируется в явной форме. Это произошло, в частности, из-за того, что на первых этапах своего развития ЭВМ не обладали достаточными возможностями для обработки текстов на естественном языке – основном языке интерпретации данных. Во-вторых, стоимость памяти ЭВМ была первоначально весьма велика. Поэтому память использовалась для хранения самих данных, а интерпретация данных являлась частью программы, которая «знала», например, что шестое вводимое значение связано с массой проходческого комбайна, а пятое – с его высотой по корпусу. В связи с тем, что данные представляют собой не более чем совокупность битов в памяти ЭВМ, существующую вне интерпретации, это существенно повышало роль программы.

При совместном использовании одних и тех же данных возникает масса проблем. Часто встречаются случаи, когда пользователи создают разные наборы данных с одной и той же или сходной информацией, не



зная, что эта информация уже имеется. Кроме того, разработчики прикладных программ пишут их на разных языках (например, на Си, Паскале, Бейсике), или в разных средах разработки (например, в Delphi или в Visual Basic), организуя файлы с данными так, как это принято в данном языке или среде разработки. При этом одни и те же данные могут иметь в разных приложениях разную последовательность размещения в записи, разные форматы одних и тех же полей, т.е. совершенно разную организацию. Чрезвычайно трудно при этом обобщить данные, любое изменение структуры данных приводит к необходимости переписывания программ, которые используют эти данные.

## 4.2. Файлы и файловые системы

Для долговременного хранения больших объемов информации основными программными инструментами организации являются **файловые системы** и **базы данных**. Функциональное назначение у них общее – хранение данных, но они имеют существенные отличия и, как следствие, свои области применения.

С точки зрения прикладной программы **файл** – это именованная область внешней памяти, в которую можно записывать и из которой можно считывать данные. От конкретной системы управления файлами и, возможно, от типа файла, зависят структура данных, хранящихся в файле, способ доступа к этим данным, правила именования файлов. Распределение внешней памяти, отображение имен файлов в соответствующие адреса внешней памяти и обеспечение доступа к данным берет на себя система управления файлами.

**Файловые системы** – это набор прикладных программ, которые выполняют для пользователей некоторые операции, например, создание отчетов. Каждая программа хранит свои собственные данные и управляет ими [5].

Недостатком их является недостаточные возможности управления данными, большие затраты труда программистов, дублирование разработок.

Поскольку файловые системы являются общим хранилищем файлов, принадлежащих, вообще говоря, разным пользователям, системы управления файлами должны обеспечивать авторизацию доступа к файлам. В общем виде подход состоит в том, что по отношению к каждому зарегистрированному пользователю данной вычислительной системы для каждого существующего файла указываются действия, которые разрешены или запрещены данному пользователю.

Файлы в файловых системах, представляющих собой общее хранилище данных, могут принадлежать разным пользователям. Поэтому файловые системы должны обеспечивать разграничение доступа, т.е. авторизацию доступа к файлам. Общий подход состоит в том, что пользователь должен быть зарегистрирован, и для каждого файла указываются действия, которые разрешены или запрещены данному пользователю.

**Банки данных и банки знаний** являются наиболее совершенной и прогрессивной формой организации информации и знаний в персональных компьютерах. Главная их задача – ответы на информационные запросы пользователей к банку данных или банку знаний с целью получения иско-мой информации, т.е. обеспечение пользователей требуемой информацией.

**Банк данных** – это автоматизированная система, включающая базу данных, лингвистические, программные, технические, организационно-методические средства, обеспечивающие централизованное накопление и коллективное многоцелевое использование информации в различных об-ластях деятельности пользователей. В банке данных содержатся совокуп-ности фактов о качественных и количественных характеристиках конкрет-ных объектов предметной области.

Термин «банк данных» не является общепризнанным. Наиболее близким к нему в англоязычной литературе является термин «система баз данных» (data base system). Система баз данных включает базу данных, СУБД, соответствующее оборудование и персонал. Понятие «система баз данных» уже, чем банк данных, так как «банк» обозначает то, что хранится в нем и всю инфраструктуру, но по сути они одинаковы.

Банк данных является сложной человеко-машинной системой, вклю-чающей в свой состав различные взаимосвязанные и взаимозависимые компоненты. Ядром банка данных является база данных. Информац-онный компонент банка данных состоит из баз данных, схем баз данных и словарей данных. Словари данных имеют особо важное значение.

Банки данных классифицируют по следующим свойствам:

- 1) скорость доступа – определяет время реакции, т.е. получение от-вета на запрос пользователя;
- 2) доступность – определяет, какие данные, содержащиеся в БД, дос-тупны данной категории пользователей;
- 3) гибкость – определяет возможность получить ответ на сложные запросы;
- 4) целостность – отвечает за снижение избыточности данных, согласо-ванность данных при упорядочении обновления.

Структурная схема банка данных приведена на рис. 4.1.

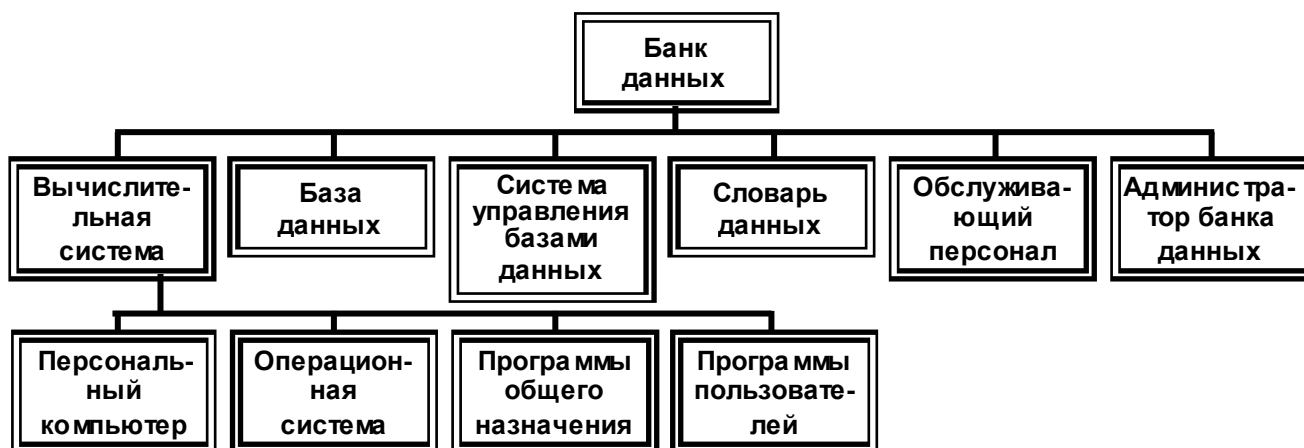


Рис. 4.1. Структурная схема банка данных

**Банк знаний** – это автоматизированная система, содержащая различные виды знаний (например, концептуальные, понятийные знания) о предметной области. Эти знания обычно выражаются в терминах данной предметной области. Знания, хранящиеся в банках знаний, используются для вывода новых знаний на основании специальных механизмов, имеющих в банках знаний.

В процессе создания и эксплуатации банков данных и банков знаний с ними взаимодействуют пользователи различных категорий, основными из которых являются конечные пользователи. Ими являются специалисты предметных областей, для удовлетворения информационных потребностей которых и создаются банки данных и банки знаний. Конечные пользователи различаются квалификацией, сферой интересов, информационными потребностями и т.п. Конечными пользователями могут быть как физические лица, так и различные вычислительные процессы, задачи, а иногда и целые системы, взаимодействующие с банками данных и банками знаний. Во всех случаях результатом взаимодействия является информация, данные, знания.

### 4.3. Концепция баз данных

Термин *«база данных»* (БД) начал применяться с 1963 г. и записывался на английском языке как *«data base»*. По мере развития вычислительной техники, эти два слова были объединены в одно (*database*). Основным смыслом, вкладываемым в термин *«база данных»*, – это база информационной системы, инструментом обработки данных в которой является ЭВМ. Информационная база или база данных представляет собой совокупность данных, предназначенных для совместного применения.

Термином база данных (*database*) в соответствии с принятой традицией обозначают набор данных, находящийся под контролем СУБД [13].

Одним из разработчиков теории баз данных Инглисом (R. Engles) в 1972 г. дано следующее рабочее определение: база данных представляет собой совокупность хранимых операционных данных, используемых прикладными системами некоторого предприятия. Другой классик теории баз данных К. Дейт в своих работах дает более предметное определение базы данных, как совокупности данных, хранящихся во вторичной памяти ЭВМ (на дисках). Одновременно российские разработчики теории баз данных предложили понимать под термином *«база данных»* даталогическое представление информационной модели предметной области. Это наиболее абстрактное и емкое определение. Государственным комитетом по науке и технике СССР (ГКНТ) в 1982 г. был принят ряд документов, определяющих базу данных как именованную совокупность данных, отражающую состояние объектов и их отношений в рассматриваемой предметной области. Таким образом, единого мнения по поводу определения термина *«база данных»* пока не существует. На основе анализа существующих определений и истории развития данной науки в дальнейшем будем пользоваться следующими определениями.

**Базы данных** – множество взаимосвязанных единиц данных, которые могут обрабатываться одной или несколькими прикладными системами.

**База данных** – это совместно используемый набор логически связанных данных (и описание этих данных), предназначенный для удовлетворения информационных потребностей организации [5].

База данных – это объект управления в банке данных. База данных описывает состояние объектов предметной области на определенный момент времени совокупностью предложений на некотором формализованном языке. При этом определяются значения всех факторов на данный момент в виде совокупности взаимосвязанных хранящихся вместе данных. Особенностью базы данных является то, что она рассчитана на использование при создании различных независимых программ и приложений. В этом основное отличие базы данных от обычных файлов данных – они используются только в конкретно разработанном для них приложении.

**Преимуществами базы данных являются наличие** метаданных, поддержка целостности базы данных, одновременный доступ к данным нескольких пользователей, ориентированные на пользователя запросы и отчеты. Назначение базы данных:

- Представление сложных структур информации, когда объектом хранения являются не только данные, но описания структур данных
- Сокращение дублирования информации.
- Независимость прикладных программ от изменений описаний данных и наоборот.
- Сокращение затрат на обслуживание базы данных.
- Интеграция данных (множество пользователей и прикладных программ).

**Характеристики баз данных и потоков данных:**

- Объем данных в логических и физических единицах.
- Потоки данных в единицу времени.
- Оперативность поступления данных.
- Полнота базы данных в %.
- Качество (достоверность) данных – вероятность ошибки (управление  $10^{-4}$   $10^{-5}$ , планирование –  $10^{-5}$ , статистика –  $10^{-5}$ , бухгалтер –  $10^{-6}$  -  $10^{-7}$ ).
- Обеспечение вероятности ошибок выше, чем  $10^{-4}$  требует увеличения капитальных и эксплуатационных затрат до 50%, времени программирования до 50%, времени работы программ, персонала.

Активная деятельность по отысканию приемлемых способов обобществления непрерывно растущего объема информации привела к созданию в начале 60-х годов специальных программных комплексов, называемых «Системы управления базами данных» (СУБД – **Database Management System – DBMS**).

**Система управления базами данных (СУБД)** состоит из языковых и программных средств, предназначенных для создания, ведения и эксплуатации баз данных.

*СУБД – это программное обеспечение, которое взаимодействует с прикладными программами пользователя и базой данных и обладает перечисленными ниже возможностями [5]:*

- Позволяет создать базу данных, что обычно осуществляется с помощью языка определения данных (**DDL – Data Definition Language**). Язык DDL предоставляет пользователям средства указания типа данных и их структуры, а также средства задания ограничений для информации, хранимой в базе данных.

- Позволяет вставлять, обновлять, удалять и извлекать информацию из базы данных, что обычно осуществляется с помощью языка манипулирования данными (**DML – Data Manipulation Language**). Наличие централизованного хранилища всех данных и их описаний позволяет использовать язык DML как общий инструмент организации запросов, который иногда называют языком запросов (query language). Наличие языка запросов позволяет устранить присущие файловым системам ограничения, при которых пользователям приходится иметь дело только с фиксированным набором запросов или постоянно возрастающим количеством программ, что порождает другие, более сложные проблемы управления программным обеспечением. Наиболее распространенным типом непроцедурного языка является язык структурированных запросов (**Structured Query Language – SQL**), который в настоящее время определяется специальным стандартом и фактически является обязательным языком для любых реляционных СУБД. (SQL произносится либо по буквам «S-Q-L», либо как мнемоническое имя «See-Que».)

- Предоставляет контролируемый доступ к базе данных с помощью перечисленных ниже средств:

- системы обеспечения защиты, предотвращающей несанкционированный доступ к базе данных со стороны пользователей;

- системы поддержки целостности данных, обеспечивающей непротиворечивое состояние хранимых данных;

- системы управления параллельной работой приложений, контролирующей процессы их совместного доступа к базе данных;

- системы восстановления, позволяющей восстановить базу данных до предыдущего непротиворечивого состояния, нарушенного в результате сбоя аппаратного или программного обеспечения;

- доступного пользователям каталога, содержащего описание хранимой в базе данных информации.

Другими словами, система управления базами данных (СУБД) – это программная система для централизованного управления данными, хранимыми в базе данных, и их поддержания в состоянии, соответствующем предметной области.

С функцией обработки данных тесно связано управление транзакциями. **Транзакция** – это последовательность операций над БД, рассматриваемых СУБД как единое целое. В процессе функционирования СУБД транзакция либо успешно выполняется и СУБД фиксирует (commit) изменения БД, произведенные ей во внешней памяти, либо ни одно из этих изменений никак не отражается на состоянии БД.

При помощи транзакций поддерживается логическая целостность базы данных за счет объединения элементарных операций над разными файлами в одну транзакцию.

Данные, хранящиеся в базе данных, представляют собой некоторый факт или совокупность фактов, с помощью которых описывается конкретная предметная область и на которых базируется последующий вывод или принимаемые решения.

Пусть, например, требуется хранить характеристики проходческих комбайнов и ряд других данных, связанных с организацией базы данных (БД «Проходка»). Используя для этого одну из современных «русифицированных» СУБД, можно подготовить следующее описание характеристик проходческих комбайнов:

```
СОЗДАТЬ ТАБЛИЦУ Проходческие_комбайны
(Тип_комбайна Текст (8)
Страна_изготовитель Текст (8)
Прочность_пород Целое
Суммарная_мощность Целое
Номинальная_мощность Целое
Высота_по_корпусу Целое
Масса Целое
```

и ввести его вместе с данными в БД "Проходка".

Язык запросов СУБД позволяет обращаться за данными как из программ, так и с терминалов. Сформировав запрос

```
ВЫБРАТЬ Тип_комбайна
ИЗ ТАБЛИЦЫ Проходческие_комбайны
ГДЕ Страна_изготовитель = 'Россия'
И Прочность_пород > 70;
```

получим характеристику комбайна КП20Б, а по запросу

```
ВЫБРАТЬ Тип_комбайна
ИЗ ТАБЛИЦЫ Проходческие_комбайны
ГДЕ Масса < 36
И Высота_по_корпусу < 1550;
```

получим характеристики всех проходческих комбайнов, масса которых не превышает 36 тонн, а высота по корпусу меньше 1550 мм. В данном случае это комбайны КПД(Украина), 1ГПКС(Россия), и КП20Б(Россия).

#### 4.4. Классификация баз данных

По технологии обработки данных базы данных подразделяют на: *централизованные* и *распределенные*.

Централизованная база данных хранится в памяти одной вычислительной системы. Если эта вычислительная система является компонентом сети ЭВМ, возможен распределенный доступ к такой базе.

Такой способ использования баз данных часто применяют в локальных сетях персональных компьютеров.

Распределенная база данных состоит из нескольких, возможно пересекающихся или даже дублирующих друг друга частей, хранимых в различных персональных компьютерах вычислительной сети. Работа с такой базой осуществляется с помощью системы управления распределенной базой данных (СУРБД).

По способу доступа к данным базы данных разделяются на *базы данных с локальным доступом* и *базы данных с удаленным (сетевым) доступом*.

Системы централизованных баз данных с сетевым доступом предполагают различные архитектуры подобных систем:

1. **Файл-сервер.** Архитектура систем баз данных с сетевым доступом предполагает выделение одной из машин сети в качестве центральной (сервер файлов). На такой машине хранится совместно используемая централизованная база данных. Все другие машины сети выполняют функции рабочих станций, с помощью которых поддерживается доступ пользовательской системы к централизованной базе данных.

Файлы базы данных в соответствии с пользовательскими запросами передаются на рабочие станции, где в основном и производится обработка. При большой интенсивности доступа к одним и тем же данным производительность информационной системы падает. Пользователи могут создавать также на рабочих станциях локальные базы данных, которые используются ими монополично.

2. **Клиент-сервер.** В этой концепции подразумевается, что помимо хранения централизованной базы данных центральная машина (сервер базы данных) должна обеспечивать выполнение основного объема обработки данных. Запрос на данные, выдаваемый клиентом (рабочей станцией), порождает поиск и извлечение данных на сервере. Извлеченные данные (но не файлы) транспортируются по сети от сервера к клиенту. Спецификой архитектуры клиент-сервер является использование языка запросов SQL.

#### 4.5. Структурные элементы и модели баз данных

Понятие базы данных тесно связано с такими понятиями структурных элементов, как поле, запись, файл (таблица).

**Поле** – элементарная единица логической организации данных, которая соответствует неделимой единице информации – реквизиту. Для

описания поля используются следующие характеристики: имя, тип, длина (определяется максимально возможным количеством символов); точность для числовых данных.

**Запись** – совокупность логически связанных полей.

**Файл (таблица)** – совокупность экземпляров записей одной структуры.

В структуре записи файла указываются поля, значения которых являются ключами: первичными (ПК), которые идентифицируют экземпляр записи, и вторичными (ВК), которые выполняют роль поисковых или группировочных признаков (по значению вторичного ключа можно найти несколько записей).

Ядром любой базы данных является модель данных. Модель данных – формализованное описание, отражающее состав и типы данных, а также взаимосвязь между ними. Модель данных представляет собой множество структур данных, ограничений целостности и операций манипулирования данными. С помощью модели данных могут быть представлены объекты предметной области и взаимосвязи между ними.

Модель данных – совокупность структур данных и операций их обработки.

СУБД основывается на использовании иерархической, сетевой или реляционной модели, на комбинации этих моделей или на некотором их подмножестве.

#### 4.5.1. Иерархическая модель данных

Иерархическая структура представляет совокупность элементов, связанных между собой по определенным правилам. Объекты, связанные иерархическими отношениями, образуют ориентированный граф (перевернутое дерево).

К основным понятиям иерархической структуры относятся: уровень, элемент (узел), связь. Узел – это совокупность атрибутов данных, описывающих некоторый объект. На схеме иерархического дерева узлы представляются вершинами графа. Каждый узел на более низком уровне связан только с одним узлом, находящимся на более высоком уровне.

Схематично иерархическая модель данных представлена на рис 4.2.

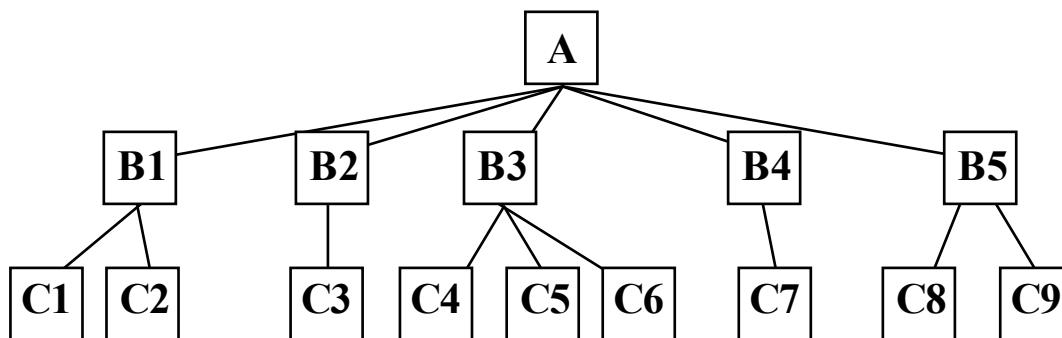


Рис. 4.2. Схема иерархической модели базы данных



### 4.5.2. Сетевая модель данных

В сетевой структуре при тех же основных понятиях (уровень, узел, связь) каждый элемент может быть связан с любым другим элементом. Сетевую модель можно представить как граф с записями в виде узлов графа и наборами в виде его ребер (рис. 4.3).

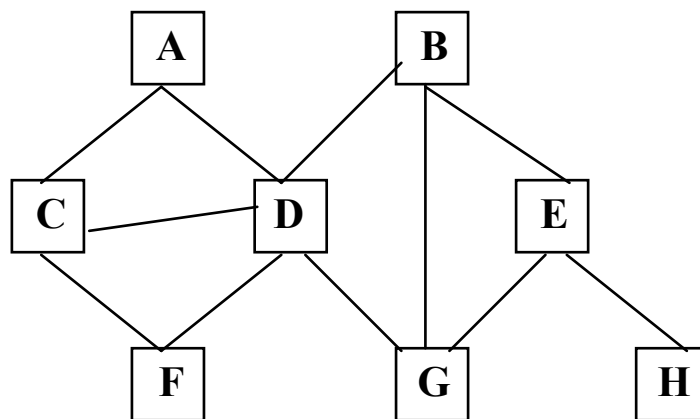


Рис. 4.3. Схема сетевой модели базы данных

Граф является более общей структурой по сравнению с деревом (иерархия), поскольку отдельный узел может иметь произвольное количество родительских узлов, а также произвольное количество подчиненных узлов.

### 4.5.3. Реляционная модель данных

В реляционной модели данных используется своеобразная терминология, но это не меняет сущности модели. Так, на логическом уровне элемент чаще всего называют атрибутом; кроме того, для него используются термины колонка, столбец, поле. Совокупность атрибутов образует кортеж (ряд, запись, строку). Совокупность кортежей образует отношение (таблицу или файл БД).

Связи между файлами в реляционной модели в явном виде могут не описываться. Они устанавливаются динамически в момент обработки данных по равенству значений соответствующих полей.

Структуры записей в реляционных базах данных – линейные.

Каждое отношение по определению имеет ключ, т.е. атрибут (простой ключ) или совокупность атрибутов (составной ключ), однозначно идентифицирующий кортеж.

Атрибут или группа атрибутов, которая в рассматриваемом отношении не является ключом, а в другом отношении ключом является, называется внешним ключом.

Если какая-то таблица содержит внешний ключ, то она:

- а) логически связана с таблицей, содержащей соответствующий первичный ключ;
- б) эта связь имеет характер один ко многим.

*Понятие реляционный* (англ. *relation* – отношение) связано с разработками известного американского специалиста в области систем баз данных Е. Кодда.

Эти модели характеризуются простотой структуры данных, удобным для пользователя табличным представлением и возможностью использования формального аппарата алгебры отношений и реляционного исчисления для обработки данных.

Реляционная модель ориентирована на организацию данных в виде двумерных таблиц. Каждая *реляционная таблица* представляет собой двумерный массив и обладает следующими свойствами:

- каждый элемент таблицы – один элемент данных;
- все столбцы в таблице однородные, т.е. все элементы в столбце имеют одинаковый тип (числовой, символьный и т.д.) и длину;
- каждый столбец имеет уникальное имя;
- одинаковые строки в таблице отсутствуют;
- порядок следования строк и столбцов может быть произвольным.

Отношения представлены в виде *таблиц*, строки которых соответствуют *записям*, а столбцы – *полям*.

Поле, каждое значение которого однозначно определяет соответствующую запись, называется *простым ключом* (ключевым полем). Если записи однозначно определяются значениями нескольких полей, то такая таблица базы данных имеет *составной ключ*.

Чтобы связать две реляционные таблицы, необходимо ключ первой таблицы ввести в состав ключа второй таблицы (возможно совпадение ключей); в противном случае нужно ввести в структуру первой таблицы *внешний ключ* – ключ второй таблицы.

*Нормализация* – это разбиение таблицы на две или более, обладающих лучшими свойствами при включении, изменении и удалении данных. Окончательная цель нормализации сводится к получению такого проекта базы данных, в котором каждый факт появляется лишь в одном месте, т.е. исключена избыточность информации. Это делается не столько с целью экономии памяти, сколько для исключения возможной противоречивости хранимых данных.

Каждая таблица в реляционной базе данных удовлетворяет условию, в соответствии с которым в позиции на пересечении каждой строки и столбца таблицы всегда находится единственное значение и никогда не может быть множества таких значений. Любая таблица, удовлетворяющая этому условию, называется *нормализованной*. Фактически, ненормализованные таблицы, т.е. таблицы, содержащие повторяющиеся группы, даже не допускаются в реляционной базе данных.

Всякая нормализованная таблица автоматически считается таблицей в первой нормальной форме, сокращенно 1НФ. Таким образом, строго говоря, «нормализованная» и «находящаяся в 1НФ» означают одно и то же. Однако на практике термин «нормализованная» часто используется в более узком смысле – «полностью нормализованная», который означает, что в проекте не нарушаются никакие принципы нормализации.

Теперь в дополнение к 1НФ можно определить дальнейшие уровни нормализации – вторую нормальную форму (2НФ), третью нормальную форму (3НФ) и т.д. По существу, таблица находится в 2НФ, если она находится в 1НФ и удовлетворяет, кроме того, некоторому дополнительному условию. Таблица находится в 3НФ, если она находится в 2НФ и, помимо этого, удовлетворяет еще другому дополнительному условию и т.д.

Таким образом, каждая нормальная форма является в некотором смысле более ограниченной, но и более желательной, чем предшествующая.

#### 4.6. Обзор СУБД

Системой управления базами данных называют *программную систему*, предназначенную для создания на ЭВМ общей базы данных, используемой для решения множества задач. Подобные системы служат для поддержания базы данных в актуальном состоянии и обеспечивают эффективный доступ пользователей к содержащимся в ней данным в рамках предоставленных пользователям полномочий.

СУБД предназначена для централизованного управления базой данных в интересах всех работающих в этой системе и представляет собой программное обеспечение, которое управляет доступом к хранимой в базе данных информации. Этот процесс производится следующим образом:

- 1) пользователь выдает запрос на доступ, применяя команды определенного языка манипулирования данными (в этих целях обычно используется язык SQL);
- 2) СУБД перехватывает этот запрос и анализирует его;
- 3) СУБД просматривает внешнюю схему для пользователя, концептуальную, внутреннюю схему и определяет структуру хранения информации;
- 4) СУБД выполняет необходимые операции над хранимой базой данных.

По степени универсальности различают два класса СУБД:

- системы общего назначения;
- специализированные системы.

СУБД общего назначения не ориентированы на какую-либо предметную область или на информационные потребности какой-либо группы пользователей. Каждая система такого рода реализуется как программный продукт, способный функционировать на некоторой модели ЭВМ в опре-

деленной операционной системе и поставляется многим пользователям как коммерческое изделие. Такие СУБД обладают средствами настройки на работу с конкретной базой данных. Использование СУБД общего назначения в качестве инструментального средства для создания автоматизированных информационных систем, основанных на технологии баз данных, позволяет существенно сокращать сроки разработки, экономить трудовые ресурсы. Этим СУБД присущи развитые функциональные возможности и определенная функциональная избыточность.

Специализированные СУБД создаются в редких случаях при невозможности или нецелесообразности использования СУБД общего назначения.

СУБД общего назначения – это сложные программные комплексы, предназначенные для выполнения всей совокупности функций, связанных с созданием и эксплуатацией базы данных информационной системы.

Рынок программного обеспечения ПК располагает большим числом разнообразных по своим функциональным возможностям коммерческих систем управления базами данных общего назначения, а также средствами их окружения практически для всех массовых моделей машин и для различных операционных систем.

Используемые в настоящее время СУБД обладают средствами обеспечения целостности данных и надежной безопасности, что дает возможность разработчикам гарантировать большую безопасность данных при меньших затратах сил на низкоуровневое программирование. Продукты, функционирующие в среде Windows, выгодно отличаются удобством пользовательского интерфейса и встроенными средствами повышения производительности.

Производительность СУБД оценивается:

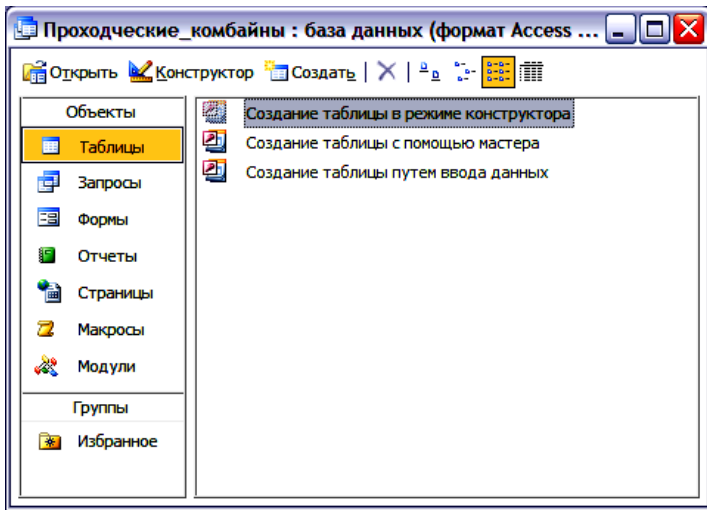
- временем выполнения запросов;
- скоростью поиска информации в неиндексированных полях;
- временем выполнения операций импортирования базы данных из других форматов;
- скоростью создания индексов и выполнения таких массовых операций, как обновление, вставка, удаление данных;
- максимальным числом параллельных обращений к данным в многопользовательском режиме;
- временем формирования отчета.

Следует отличать СУБД от системы управления файлами, не учитывающей внутреннюю структуру хранимых данных (записей), имеющих особую поддержку безопасности и целостности данных.

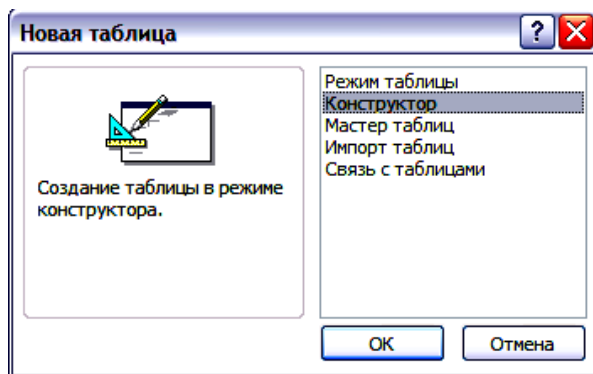
## 4.7. Пример создания таблицы в MS Access

Таблицы – основа любой базы данных. Умение создавать их, умело связывать между собой определит в дальнейшем простоту работы с ней и объем самой базы данных.

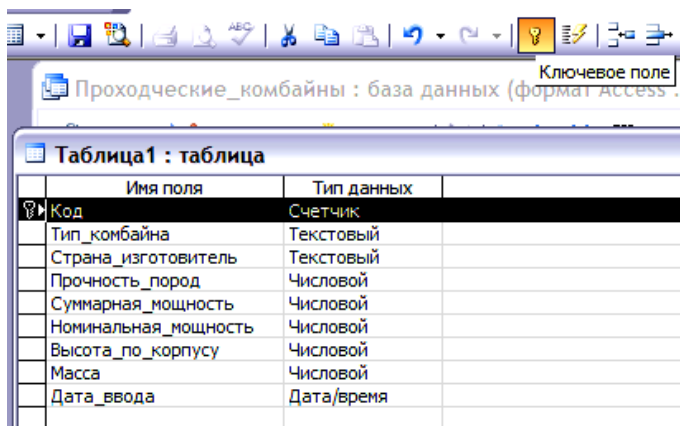
Рассмотрим технологию создания в режиме конструктора таблицы «Проходческие\_комбайны».



1. Чтобы создать простейшую таблицу, откройте окно базы данных, перейдите на вкладку «Таблицы» и нажмите кнопку «Создать».



2. Выберите в появившемся окне вариант «Конструктор» и нажмите кнопку «ОК».



3. Перед Вами развернется окно конструктора таблиц. Задайте в нем необходимые поля. Выберите для каждого поля тип.

4. Пусть в нашей таблице будут поля "Код", "Тип\_комбайна", "Страна\_изготовитель", "Прочность\_пород", "Суммарная\_мощность", "Номинальная\_мощность", "Высота\_по\_корпусу", "Масса", "Дата\_ввода".

Для поля "Код" выберите тип "Счетчик", полей "Тип\_комбайна" и "Страна\_изготовитель" – тип "Текстовый", поля "Дата\_ввода" – тип "Дата/время", для остальных полей – тип "Числовой", т.е. в поле "Тип\_комбайна" мы будем хранить некое текстовое значение, например, название проходческого комбайна, а в поле "Дата\_ввода" – дату, например, дату ввода информации о комбайне в таблицу.

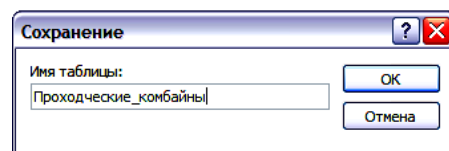
Поле "Код" будет хранить некий уникальный код для строки. Существование такого уникального кода позволит нам однозначно идентифицировать запись (строку) таблицы, т.е. выбирая строку с кодом, равным единице, мы выбираем только эту строку и никакую другую. С другой стороны, при выборке строк с датой, равной 1 января 2000 года, может оказаться, что в ней несколько записей.

Чтобы указать, что поле является уникальным ключом, необходимо поставить курсор в это поле и нажать кнопку на панели инструментов с изображением ключика. Теперь при попытке создания записи с таким же кодом программа сообщит об ошибке ввода и, тем самым, предупредит создание записей с одинаковым кодом.

Тип данных «Счетчик» означает, что программа сама при создании записи будет вносить в это поле данные по особому алгоритму. MS Access позволяет присваивать номера в поле типа «Счетчик» либо по порядку, т.е. «1, 2, 3 и т.д.», либо в случайном порядке в диапазоне целых чисел.

В случае, если счетчик дойдет до конца диапазона (хотя это маловероятно), то счетчик начнет присваивать значения снова с единицы.

5. Сохраните таблицу под именем «Проходческие\_комбайны».

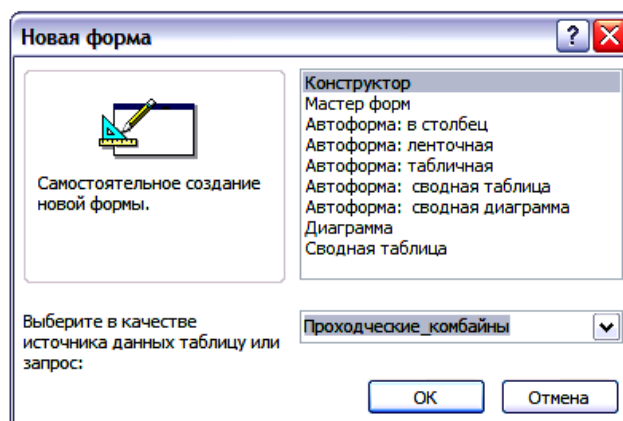


#### 4.8. Пример создания формы для ввода данных в MS Access

Формы – одно из основных средств для работы с базами данных в Access – используются для ввода новых записей (строк таблиц), просмотра и редактирования уже имеющихся данных, задания параметров запросов и вывода ответов на них и др. *Формы* представляют собой прямоугольные окна с размещенными в них элементами управления. Существует возможность создания форм динамически при исполнении программы, однако естественным режимом их создания является режим визуального конструирования.

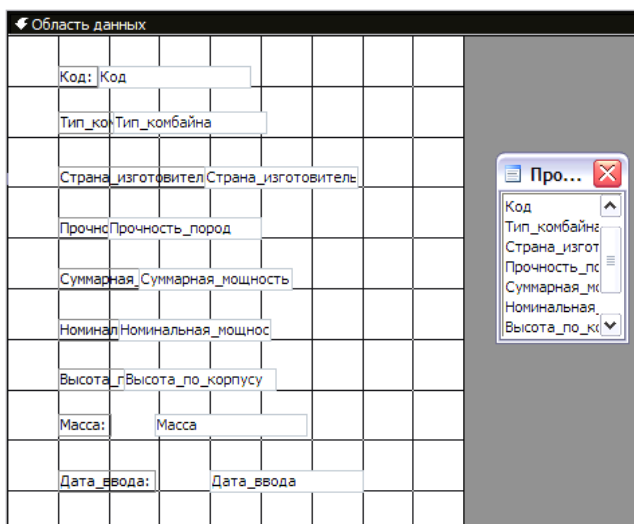
Рассмотрим технологию создания однотоличной формы в режиме конструктора на примере формы для редактирования таблицы "Проходческие\_комбайны".

1. В окне БД выберите пункт «Формы», режим конструктора.



2. В окне «Новая форма» выберите источник данных – таблицу «Проходческие\_комбайны».

3. Установите панель элементов нажатием соответствующей кнопки.

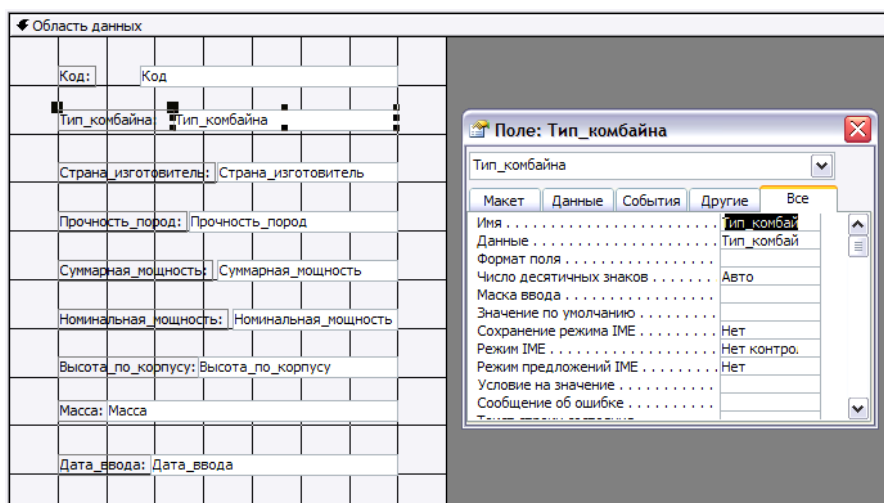


4. Установите на форме в один столбец 8 элементов. Обратите внимание, что элементы устанавливаются вместе с надписями

5. Для каждого из установленных элементов откройте окно свойств и задайте:

– на вкладке "Данные" в поле "Данные" – поле – источник данных;

– на вкладке "Другие" неформальное имя – идентификатор поля.



6. Замените формальные тексты в надписях на содержательные. Это можно сделать прямо на форме щелчком мыши внутри области надписи.

7. Измените размеры полей, потянув мышью за маркеры. Большие маркеры в левых верхних углах элементов служат для перемещения только этих элементов. Если же "потянуть" мышью за рамку объекта, то надпись и поле будут перемещаться вместе. В результате после открытия форма должна иметь вид, показанный на рисунке.

8. Занесите данные в соответствующие поля формы.

The image shows two side-by-side screenshots of a form. The left screenshot shows the form with the following data: Код: 5, Тип\_комбайна: AM50, Страна\_изготовитель: Австрия, Прочность\_пород: 80, Суммарная\_мощность: 155, Номинальная\_мощность: 100, Высота\_по\_корпусу: 1645, Масса: 24, Дата\_ввода: 21.07.2006. The right screenshot shows the same form, but the 'Код' field is highlighted with a black box containing the word 'Счетчик' (counter), and the other fields are empty.

#### 4.9. Введение в язык SQL

SQL (Structured Query Language – структурированный язык запросов) предназначен для обработки реляционных баз данных [14].

В различных СУБД используются два типа языка SQL:

1) интерактивный SQL – используется для выполнения действий непосредственно над базами данных в оперативном режиме. Как правило, интерактивному SQL сопутствует некоторая программа-сервер SQL. Наиболее популярны в настоящее время Oracle SQL-Server, MS-SQL Server, InterBase и некоторые другие;

2) встроенный SQL – состоит из команд SQL, включенных непосредственно в программы, которые написаны на другом языке программирования. Например, в таких системах программирования, как Delphi, C++ Builder, VBasic, VC++ имеется встроенный SQL. В различных СУБД, например в MS Access, также имеются возможности использовать команды встроенного SQL.

Основу языка SQL составляют операторы, которые можно условно разбить на несколько групп по функциональному назначению. Основными группами операторов, наиболее часто используемыми пользователями реляционных баз данных, являются:



– операторы определения объектов базы данных – операторы DDL (Data Definition Language). Примерами таких операторов будут:

- **CREATE SCHEMA** – создать схему базы данных;
- **DROP SCHEMA** – удалить схему базы данных;
- **CREATE TABLE** – создать таблицу;
- **ALTER TABLE** – изменить таблицу;
- **DROP TABLE** – удалить таблицу;
- **CREATE VIEW** – создать представление;
- **DROP VIEW** – удалить представление.

– операторы манипулирования данными – операторы DML (Data Manipulation Language), например,

- **SELECT** – выбрать строки из таблицы;
- **INSERT** – добавить строки в таблицу;
- **UPDATE** – изменить строки в таблице;
- **DELETE** – удалить строки из таблицы;
- **COMMIT** – зафиксировать изменения;
- **ROLLBACK** – отменить внесенные изменения.

– операторы защиты и управления доступом, такие как:

- **CREATE ASSERTION** – создать ограничение;
- **DROP ASSERTION** – отменить ограничение;
- **GRANT** – предоставить привилегии;
- **REVOKE** – отменить привилегии.

Язык SQL оперирует иными терминами, чем это принято в реляционной теории. Так, вместо «отношений» используется термин «таблицы», вместо «кортежей» – «строки», вместо «атрибутов» – «колонки» или «столбцы». SQL также имеет специальные термины, среди которых используются такие слова, как «запрос», «предложение», «предикат».

Команды языка SQL являются инструкциями, с помощью которых пользователь обращается к базе данных. Команды состоят из одной или нескольких логических частей, называемых предложениями. Предложения начинаются ключевым словом, то есть словом, которое имеет специальное значение в SQL, и состоят из ключевых слов, определяющих имя предложения, и аргументов. Примерами предложений являются следующие выражения:

- **FROM** Проходческие\_комбайны;
- **WHERE** Страна\_изготовитель = «Украина».

«FROM» является ключевым словом предложения FROM, а «Проходческие\_комбайны» – аргументом предложения. Аналогично «Страна\_изготовитель = «Украина» – аргумент предложения WHERE. Таким образом, приведенные выше операторы являются ключевыми словами и могут использоваться для построения предложений.

Реализация в SQL концепции операций, ориентированных на табличное представление данных, позволило создать компактный язык с небольшим набором предложений. В нем существуют:

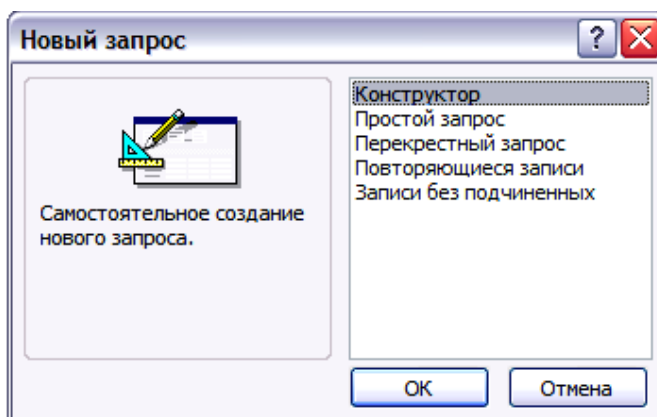
- предложения определения данных: определение баз данных, а также определение и уничтожение таблиц и индексов;
- запросы на выбор данных: предложение **SELECT**;
- предложения модификации данных: добавление, удаление и изменение;
- предложения управления данными: предоставление и отмена привилегий на доступ к данным, управление транзакциями и другие.

Кроме того, он предоставляет возможность выполнять в этих предложениях:

- арифметические вычисления (включая разнообразные функциональные преобразования), обработку текстовых строк и выполнение операций сравнения значений арифметических выражений и текстов;
- упорядочение строк и (или) столбцов при выводе содержимого таблиц на печать или экран дисплея;
- создание представлений (виртуальных таблиц), позволяющих пользователям иметь свой взгляд на данные без увеличения их объема в базе данных;
- запоминание выводимого по запросу содержимого таблицы, нескольких таблиц или представления в другой таблице (реляционная операция присваивания);
- агрегирование данных: группирование данных и применение к этим группам таких операций, как среднее, сумма, максимум, минимум, число элементов и других.

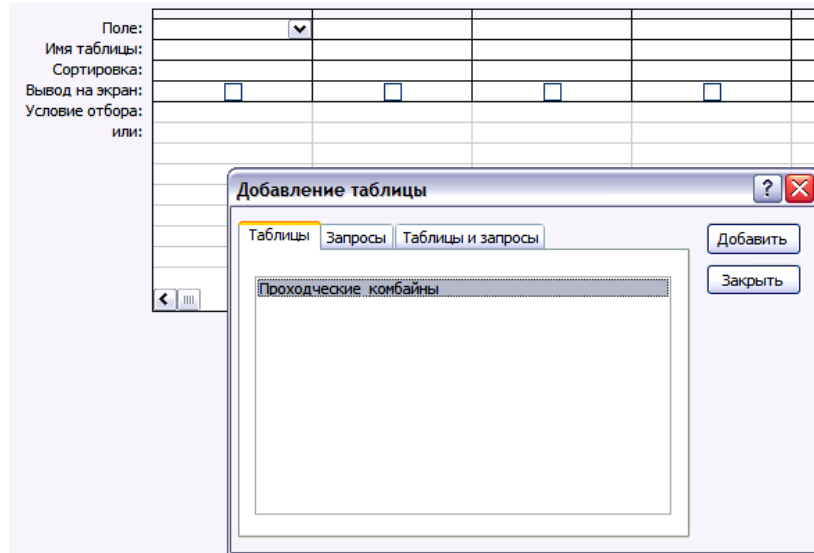
#### 4.10. Пример создания запроса в MS Access

Умение создавать запросы позволит вам легко и быстро сделать любую выборку из базы. Не говоря о том, что любой источник данных (списков, форм, отчетов) тоже запрос. Здесь рассматривается создание простейшего запроса.



1. Для создания запроса, перейдите в окне базы данных на вкладку "Запросы" и нажмите кнопку "Создать". Выберите вариант "Конструктор" и нажмите кнопку "ОК".

2. По умолчанию, вам будет предложено добавить в запрос таблицы (другие запросы) из списка. Если этого не произошло или запрос был создан ранее, то нажмите на панели инструментов кнопку, где изображен желтый плюсик над пиктограммой таблицы. Двойным кликом (кнопкой «Добавить») добавьте таблицы в запрос. Затем нажмите кнопку «Закреть».



Мы используем в запросе всего одну таблицу. Чтобы выбрать нужные поля из таблицы, просто двойным кликом нажмите на их названия в окошке таблицы. Другой вариант: выберите их из списка в первой строке любого столбца в запросе. Если таблиц несколько, то чтобы ограничить список в первой строке, во второй строке соответствующего столбца выберите из списка нужную таблицу.

3. Чтобы задать условие, в соответствующей строке "Условие" и нужных столбцах (в нашем случае "Страна\_изготовитель" и "Прочность\_пород") запишите условия. Допустимым является использование любых операторов и функций MS Access. Даты, если задаются константой, а не переменной, должны быть ограничены символом # по краям. Текстовые константы должны быть помещены в кавычки. Впрочем, MS Access сам поместит константу в кавычки, если поле текстовое.

Поле:	Тип_комбайна	Страна_изготовитель	Прочность_пород
Имя таблицы:	Проходческие_комбайны	Проходческие_комбайны	Проходческие_комбайны
Сортировка:			
Вывод на экран:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Условие отбора:		= "Россия"	>70
или:			

4. Результат выборки можно посмотреть, сохранив запрос, а потом нажать кнопку "Открыть" или открыть его в режиме просмотра (крайняя левая кнопка на панели инструментов).

	Тип_комбайна	Страна_изготовитель	Прочность_пород
	КП20Б	Россия	80
▶			0

5. В соответствии с запросом получаем: комбайн КП20Б, страна-изготовитель – Россия, предельная прочность разрушаемых пород – 80 МПа (запросу соответствует единственная запись в таблице).

Более интересным и полезным для изучения языка SQL оказывается просмотр запроса в режиме SQL.

```
SELECT Проходческие_комбайны.Тип_комбайна, Проходческие_комбайны.Страна_изготовитель,
Проходческие_комбайны.Прочность_пород
FROM Проходческие_комбайны
WHERE (((Проходческие_комбайны.Страна_изготовитель)="Россия") AND
(((Проходческие_комбайны.Прочность_пород)>70));
```

Создание запросов в конструкторе достаточно простое дело, пока не требуется создание вложенных запросов или запросов на объединение, где без хорошего знания языка SQL (Structured Query Language) не обойтись. Поэтому создавайте запросы и изучайте, как они выглядят в SQL.

## 4.11. Разработка баз данных

### Этапы разработки баз данных

При проектировании базы данных преследуются те же цели, что и при проектировании автоматизированной системы управления (АСУ) предприятием, а этапы и стадии её создания практически совпадают с работами по созданию АСУ. Принято выделять следующие этапы при разработке базы данных, при помощи которых осуществляется переход от предметной области к её конкретной реализации:

- изучение предметной области;
- разработка концептуальной модели предметной области;
- разработка логической модели данных;
- разработка физической модели данных;
- разработка собственно базы данных.

В настоящее время существуют три уровня абстракции для определения структуры базы данных:

- концептуальный;
- логический;
- физический.

**Изучение предметной области** заключается в выявлении существенных процессов и факторов, оказывающих определяющее влияние на достижение целей внедрения информационной системы.

**Концептуальная модель** представляет собой объекты и их взаимосвязи без указания способов их физического хранения, то есть концептуальная структура (или схема) состоит из основных элементов данных предметной области (личности, факты), называемых объектами; элементарных данных, описывающих свойства и признаки объектов и называемых атрибутами; связей между экземплярами данных, которые могут быть либо ассоциациями, либо отображениями. Таким образом, концептуальная модель является, по существу, моделью предметной области.

Модель предметной области – это записанные знания об объектах реального мира, которыми необходимо управлять наиболее рациональным образом. Эти знания могут быть представлены как в чисто текстовом виде, так и с использованием методологий структурного функционального моделирования – SADT, IDEF0, IDEF3, методологий и стандартов описания состава, структуры и взаимосвязей используемой в деятельности предприятия информации – IDEF1, DFD и соответствующих инструментальных CASE-средств – BPWin, AIO WIN, ProCap, ProSim, SmartER.

Концептуальная модель трансформируется затем в модель данных, совместимую с выбранной СУБД. Возможно, что отраженные в концептуальной модели взаимосвязи между объектами окажутся впоследствии нереализуемыми с помощью средств выбранной СУБД. Это потребует изменения концептуальной модели.

Логической называется версия концептуальной модели, которая может быть обеспечена средствами СУБД.

**Логическая модель данных** описывает понятия предметной области и их взаимосвязи и является прототипом будущей базы данных. Логическая модель разрабатывается в терминах информационных понятий, но без какой-либо ориентации на конкретную СУБД. Наиболее широко используемым средством разработки логических моделей баз данных являются диаграммы «сущность-связь» – Entity-Relationship (ER-диаграммы). Следует заметить, что логическая модель данных, представленная ER-диаграммами, в принципе, может быть преобразована как в реляционную модель данных, так и в иерархическую, сетевую, постреляционную.

**Физическая модель данных** строится на базе логической модели и описывает данные уже средствами конкретной СУБД. Отношения, разработанные на стадии логического моделирования, преобразуются в таблицы, атрибуты в столбцы, домены в типы данных, принятых в выбранной конкретной СУБД. На этапах логического и физического моделирования, как правило, используется стандарт IDEF1X и CASE-средства ERWin или SmartER. Указанные инструментальные средства проектирования поддерживают несколько десятков наиболее популярных СУБД. Результатом фи-

зического моделирования является генерация программного кода базы данных на соответствующем выбранной СУБД диалекте структурированного языка запросов SQL.

Несмотря на постоянно совершенствуемые возможности CASE-средств по автоматической генерации кода баз данных, детальное её проектирование все-таки остается работой и заботой человека. CASE-средства помогают создать прототип базы данных, на котором строится её рабочая версия. Поскольку практически любая база данных кроме таблиц содержит дополнительный программный код в виде триггеров и хранимых процедур, которые пишутся на процедурных расширениях языка SQL или универсальном языке программирования, то полностью автоматизировать её создание из логической модели пока не представляется возможным, а может быть и нужным.

**Хранимые процедуры**, основным назначением которых является реализация бизнес-процессов предметной области, – это процедуры и функции, хранящиеся непосредственно в базе данных в откомпилированном виде, которые могут запускаться непосредственно пользователем или прикладными программами, работающими с базой данных.

**Триггеры** – это хранимые процедуры, связанные с некоторыми событиями, происходящими во время работы базы данных. В качестве таких событий обычно выступают операции вставки, обновления и удаления строк таблиц. Если в базе данных определен некоторый триггер, то он запускается автоматически всегда при возникновении события. Важным является то, что пользователи не могут обойти триггер, независимо от того, кто из них и каким образом инициировал запускающее его событие. Основным назначением триггеров является автоматическая поддержка целостности базы данных, но они могут использоваться и для реализации достаточно сложных ограничений, накладываемых предметной областью, например, с операцией вставки нового товара в накладную может быть связан триггер, который проверяет наличие необходимого товара на складе и выполняет другие необходимые действия.

#### **4.12. Перспективы развития баз данных**

Современные информационные системы основываются на базах данных, в которых накапливается ценная информация. Поэтому сейчас разрабатываются и широко распространяются методы обработки баз данных с целью извлечения из них дополнительных знаний, методов. Эти знания и методы связаны с обобщением и различными дополнительными способами обработки данных. В этом случае базы данных служат хранилищами информации, это направление называется «Хранилища данных» (*Data Warehouse*) [15].

Для работы с «Хранилищами данных» используется интеллектуаль-

ный анализ данных (ИАД), или *data mining*, – это процесс выявления значимых корреляций, образцов и тенденций в больших объемах данных. Это важнейшая современная информационная технология, учитывая высокие темпы роста объемов накопленной в современных хранилищах данных. Ее активно используют как крупные корпорации, так и более мелкие фирмы, которые серьезно относятся к вопросам анализа и прогнозирования своей деятельности. Особенно широко методы ИАД применяются в бизнес-приложениях аналитиками и руководителями компаний.

В бизнес-приложениях наибольший интерес представляет интеграция методов интеллектуального анализа данных с технологией оперативной аналитической обработки данных (On-Line Analytical Processing, **OLAP**). Это обобщенный термин, характеризующий принципы построения систем поддержки принятия решений – Decision Support System (DSS), хранилищ данных – Data Warehouse, систем интеллектуального анализа данных – Data Mining. Такие системы предназначены для нахождения зависимостей между данными, для проведения динамического анализа по принципу «что если...» и тому подобных задач. OLAP-приложения оперируют с большими массивами данных, накопленными на предприятии или взятыми из других источников. Такие системы характеризуются следующими признаками:

- добавление в систему новых данных происходит относительно редко крупными блоками, например, один раз в месяц или квартал;
- данные, добавленные в систему, как правило, никогда не удаляются;
- перед загрузкой данные проходят различные подготовительные процедуры, связанные с приведением их к определенным форматам и тому подобное;
- запросы к системе являются нерегламентированными и достаточно сложными;
- скорость выполнения запросов важна, но не критична.

Базы данных OLAP-приложений обычно представлены в виде одного или нескольких гиперкубов, измерения которого представляют собой справочные данные, а в ячейках самого гиперкуба хранятся значения этих данных. Физически гиперкуб может быть построен на основе специальной многомерной модели данных – Multidimensional OLAP (MOLAP) или представлен средствами реляционной модели данных – Relational OLAP (ROLAP).

Сильно нормализованные модели данных хорошо подходят для OLTP-приложений – On-Line Transaction Processing (**OLTP**) – приложений оперативной обработки транзакций. Типичными примерами OLTP-приложений являются системы складского учета, заказов билетов, операционные банковские системы и другие. Основная функция подобных систем заключается в выполнении большого количества коротких транзакций.

Следующим направлением развития баз данных является появление

так называемых **темпоральных** баз данных, то есть баз данных, чувствительных ко времени. Фактически БД моделирует состояние объектов предметной области в некоторый текущий момент времени. Однако в ряде прикладных областей необходимо исследовать именно изменение состояний объектов во времени.

Еще одним из перспективных направлений развития баз данных является направление, связанное с объединением технологии экспертных систем и баз данных, и развитие так называемых **дедуктивных** баз данных. Эти базы основаны на извлечении новых знаний из баз данных не путем запросов или аналитической обработки, а путем использования правил вывода и построения цепочек применения этих правил для вывода ответов на запросы. Для этих баз данных существуют языки запросов, отличные от классического SQL. В экспертных системах также знания экспертов хранятся в форме правил, чаще всего используются так называемые продукционные правила типа «если описание ситуации, то описание действия».

И, наконец, последним, но, может быть, самым значительным направлением развития баз данных является перспектива взаимодействия Web-технологии и баз данных. Простота и доступность Web-технологии, возможность свободной публикации информации в Интернете, так чтобы она была доступна любому количеству пользователей, несомненно, сразу завоевали авторитет у большого числа пользователей.

### *Контрольные вопросы*

1. Дайте определение понятие «информация».
2. Дайте определение понятие «данные».
3. Каково назначение банка данных?
4. Каково назначение банка знаний?
5. Опишите структуру банка данных.
6. Как можно обеспечить надежность хранения данных?
7. Назовите проблемы создания БД.
8. Что такое предметная область в информационных системах?
9. Какие этапы проектирования необходимо выполнить при создании БД?
10. Перечислите модели данных.
11. Что называется СУБД?
12. Каковы функция и состав универсальной СУБД?
13. Опишите перспективы развития баз данных.



## 5. ИСПОЛЬЗОВАНИЕ КОМПЬЮТЕРНОЙ ГРАФИКИ

---

### 5.1. Задачи компьютерной графики

Задача компьютерной графики сводится к созданию изображений по их описанию. **Предмет компьютерной графики** – изучение способов построения, хранения, обработки, описания моделей, объектов и их изображений с помощью ЭВМ. В том случае, если пользователь может управлять характеристиками объектов, говорят об **интерактивной** компьютерной графике.

При обработке информации, связанной с изображением на мониторе, принято выделять три основных направления:

- распознавание образов;
- обработку изображений;
- машинную графику.

Основная задача распознавания образов состоит в преобразовании уже имеющегося изображения на формально понятный язык символов. **Распознавание образов** или система технического зрения (COMPUTER VISION) – совокупность методов, позволяющих получить описание изображения, поданного на вход, либо отнести заданное изображение к некоторому классу (так поступают, например, при сортировке почты). Одной из задач COMPUTER VISION является так называемая скелетизация объектов, при которой восстанавливается некая основа объекта, его «скелет».

**Обработка изображений** (IMAGE PROCESSING) рассматривает задачи, в которых и входные, и выходные данные являются изображениями. Например, передача изображения с устранением шумов и сжатием данных, переход от одного вида изображения к другому (от цветного к черно-белому) и т.д.

**Компьютерная (машинная) графика** (COMPUTER GRAPHICS) воспроизводит изображение в случае, когда исходной является информация неизобразительной природы. Например, визуализация экспериментальных данных в виде графиков, гистограмм или диаграмм, вывод информации на экран компьютерных игр, синтез сцен на тренажерах.

### 5.2. Приложения компьютерной графики

Компьютерная графика стала основным средством взаимодействия человека с ЭВМ. Важнейшими сформировавшимися областями приложений являются:

- компьютерное моделирование, которое явилось исторически первым широким применением компьютерной графики;
- системы автоматизации научных исследований, системы автоматизации проектирования, системы автоматизации конструирования, системы автоматизации производства, автоматизированные системы управления технологическими процессами;
- бизнес;
- искусство;
- средства массовой информации;
- досуг.

В настоящее время появилось новое, очень интересное приложение компьютерной графики – виртуальная реальность.

Исторически первыми интерактивными системами считаются системы автоматизированного проектирования (САПР), которые появились в 60-х годах. Они представляют собой значительный этап в эволюции компьютеров и программного обеспечения. В системе интерактивной компьютерной графики пользователь воспринимает на дисплее изображение, представляющее некоторый сложный объект, и может вносить изменения в описание (модель) объекта. Такими изменениями могут быть как ввод и редактирование отдельных элементов, так и задание числовых значений для любых параметров, а также иные операции по вводу информации на основе восприятия изображений.

Системы типа САПР активно используются во многих областях, например в машиностроении и электронике. Одними из первых были созданы САПР для проектирования самолетов, автомобилей, системы для разработки микросхем интегральных схем, архитектурные системы. Такие системы на первых порах функционировали на достаточно больших компьютерах. Потом распространилось использование быстродействующих компьютеров среднего класса с развитыми графическими возможностями – **графических рабочих станций**. С ростом мощностей персональных компьютеров все чаще САПР использовали на дешевых массовых компьютерах, которые сейчас имеют достаточные быстродействие и объемы памяти для решения многих задач. Это привело к широкому распространению систем САПР.

### 5.3. Технические средства компьютерной графики

Дисплейные устройства, разработанные в середине 60-х годов и используемые до сих пор, называются **векторными**, **штриховыми** или **каллиграфическими**. Они состоят из дисплейного процессора, дисплейной буферной памяти и ЭЛТ с относящейся к ней электронной частью. Буфер служит для запоминания подготовленного на ЭВМ дисплейного списка

(или дисплейной программы); последняя включает команды вывода точек и отрезков (по координатам конечных точек), а также команды вывода литер.

Команды рисования точек, отрезков и литер интерпретируются дисплейным процессором, который преобразует цифровые значения в аналоговые напряжения, управляющие электронным лучом, последний вычерчивает линии на люминофорном покрытии ЭЛТ. Поскольку светоотдача люминофора падает до 0 за десятки микросекунд, дисплейный процессор должен осуществлять цикл по заданной программе с целью регенерации изображения на люминофоре с частотой не менее 30 раз в 1 с для устранения мерцания; в связи с этим буфер, в котором хранится дисплейная программа, обычно называется **буфером регенерации**.

В середине 70-х годов была изобретена дешевая **растровая** графика, основанная на телевизионной технике, которая оказала сильное влияние на развитие машинной графики. В растровых дисплеях дисплейные примитивы (отрезки, литеры и закрашенные участки – обычно многоугольники) хранятся в памяти для регенерации в виде совокупности образующих их точек, называемых **пикселями** (pixels) или **пэлами** (pels) от словосочетания **picture element**. Изображение формируется на **растре**, представляющем собой совокупность горизонтальных растровых строк, каждая из которых состоит из отдельных пэлов; таким образом, растр – это матрица из пэлов, покрывающая всю площадь экрана. Все изображение последовательно сканируется 30 раз в 1 с по отдельным строкам растра в направлении сверху вниз, при этом изменяется лишь интенсивность электронного луча для каждого пэла в строке. Благодаря этому резко возрастает потребность в памяти.

**Отличие растровой графики от векторной.** Растровая графика (в отличие от векторной) позволяет закрашивать отдельные участки изображения (обычно разными цветами), что существенно повышает возможности передачи информации. Кроме того, процесс регенерации в этом случае не зависит от сложности изображения (число линий и т. д.), так как аппаратура обладает достаточным быстродействием, чтобы считать все пэлы из буферной памяти в течение цикла регенерации независимо от того, содержат они информацию или же служат лишь фоном. Благодаря этому устраняется мерцание. Векторные же дисплеи часто начинают мерцать, когда число примитивов в буфере становится большим.

Чтобы переместить часть изображения в векторном дисплее, нужно скорректировать параметры программы, в растровом нужно переместить блок данных из одного места в другое. В растровом дисплее можно закрасить область, а в векторном – только заштриховать.

**Аппаратными компонентами** являются основная ЭВМ и управляемое ею **дисплейное устройство**, называемое дисплеем или **графическим терминалом**. Дисплей состоит из компоненты для **вывода** (**дисплейный экран** или видовая поверхность, на которую выводится изображение) и

компоненты для ввода или интерактивной компоненты. Интерактивная компонента обычно представляет собой набор логических устройств, которые содержат:

- устройства типа **клавиатуры**;
- **кнопочное устройство** для вызова заданных функций или осуществления определенных операций;
- устройства для указания (**световое перо**), используемые при указании составных частей изображения на экране;
- **валюаторы**, т.е. устройства типа вращающихся ручек или рычагов для ввода скалярных величин;
- указатели положения по осям  $x$  и  $y$ , или **локаторы**.

**Программное обеспечение** состоит из трех компонент. Первой компонентой является **прикладная программа**. Она засылает информацию и извлекает ее из второй компоненты – **прикладной структуры данных** (базы данных) – и направляет графические команды в третью компоненту – **графическую систему**.

Структура данных содержит описание реальных или абстрактных объектов, изображения которых должны появляться на экране. Поэтому в структуре данных может храниться необходимая информация для таких разнообразных объектов, как электрические схемы, здания, математические и статистические функции и т. д.

В описания объектов обычно включаются **геометрические** данные о координатах (определяющее форму объекта), **атрибуты** объекта (тип линии, цвет или фактура поверхности), а также данные о связности и положении (т.е. данные о том, каким образом компоненты соединены между собой).

**Графический пакет (графическая схема)** – получает набор команд от пользователя и преобразует их в изображение. Графическая схема по назначению эквивалентна подсистемам ввода/вывода. **Прикладная структура данных** хранит описание объекта, т.е. информацию о геометрии объекта, связности объекта, об атрибутах объекта и характеристики объекта. **Прикладная программа** обеспечивает преобразование вывода, т.е. элементов структуры данных в поток примитивов и преобразование ввода, т.е. поток сообщений об изменении структуры данных.

#### 5.4. Графические файлы и их форматы

Наиболее распространенными форматами графических файлов являются:

**ВМР (Windows Device Independent Bitmap)**

Формат ВМР является «родным» форматом Windows, он поддерживается всеми графическими редакторами, работающими под ее управлением.

ем. Применяется для хранения растровых изображений, предназначенных для использования в Windows. Способен хранить как индексированный (до 256 цветов), так и RGB-цвет (16 млн оттенков). Возможно применение сжатия по принципу RLE (они могут иметь расширение .rle).

#### **WMF (Windows Metafile)**

Еще один «родной» формат Windows. Служит для передачи векторов через буфер обмена (Clipboard). Понимается практически всеми программами Windows, так или иначе связанными с векторной графикой. Однако, несмотря на кажущуюся простоту и универсальность, пользоваться форматом WMF стоит только в крайних случаях для передачи «голых» векторов. WMF искажает цвет, не может сохранять ряд параметров, которые могут быть присвоены объектам в различных векторных редакторах.

#### **GIF (CompuServe Graphics Interchange Format)**

Разработан фирмой CompuServe для передачи растровых изображений по сетям. Он использует LZW-компрессию, что позволяет хорошо сжимать файлы, в которых много однородных заливок (логотипы, надписи, схемы). GIF-формат позволяет записывать изображение «через строчку» (Interlaced), благодаря чему, имея только часть файла, можно увидеть изображение целиком, но с меньшим разрешением. Эта возможность широко применяется в Интернете. Сначала вы видите картинку с грубым разрешением, а по мере поступления новых данных ее качество улучшается. В GIF'e можно назначить один или более цветов прозрачными, они станут невидимыми в интернетовских браузерах и некоторых других программах. Прозрачность обеспечивается за счет дополнительного Alpha-канала, сохраняемого вместе с файлом.

Кроме того, файл GIF может содержать не одну, а несколько растровых картинок, которые интернетовские браузеры могут подгружать одну за другой с указанной в файле частотой. Это называется GIF-анимация.

Основное ограничение формата GIF состоит в том, что цветное изображение может быть записано только в режиме 256 цветов. Для полиграфии этого явно недостаточно.

#### **PNG (Portable Network Graphics)**

PNG – недавно разработанный формат для Сети, призванный заменить собой устаревший GIF. Использует сжатие без потерь. Глубина цвета может быть любой, вплоть до 48 бит (RGB, для сравнения, – 24), используется Interlacing, причем не только строк, но и столбцов, поддерживается плавно переходящая прозрачность. В файл формата PNG записывается информация о гамма-коррекции. Гамма представляет собой некое число, характеризующее зависимость яркости свечения экрана вашего монитора от напряжения на электродах кинескопа. Это число, считанное из файла, позволяет ввести поправку яркости при отображении. Нужно оно для того, чтобы картинка, созданная на Макинтош, выглядела одинаково и на PC и на Silicon Graphics. Таким образом, эта особенность помогает реализации

основной идеи WWW – одинакового отображения информации независимо от аппаратуры пользователя. Файлы PNG могут делать все основные графические редакторы.

### **TGA (Targa)**

«Targa» – это имя графического адаптера фирмы Truevision, который впервые использовал TGA-формат. Первая редакция TGA-формата имеет название «Original TGA format» (оригинальный TGA-формат), а вторая – «New TGA Format» (новый TGA-формат). Формат может хранить изображения с глубиной цвета до 32 бит. Наряду со стандартными тремя RGB-каналами TGA-файл имеет дополнительный альфа-канал для представления информации о прозрачности изображения. Информация может быть сжата. Формат используется программными продуктами многих известных в мире компьютерной графики фирм.

### **JPEG (Joint Photographic Experts Group)**

Строго говоря, JPEG'ом называется не формат, а алгоритм сжатия, основанный не на поиске одинаковых элементов, как в RLE и LZW, а на разнице между пикселями. JPEG ищет плавные цветовые переходы. Вместо действительных значений JPEG хранит скорость изменения от пикселя к пикселу. Лишнюю, с его точки зрения, цветовую информацию он отбрасывает, усредняя некоторые значения. Можно задать уровень компрессии. Чем выше уровень компрессии, тем больше данных отбрасывается и тем ниже качество. Используя JPEG, можно получить файл в 10-500 раз меньше, чем BMP. Формат аппаратно независим. Из сказанного можно сделать следующий вывод: JPEG'ом лучше сжимаются растровые картинки фотографического качества, чем логотипы или схемы – в них больше полутонных переходов, среди же однотонных заливок появляются нежелательные помехи. В JPEG'e следует сохранять только конечный вариант работы, потому что каждое пересохранение приводит к все новым потерям (отбрасыванию) данных и превращению исходного изображения в кашу.

### **TIFF (Tagged Image File Format)**

Аппаратно независимый формат TIFF на сегодняшний день является одним из самых распространенных и надежных, его поддерживают практически все программы на PC и Macintosh, так или иначе связанные с графикой. TIFF является лучшим выбором при импорте растровой графики в векторные программы и издательские системы. Ему доступен весь диапазон цветовых моделей от монохромной до RGB, CMYK и дополнительных цветов Pantone. TIFF может сохранять векторы Photoshop'a, Alpha-каналы для создания масок в видеоклипах Adobe Premiere и массу других дополнительных данных. Наибольшие проблемы обычно вызывает LZW-компрессия, иногда применяемая в TIFF'e. Ряд программ (например, QuarkXPress 3.x и Adobe Streamline) не умеют читать такие файлы, кроме того, они могут дольше выводиться на принтеры и фотонаборные автоматы. Только если файл компрессуется в 3-4 раза, получается выигрыш во времени вывода.

### **PSD (Adobe Photoshop Document)**

PSD – «родной» формат популярного растрового редактора Photoshop. Он позволяет записывать изображение со многими слоями, их масками, дополнительными каналами, контурами и другой информацией – все, что может сделать Photoshop. Начиная с версии 3.0, используется RLE-компрессия, в 4-й версии файлы становятся еще меньше. PSD понимают некоторые программы.

### **CDR (CorelDRAW Document)**

Формат известен в прошлом низкой устойчивостью и плохой совместимостью файлов, тем не менее, пользоваться CorelDRAW чрезвычайно удобно, он имеет неоспоримое лидерство на платформе PC. Многие программы на PC (FreeHand, Illustrator, PageMaker, ...) могут импортировать файлы CDR. Версии CorelDRAW, начиная с 7-й можно без натяжек назвать профессиональными. В файлах этих версий применяется компрессия для векторов и раstra отдельно, могут внедряться шрифты, файлы CDR имеют огромное рабочее поле 45×45 м (этот параметр важен для наружной рекламы); начиная с 4-й версии, поддерживается многостраничность.

## **5.5. Система автоматизированного проектирования – AutoCAD**

AutoCAD – самая популярная в мире система автоматизированного проектирования и выпуска рабочей конструкторской и проектной документации. С помощью AutoCAD создаются двумерные и трехмерные проекты различной степени сложности в области архитектуры и строительства, машиностроения, генплана, геодезии и т.д. Формат хранения данных AutoCAD признан международным стандартом хранения и передачи проектной документации. AutoCAD является платформой, на которой построено множество специализированных программ, имеющих общий формат хранения данных. Высокопрофессиональные приложения от авторизованных разработчиков Autodesk дополняются утилитами и программами, которые (с помощью встроенных языков программирования) создают сами пользователи.

На сегодня AutoCAD охватывает весь спектр инженерных задач: создание трехмерных моделей, разработку и оформление чертежей, выполнение различного рода расчетов, инженерный анализ, формирование фотореалистичных изображений готовой продукции.

### **5.5.1. Модули AutoCAD**

Оптимальная система автоматизированного проектирования для выполнения двумерных чертежей и создания проектной документации **Auto-**

**CAD LT** – эффективное и недорогое решение для оборудования рабочих мест, где не требуется создавать трехмерные объекты. Продуманный интуитивный интерфейс, развитые инструменты черчения, редактирования и вывода на печать позволят быстро освоиться с системой и приступить к разработке проектов. Удобный и простой в использовании, AutoCAD LT существенно ускорит выполнение машиностроительных, архитектурных и электротехнических чертежей, планов, схем, эскизов. На базе AutoCAD LT построен ряд готовых решений для машиностроения, строительства и обработки сканированной технической документации. При работе со сканированной технической документацией в среде AutoCAD повышается качество раstra, выполняется гибридное редактирование, векторизация.

**Autodesk Land Desktop** – это базовая система автоматизированного проектирования для решения задач изысканий, картографии, построения трехмерных моделей, генерального плана, кадастра, проектирования площадных и линейных объектов.

Комплексная система, позволяющая вводить и рассчитывать данные, полученные в ходе проведения инженерно-геологических изысканий, строить графические зависимости, производить обработку и интерпретацию результатов лабораторных испытаний и статического зондирования грунтов, выполнять построение инженерно-геологических разрезов и инженерно-геологических колонок, а также формировать отчетную документацию, соответствующую государственным стандартам стран СНГ.

**GeoniCS Инженерная геология (GeoDirect)**. Система GeoniCS Инженерная геология предназначена для комплексной обработки данных инженерно-геологических изысканий. Программа выполняет следующие виды расчетов: расчет физико-механических, прочностных и деформационных характеристик грунтов, статистическая обработка информации по выделенным инженерно-геологическим элементам (ИГЭ); расчет предельных сопротивлений и несущей способности свай.

Инструменты системы обеспечивают построение инженерно-геологического разреза по группе выработок. Нанесение графиков изменения характеристик грунтов (в том числе и результатов статического зондирования) осуществляется по указанию пользователя. В программе реализовано построение инженерно-геологических колонок с автоматической штриховкой грунта по ИГЭ. Предусмотрена возможность задания пользовательских значений параметров штриховки и цветовых характеристик.

**SurvCADD** – модульная система на основе AutoCAD для проектирования и разработки шахт и карьеров. Программный комплекс SurvCADD XML предназначен для проектирования объектов горнорудной промышленности: подземных шахт, открытых карьеров, дорог, дамб и котлованов. В едином комплексе решаются задачи обработки данных геодезических изысканий, геологического моделирования и анализа месторождений. SurvCADD поддерживает обмен данными с другими системами проекти-



рования через формат LandXML. Работает в среде AutoCAD 2002, Autodesk Map 5 или Autodesk Land Desktop 3.

Сферы применения:

- инженерно-геодезические и инженерно-геологические изыскания;
- цифровые модели рельефа и объемы;
- скважины и разрезы;
- анализ месторождений;
- проектирование рудников.

**Модули SurvCADD XML.** Назначение модуля SurvCADD COGO-Design – сбор и обработка данных инженерно-геодезических изысканий, а также создание планов местности. Средствами модуля возможен прямой ввод/вывод с электронных приборов TDS, Leica, Nikon DTM700, Nikon DTM300, SDR/Sokkia, Geodimetr, FOP-CON FC-1, FC-4, MDL Laser, Zeiss, SurvStar, Dozer2000. Предусмотрен также ввод данных из полевых журналов. Результаты разных измерений показываются на экране разным цветом, что позволяет быстро обнаружить ошибки. Введенные данные уравниваются, выводятся в рисунок и заносятся в базу данных. Объекты съемки можно вычерчивать автоматически. Модуль включает полный набор команд, необходимых для выполнения горизонтальной планировки участка: работу с базовой геометрией, вставку символов, подсчет площадей, размещение осевых линий дорог, пересечений, тупиков, границ участков и других элементов плана.

Модуль SurvCADD Contour-DTM предназначен для построения цифровой трехмерной модели местности, анализа рельефа и проектирования поверхностей. При построении поверхности рельефа используются различные алгоритмы, в том числе методы триангуляции, кригинга, наименьших квадратов, обратных расстояний, полиномиальный. Позволяет выбрать метод моделирования рельефа и в зависимости от исходной информации создать модель, которая наиболее точно описывает поверхность. В системе можно работать с любым количеством поверхностей. SurvCADD Contour-DTM включает средства для работы с горизонталями (вывод, сглаживание, разметка), анализа поверхности по диапазонам высот и уклонов, получения информации о водостоках и водоразделах. Предлагается полный набор средств для проектирования площадок, дамб, котлованов, плотин, подсчета объемов земляных работ, построения картограммы земляных работ и создания отчетов. Визуализация спроектированной поверхности может быть выполнена средствами, включенными в состав модуля.

Модуль SurvCADD Section-Profile необходим при построении профилей любых линейных объектов, в том числе инженерных коммуникаций. Для построения профилей используются данные с цифровых моделей рельефа. Базовые функции модуля позволяют проектировать автомобильные дороги и работать с поперечным профилями плана трассы. Преду-

смотрены удобные средства черчения, редактирования, а также функции для подсчета объемов.

Модули **SurvCADD Mining** и **SurvCADD Advanced Mining** предназначены для ввода данных геологических изысканий, моделирования анализа месторождения, проектирования открытых карьеров и подземных шахт.

Для описания геологии система оперирует понятиями «слой», «скважина», «сетка слоя». Количество слоев не ограничено, а их чередование произвольно. С каждым слоем связаны атрибуты, определяющие геологические характеристики слоя. Значения атрибутов задаются при вводе данных опробования по скважинам и используются для анализа месторождения. Скважины могут различаться по типам, быть как вертикальными, так и наклонными. Для каждого типа задаются обозначение скважины и дополнительные атрибуты. Высота устья скважины задается при вводе или берется из ранее определенной поверхности.

Добавление скважины в модель производится либо вручную, либо автоматически, из ранее подготовленных файлов. Файл может быть подготовлен в одном из четырнадцати международных форматов описания скважин или в формате пользователя с помощью различных текстовых редакторов. Модуль включает команды для быстрого поиска ошибок и коррекции введенной информации. Для каждого слоя строится сеть. Набор сетей представляет геологическую модель месторождения. Автоматически определяются выклинивания и выходы пород на поверхность. Результаты моделирования выводятся графически (разрезы, блок-диаграммы, изолинии) и в виде отчетов различной формы. Можно провести качественный анализ месторождения и подсчитать запасы как для месторождения в целом, так и по отдельным участкам, уступам карьера, пластам. Результаты анализа выводятся в виде отчетов, форматы которых создаются пользователем. Набор команд, предусмотренный в модулях **SurvCADD Mining** и **SurvCADD Advanced Mining**, удобен и позволяет значительно повысить производительность.

**SurvCADD Mining** и **SurvCADD Advanced Mining** позволяют построить календарный план и графики использования оборудования. Предварительно заполняется база данных оборудования и определяется график его работы (количество смен, праздники и т.п.). Результат планирования, помимо отчетов, выводится и графически в виде закрашенных пятен, наглядно показывающих распределение времени добычи руды и эффективность использования оборудования.

Модуль **SurvCADD Hydrology** предназначен для выполнения расчетов по анализу гидрологической ситуации на участке, а также проектирования водозащитных сооружений. Включает команды для расчета ливневых стоков (с учетом данных по интенсивности осадков и характеристикам почв), построения линий водоразделов, направления стоков, гидрографов, кривых наполняемости прудов. Предлагаются эффективные средства для

проектирования дренажной канализации и прудов различного типа: по шаблону, по существующим горизонталям, с уступами.

**Autodesk Civil Design** – дополнительный модуль к Autodesk Land Desktop, предназначенный для решения задач проектирования гражданских объектов, позволяет в едином программном окружении автоматизировать работы по плановой геометрии осей сооружений, продольным и поперечным профилям, проектированию и учету естественных преград и искусственных сооружений. Применение Autodesk Civil Design предоставляет возможность оперативно произвести необходимые инженерные расчеты и выпустить рабочую проектную документацию. Использование комплекса при коллективной работе над проектом ускоряет выполнение и повышает качество работ проектной группы и проектных институтов в целом.

Познакомимся с пользовательским интерфейсом AutoCAD и основными принципами создания графических документов.

### **5.5.2. Общие принципы работы в AutoCAD**

При первом запуске рабочий стол AutoCAD настроен по умолчанию.

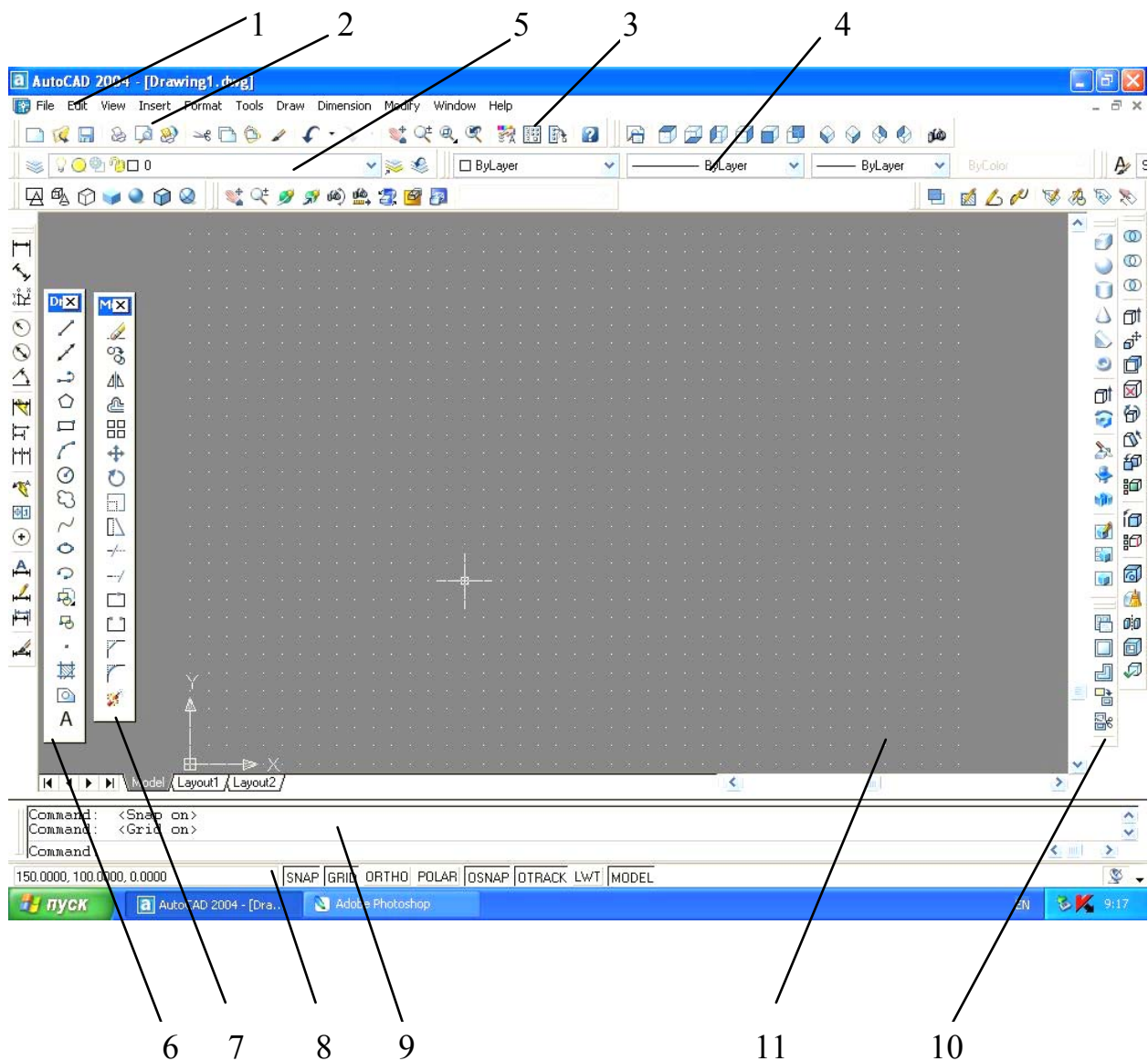
В рабочий стол AutoCAD включены (рис.5.1):

- *падающее меню* (1) – самая верхняя строка меню;
- *необязательные панели инструментов*:
  - *стандартная Standard* (2) и *панель стилей Styles* (3) – вторая строка от заголовка;
  - *свойства объектов Properties* (4) и *панель слоев Layers* (5) – третья строка;
  - *панели рисования Draw* (6) и *редактирования Modify* (7) – столбцы слева или справа;
- *строка состояния* (8) – нижняя строка;
- *окно командных строк* (9) – выше строки состояния;
- *необязательное экранное меню* (10) – столбец справа;
- *графическое поле* (11), занимающее остальную область рабочего стола.

Программа AutoCAD по умолчанию записывает внутреннее представление рисунка в файл с расширением .dwg.

Создать рисунок позволяет команда File →New... После запуска команды необходимые настройки рабочей среды производятся в диалоговом окне Great New Drawing.

AutoCAD позволяет создавать и редактировать чертеж множеством различных способов. Существует множество команд редактирования чертежа: любую часть чертежа можно стирать, перемещать, «размножать», растягивать и пр.



**Рис. 5.1. Интерфейс программы AutoCAD**

Значения координат всегда связаны с некоторой системой координат. По умолчанию в AutoCAD используется Мировая Система Координат (МСК). Она определена так, что ось  $OX$  направлена слева направо, ось  $OY$  – снизу вверх, ось  $OZ$  – перпендикулярна экрану во вне. Для удобства работы может быть определена Пользовательская система координат (ПСК), которую можно сместить относительно мировой, можно повернуть под любым углом. Работа в ПСК сопровождается изменением на экране пиктограммы системы координат. Таких систем координат может быть несколько, и в любой момент возможен переход из одной в другую.

Для управления сложными командами или группой связанных между собой команд служат диалоговые окна. Диалоговые окна позволяют выполнять различные установки и предоставляют возможность предварительного просмотра результатов. Из большинства диалоговых окон можно

выйти с сохранением сделанных установок, щелкнув мышью на командной кнопке ОК. Для выхода из диалогового окна без сохранения изменений достаточно указать мышью на командную кнопку «Отмена» или нажать на клавиатуре клавишу Esc.

Командная строка (рис. 5.1, поз. 9) имеет свое отражение в текстовом окне. Вызывается и убирается она с экрана клавишей (F2). Практически текстовое окно является протоколом данных, введенных в командной строке.

Текстовое окно подобно окну команд, в нем также можно вводить команды и наблюдать подсказки и сообщения, выдаваемые AutoCAD. Содержимое текстового окна предназначено только для чтения и не может быть изменено. Но протокол команд можно копировать для последующей вставки в командную строку или текстовый редактор.

В AutoCAD 2005 сохраняется возможность отображения экранного меню. Чтобы задать его отображение, выберите в меню «Инструменты» команду «Опции...». В результате появится окно «Параметры». Перейдите на вкладку «Экран» этого окна и отметьте флажок «Отобразить экранное меню».

Экранное меню AutoCAD имеет иерархическую структуру. Используя переходы в различные подменю, можно передвигаться по дереву меню. Для перехода в корень этого дерева служит верхняя строка экранного меню, в которой всегда находится слово AutoCAD. В этом меню нет необходимости, если вы предпочитаете вводить команды при помощи панелей инструментов, меню или клавиатуры.

Контекстное меню – это удобный инструмент, помогающий выбрать следующий шаг работы. Оно вызывается при нажатии правой кнопки мыши практически в любой момент времени. Содержание контекстного меню зависит от выполняемой в данный момент команды. Существует пять основных видов контекстного меню:

1. Контекстное меню по умолчанию. Открывается после нажатия правой кнопки мыши в области чертежа.

2. Контекстное меню режима редактирования. Открывается после выбора какого-либо объекта и нажатия правой кнопки мыши.

3. Контекстное меню командного режима. Открывается при нажатии правой кнопки мыши в командной строке. В меню отображаются опции этой команды.

4. Контекстное меню диалогового режима. Открывается при нажатии правой кнопки мыши в поле диалогового окна.

5. Служебное контекстное меню. Открывается при нажатии правой кнопки мыши в командной строке (показывает список из последних семи команд).

### 5.5.3. Создание изображений

Рисунки строятся из набора геометрических примитивов, под которыми понимается элемент чертежа, обрабатываемый системой как целое, а не совокупность точек и объектов. Графические примитивы создаются командами вычерчивания или рисования, которые вызываются из падающего меню «Draw» или одноименной панели инструментов. Для окончания команды используется клавиша Enter или правая кнопка мыши.

К основным *геометрическим примитивам* относятся:

**ТОЧКА (POINT)**. Самый простой геометрический объект. Может быть использован для построений новых объектов с помощью функции объектной привязки УЗЕЛ, а сами точки поэтому очень часто называются узловыми точками. Команда Точка определяется указанием ее координат. Местоположение точки можно указать щелчком мыши.

Запрос команды POINT

Current point modes: PDMODE=33PDSIZE=10.0

(текущие режимы точек: PDMODE= 33 PDSIZE=10.0)

Specify a point

(Укажите точку:).

На экране точка может отображаться различными знаками (их около 20). Стиль представления объекта ТОЧКА определяется в диалоговом окне «Стиль точки», вызываемое из меню Формат командой Стиль точек... Размер ее можно задать в этом же окне относительно размера экрана либо в относительных единицах.

**ОТРЕЗОК (LINE)** – это простой базовый примитив. Он может быть одиночным или объединяться конечными точками в ломаную линию, каждый сегмент которой является отдельным отрезком.

Запросы команды LINE:

Specify first point: – начало отрезка

(Первая точка отрезка:)

Specify next point or [Undo]: – конец отрезка

(Следующая точка или [Отменить]:)

Specify next point or [Close/Undo]:

(Следующая точка или [Замкнуть/Отменить]:).

Курсор устанавливается в нужное положение и нажимается левая кнопка мыши. При перемещении к каждой следующей точке за перекрестием тянется «резиновая нить». Это позволяет отслеживать положение каждого следующего отрезка ломаной линии.

**ПРЯМАЯ (XLINE)** – это бесконечная в обе стороны линия. Она служит для проведения вспомогательных линий, которые могут использоваться, например, в качестве линий связи между проекциями детали.

Запросы команды XLINE:

Specify a point or [Hor/Ver/Ang/Bisect/Offset]:

(Укажите точку или [горизонтальная/ вертикальная/ под углом/ биссектриса/ смещение]:)

Specify through point:

(Через точку:)

Команда ПРЯМАЯ имеет пять опций:

Hor – для построения горизонтальных линий;

Ver – для построения вертикальных линий;

Ang – для построения наклонных линий под определенным углом, значение которого задается на соответствующий запрос;

Bisect – для построения биссектрисы угла по его вершинам и двум точкам, расположенным на сторонах угла;

Offset – для построения прямой параллельно любому указанному отрезку на заданном расстоянии или через заданную точку.

**ЛУЧ (RAY)** – примитив, начинающийся в некоторой точке и бесконечный в одну сторону.

Запросы:

Specify start point:

(Начальная точка:)

Specify through point:

(Через точку:)

**ОКРУЖНОСТЬ (CIRCLE)**. По умолчанию окружность строится путем задания центра и радиуса. Однако окружность можно строить различными способами.

Запросы:

Specify center point for circle or [3P/2P/Tr (tan tan radius)]:

(Центр круга или [3Т/2Т/ККР (касательная, касательная, радиус)]:

Specify radius of circle or [Diameter]:

(Радиус круга или [Диаметр]:)

Опции команды КРУГ:

3P (3 точки) – строит окружность по трем точкам, лежащим на ней;

2P (2 точки) – строит окружность по двум точкам, лежащим на диаметре;

Tr (Касательная, касательная, радиус) – строит окружность, касающуюся двух объектов и по радиусу;

tan tan radius (Касательная, касательная, радиус) – строит окружность, касающуюся трех объектов.

**ДУГА (ARC)** – примитив, являющийся частью окружности.

Команда дуга обеспечивает одиннадцать способов создания дуги:

- Три точки (построение дуги по умолчанию) – построение дуги по трем лежащим на ней точкам;

- Начало, центр, конец – построение по начальной точке дуги, ее центру и конечной точке.
- Начало, центр, угол = дуга строится заданием начала, центра и величины центрального угла;
- Начало, Центр, длина – после ввода начальной точки и центра задается длина хорды, которая определяет конечную точку дуги. Дуга строится против часовой стрелки от начальной точки, причем по умолчанию строится меньшая из двух возможных дуг (та, что меньше  $180^\circ$ ). Если же вводится отрицательное значение длины хорды, то будет нарисована большая дуга;
- Начало, Конец, Угол – дуга строится против часовой стрелки, по введенной начальной точке дуги, ее конечной точке и центральному углу. Чтобы построить дугу по часовой стрелке, необходимо значение угла задавать со знаком минус.
- Начало, Конец, Направление – этот способ позволяет построить дугу, которая в начальной точке касается другого объекта;
- Начало, Конец, Радиус – AutoCAD всегда строит меньшую дугу против часовой стрелки. Если ввести отрицательное значение радиуса, будет построена большая дуга;
- Центр, Начало, Конец – дуга строится против часовой стрелки от начальной к конечной точке радиусом, вычисленным AutoCAD по расстоянию между центром и начальной точкой;
- Центр, Начало, Угол – построение дуги против часовой стрелки с последовательным вводом центра дуги, точки начала дуги и центрального угла.
- Центр, Начало, Длина – построение дуги аналогично построению по началу, центру и длине хорды, изменен лишь порядок ввода точек;
- Продолжить – построение дуги как продолжения предшествующей линии или дуги. При этом начальной точкой дуги и ее начальным направлением станут соответственно конечная точка и конечное направление последней созданной дуги или отрезка.

Опции команды ДУГА:

- Center (Центр) – точка центра дуги;
- End (Конец) – конечная точка дуги;
- Angle (Угол) – величина угла;
- Chord Length (Длина хорды) – длина хорды;
- Direction (Направление) – направление касательной;
- Radius (Радиус) – радиус дуги.

**Эллипс (ELLIPSE)** – геометрическое место точек, сумма расстояний до которых от двух фиксированных точек постоянна.

Имеется возможность строить как математически точные эллипсы, так и эллиптические дуги, представляющие часть эллипса. Основными па-



раметрами эллипса являются координаты центра, направление и размер большой и малой осей.

Запросы:

Specify axis endpoint for ellipse or [Arc/Center]:

(Конечная точка оси эллипса или [дуга/Центр]:)

Specify other endpoint of axis:

(Вторая конечная точка оси:)

Specify distance to other axis of [Rotation]:

(Длина другой оси или [Поворот]:)

Опции:

Center (Центр) – указание центра эллипса;

Arc (Дуга) – режим построения эллиптических дуг. По умолчанию эллиптические дуги, как и эллипсы, строятся указанием конечных точек первой оси и половины длины второй. После этого задаются начальный и конечный углы. Нулевым углом здесь считается направление от центра эллипса вдоль его большой оси. Если начальный и конечный углы совпадают, строится полный эллипс. Вместо задания конечного угла можно указать центральный угол дуги, измеренный от начальной точки.

**СПЛАЙН (SPLINE)** – сглаженная кривая, которая проходит через заданные точки. Сплайны применяются там, где необходимо построение кривых произвольной формы.

Запросы:

Specify first point or [Object]:

(Первая точка или [Объект]:)

Specify next point:

(Следующая точка:)

Specify next point or [Close/Fit tolerance]<start tangent>:

(Следующая точка или [Замкнуть/Допуск]<касательная в начале>:).

Сплайны строятся путем задания координат точек и могут быть разомкнутыми и замкнутыми. Разомкнутые сплайновые кривые имеют разные точки начала и конца, с различными направлениями касательных в них. Направление касательных определяет изгиб сплайновой линии в ее начале и конце. У замкнутых сплайнов совпадают начальная и конечная точки и направления касательных в них.

При построении сплайновых кривых можно изменять допуск сплайновой аппроксимации – величину, определяющую насколько близко проходит сплайн к указанным пользователем точкам. Чем меньше значение допуска, тем сплайн ближе к определяющим точкам. При нулевом допуске сплайн проходит через определяющие точки.

**ПОЛИЛИНИЯ (PLINE)** – сложный примитив, состоящий из одного или нескольких связанных между собой прямолинейных и дуговых сегментов. Полилиния обрабатывается как единое целое.

Запросы:

Specify start point:

(Начальная точка:)

Current line-width is 0.0

(Текущая ширина полилинии равна 0.0)

Specify next point or [Arc/Close/Half width/Length/Undo/Width]:

(Следующая точка или [Дуга/ Замкнуть/ Полуширина/Длина/ Отменить/ Ширина]:)

Опции:

Arc – переводит команду в режим вычерчивания дуг;

Close – соединяет последнюю точку сегмента с первой. При этом происходит завершение работы команды ПОЛИЛИНИЯ и выход из нее;

Half width – позволяет задать полуширину, т.е. расстояние от осевой линии широкого сегмента до края;

Length – длина сегмента, созданного как продолжение предыдущего в том же направлении;

Undo – отменяет последний созданный сегмент;

Width – позволяет задать ширину последнего сегмента.

В любой момент времени можно от режима рисования прямолинейных сегментов перейти в режим рисования дуговых сегментов. В случае перехода в режим рисования дуг предлагается следующий выбор:

Current line-width is 0.0

Specify next point or [Arc/Close/Half width/Length/Undo/Width]:

(Следующая точка или [Дуга/ Замкнуть/ Полуширина/Длина/ Отменить/ Ширина]:)

Specify endpoint of arc or [Angle/ Center/ Close/ Direction/ Half width/ Line/ Radius/ Second pt/ Undo/ Width]:

(Конечная точка дуги или [Угол/ Центр/ Замкнуть/ Направление/ Полуширина/ Линейный/ Радиус/ Вторая/ Отменить/ Ширина]:).

В этот момент после указания конечной точки дуги строится дуговой сегмент, касающийся предыдущего участка полилинии.

Опции:

Angle – задание величины центрального угла для дугового сегмента;

Center – задание центра для дугового сегмента;

Close – замыкание полилинии с помощью дугового сегмента;

Direction – задание направления для построения дугового сегмента;

Half width – задание полуширины для следующего сегмента;

Line – переход в режим рисования прямолинейных сегментов;

Radius – задание радиуса для дугового сегмента;

Second pt – задание второй точки для построения дугового сегмента по трем точкам;

Undo – отмена последнего действия;

Width – задание ширины для следующего сегмента.

## **ПРЯМОУГОЛЬНИК (RECTANG)**

Начальный запрос команды:

Specify first corner point or [Chamfer/ Elevation/ Fillet/ Thickness/ Width]:

(Первый угол или [Фаска/ Уровень/ Скругление/ Высота/ Ширина]:)

Данная команда позволяет строить прямоугольник по двум противоположащим вершинам. Для задания вершин можно использовать любой способ ввода координат. Если какие-то установки необходимо сменить, нужно ввести их значения с помощью соответствующих опций.

Опции:

Chamfer – задание длин фаски, снимаемой в каждом углу прямоугольника;

Elevation – задание уровня для построения прямоугольника, смещенного по оси Z трехмерного пространства;

Fillet – задание радиуса скругления углов прямоугольника;

Thickness – задание высоты для построения прямоугольника, выдавленного вдоль оси Z трехмерного пространства;

Width – задание ширины полилинии, которая является строящимся прямоугольником.

**Многоугольник МН-УГОЛ (POLYGON)** – замкнутая геометрическая фигура с равными внутренними углами и со сторонами одинаковой длины. Допустимое количество сторон многоугольника от 3 до 1024. Так как длины сторон многоугольников всегда равны, с их помощью легко строить квадраты и встроенные треугольники.

Запрос:

Enter number of sides <default>:

(Число сторон <по умолчанию>:)

Specify Center of polygon or [Edge]:

(Укажите центр многоугольника или [Сторона]:)

Опции:

Edge – определяет положение любого ребра многоугольника по двум заданным точкам.

При указании центра многоугольника последует запрос:

Inscribed circle – формирует воображаемую окружность по заданному центру и радиусу, в которую будет вписан многоугольник.

Circumscribed about circle – формирует по заданному центру и радиусу воображаемую окружность, вокруг которой будет описан многоугольник.

**Кольцо (DONUT)** – часть плоскости, ограниченная двумя окружностями. С помощью функции построения колец легко построить закрашенные кольцеобразные объекты и круги. За один вызов команды можно построить любое число колец, имеющих одинаковые диаметры, но разные центры. Если требуется построить заполненный круг, задайте нулевой диаметр кольца.

Запросы команды DONUT:

Specify inside diameter of donut <default>:

(Внутренний диаметр кольца <по умолчанию>:)  
Specify outside diameter of donut <default>:

(Внешний диаметр кольца <по умолчанию>:)  
Specify Center of donut or [exit]:

(Центр кольца или [выход]:)  
Specify Center of donut or [exit]:

(Центр кольца или [выход]:)  
Enter – завершает работу команды.

**МУЛЬТИЛИНИЯ (MLINE)** – объект, состоящий из пучка ломаных, параллельных друг другу линий. Количество линий, входящих в мультилинию составляет от 2 до 16. Мультилинии могут иметь торцевые ограничители различного вида (например, отрезки, дуги и т.п.).

Запросы:

Current setting: Justification=Top, Scale = 20.00, Style = STANDARD

(Текущие настройки: Расположение = Верх, Масштаб = 20.00, Стил  
= STANDARD)

Specify start point or [Justification/ Scale/ Style]:

(Начальная точка или [Расположение/ Масштаб/ Стил]:)  
Specify next point:

(Следующая точка):  
Specify next point or [Undo]:

(Следующая точка или [Отменить]):  
Specify next point or [Close/Undo]:

(Следующая точка или [Замкнуть/ Отменить]):

Опции

Justification – определение положения точки начала черчения;

Top (Верх) – линия проходит с максимальным положительным смещением от заданной точки ;

Zero (Центр) – линия проходит с нулевым смещением от заданной точки;

Bottom (Низ) – линия проходит с максимальным отрицательным смещением от заданной точки;

Scale – число, на которое умножается смещение между линиями, устанавливаемое в стиле мультилинии;

Style – выбор стиля.

Команда **ЭСКИЗ (SKETCH)** – рисование эскиза – наилучший способ ввода графических образов, не имеющих строгой геометрической формы.

Эскизы состоят из множества прямолинейных сегментов. Каждый из сегментов представляет собой либо отдельный объект, либо сегмент полилинии. Эскизы, состоящие из множества маленьких линейных сегментов, позволяют рисовать с достаточно высокой точностью, но при этом резко увеличивается размер файла рисунка. Так что данное средство применяется

только в крайнем случае.

Запросы команды SKETCH:

Record increment <default>:

(Приращение <по умолчанию>:)

Sketch. Pen Exit Quit Record Erase Connect.

(Эскиз. Перо Выход Покинуть Записать Стереть Продолжить.)

Опции команды ЭСКИЗ:

Pen – вызов этой опции приводит к опусканию пера, если оно было поднято, и к поднятию, если оно было опущено.

Exit – данная опция записывает эскиз в базу данных текущего рисунка и завершает работу команды.

Quit – опция приводит к уничтожению всех временных отрезков, созданных текущей командой и заканчивает работу с ней.

Record – данная опция записывает существующий на текущий момент эскиз в базу данных рисунка, не изменяя состояния или положения пера.

Erase – данная опция позволяет выборочно стереть любой фрагмент, нарисованный эскизной линией.

Connect – опция дает возможность продолжать выполнение эскиза, если перо было поднято опциями Erase или Pen.

#### **5.5.4. Оформление чертежей**

*Тип линии* – это повторяющаяся последовательность штрихов, точек и пробелов, определяющих внешний вид линии.

Для работы с любым типом линии его предварительно необходимо загрузить в рисунок. В раскрывающемся списке «Тип линий» помимо двух служебных значений ByLayer и ByBlock, есть только тип линии Continuous, который используется в качестве имени для обычной сплошной тонкой линии. Загрузить другие типы линии можно щелкнув по строке «Другое...», либо выбрав пункт «Тип линии...» падающего меню «Формат». Раскрывается окно «Менеджер типов линии». Это окно позволяет назначить новый текущий тип линии, удалить существующий и загрузить новый тип линии.

*Текстовые стили.* С каждой новой текстовой надписью связан некоторый текстовый стиль. При нанесении надписей используется текущий стиль, который задает шрифт, высоту, угол поворота, ориентацию и другие параметры. Текстовые стили представляют собой неграфические объекты, которые также хранятся в файле рисунка.

Создание и модификация текстового стиля производится в диалоговом окне «Стиль текста» из падающего меню «Формат».

Однострочный текст может состоять из одного символа, одного слова или любого количества слов. Для включения в чертеж одной строки текстовой информации используется команда ТЕКСТ (TEXT), которая вызывается из меню Черчение→Текст→Однострочный.

Запросы:

Current text style: "Standard" Text height: 2.5

(Текущий текстовый стиль : "Standard". Высота текста: 2.5)

Specify start point of text or [Justify/ Style]:

(Начальная точка текста или [Выравнивание/ Стиль]:)

Specify height <default>:

(Высота по <умолчанию>:)

Specify rotation angle of text <0>:

(Угол поворота текста <0>:)

Enter text:

Enter text:

Для завершения команды нажать клавишу Enter.

Запрос определения высоты Specify height <default>: появляется в том случае, если при описании текущего текстового стиля высота была задана равной 0.

При вводе текста символы отображаются на экране, но не размещаются окончательно. Если в процессе ввода текста указать точку, то курсор перемещается на нее. После этого можно продолжить вводить текст. Фрагмент текста в новом положении представляет собой самостоятельный объект.

Опции:

Style – установит текущий стиль;

Justify – установит режим выравнивания текстовой строки с использованием опции выравнивания;

Align – формирует вписанный текст. Запрашивает начальную и конечную точки текста. Высота и ширина каждого символа автоматически вычисляется так, чтобы текст точно вписался в заданную область;

Fit – формирует вписанный по ширине и высоте текст. Запрашивает начальную и конечную точки текста, а также его высоту;

Center – обеспечивает центрирование базовой линии текстовой строки относительно заданной точки;

Middle – обеспечивает горизонтальное и вертикальное центрирование текстовой строки относительно заданной точки;

Right – служит для выравнивания текстовой строки по правому краю;

С помощью перечисленных ниже опций команда Текст формирует текстовую строку, выровненную:

TL – вверх влево;

TC – вверх по центру;

TR – вверх направо;

ML – по середине влево;

MC – по вертикали и горизонтали в средней точке;

MR – по середине вправо;

BL – вниз влево;

BC – вниз по центру;

BR – вниз вправо.

Многострочный текст представляет собой массив текста, состоящий из текстовых строк или абзацев, вписанных в задаваемую пользователем ширину абзаца. Основное отличие его от однострочного текста состоит в том, что весь массив представляет собой один объект. Формирование многострочного текста происходит при активизации команды МТЕКСТ (МТЕХТ) из выпадающего меню Черчение → Текст → Многострочный.

Запросы:

Current text style: "Standard" Text height: 10.0

(Текущий текстовый стиль : "Standard". Высота текста: 10.0)

Specify first corner or [Height/ Justify/ Line spacing/ Rotation/ Style/ Width]:

(Противоположный угол или [Высота/ Выравнивание/ Межстрочный интервал/ Поворот/ Стиль/ Ширина]:)

После указания размеров абзаца загружается диалоговое окно «Редактор многострочного текста», которое содержит закладки, позволяющие определить параметры: шрифт, размер шрифта, тип шрифта, цвет, символ и т.п.

*Ввод специальных символов.* Специальные символы выводятся на чертеж при помощи управляющих кодов. Эти коды начинаются с двух символов процента %%.

Например:

градус (°) – %%d;

допуск (±) – %%p;

диаметр (∅) – %%c;

процент (%) – %%%

*Штриховка* – заполнение замкнутой области определенным узором или заливка сплошным цветом.

*Размеры.* Размерный стиль – это набор параметров, задающих внешний вид и формат представления данных в размерном блоке.

Линейные размеры проставляются под различным углом и вызываются командой РЗМЛИНЕЙНЫЙ (DIMLINEAR), создают горизонтальные, вертикальные и повернутые размеры.

Запросы:

Specify first extension line origin or <select object>:

(Начало первой выносной линии или <выбрать объект>:)

Specify second extension line origin:

(Начало второй выносной линии:)

Specify dimension line location or [Mtext/ Text/ Angle/ Horizontal/ Vertical/ Rotated]:

(Положение размерной линии или [Мтекст/ Текст/ Угол/ Горизонтальный/ Вертикальный/ Повернутый]:)

Dimension text = значение

(Размерный текст = значение).

Если на первый запрос нажать клавишу Enter, то команда выдает следующие запросы:

Specify first extension line origin or <select object>:

(Начало первой выносной линии или <выбрать объект>:) – нажать

клавишу Enter для указания объекта;

Specify object of dimension:

(Выберите объект для нанесения размера:)

Specify dimension line location or [Mtext/ Text/ Angle/ Horizontal/ Vertical/ Rotated]:

Dimension text = значение

Опции:

Mtext – редактирование размерного текста с помощью редактора многострочного текста;

Text – редактирование размерного текста;

Angle – задается угол поворота размерного текста;

Horizontal – определяется горизонтальная ориентация размера, отмеряется расстояние между двумя точками по оси X;

Vertical – определяется вертикальная ориентация размера, отмеряется расстояние между двумя точками по оси Y;

Rotated – осуществляется поворот размерной и выносной линий, отмеряется расстояние между двумя точками вдоль заданного направления.

Параллельный размер создается при помощи команды РЗМПАРАЛ (DIMALIGNED) параллельно измеряемой линии.

Запросы:

Specify first extension line origin or <select object>:

(Начало первой выносной линии или <выбрать объект>:)

Specify second extension line origin:

(Начало второй выносной линии:)

Specify dimension line location or [Mtext/ Text/ Angle]:

(Положение размерной линии или [Мтекст/ Текст/ Угол]:)

Dimension text = значение

Радиальные размеры создаются командами РЗМДИАМЕТР (DIMDIAMETER). Команда строит диаметр окружности или дуги. Команда РЗМРАДИУС (DIMRADIUS) строит радиус окружности или дуги.

Запросы:

Specify arc or circle:

(Выберите дугу или круг:)

Dimension text = значение

Specify dimension line location or [Mtext/ Text/ Angle]:

(Положение размерной линии или [Мтекст/ Текст/ Угол]:)

Угловые размеры РЗМУГЛОВОЙ (DIMANGULAR).

Запросы:

Select arc, circle, line or <specify vertex>:

(Выберите дугу, круг, отрезок или <указать вершину>:)

Select second line: – если первое указание было отрезком, то указать второй отрезок, не параллельный первому.

Specify dimension arc line location or [Mtext/ Text/ Angle]:

(Положение размерной дуги или [Мтекст/ Текст/ Угол]:).

Если на первый запрос была нажата клавиша Enter, то угловой размер строится по трем точкам и команда DIMANGULAR выдает следующие запросы:



Specify angle vertex:

(Вершина угла:)

Specify first angle endpoint:

(Первая конечная точка угла:)

Specify second angle endpoint:

(Вторая конечная точка угла:)

Specify dimension arc line location or [Mtext/ Text/ Angle]:

(Положение размерной дуги или [Мтекст/ Текст/ Угол]:)

Ординатные размеры РЗМОРДИНАТА (DIMORDINATE) выражают расстояние по оси координат от базовой точки до образмериваемого объекта. Ординатный размер состоит из значения координаты X или Y и выноски. Ординатный размер X выражает расстояние от начала координат до объекта вдоль оси X, по оси Y – то же, но вдоль оси Y. Если указана точка, то автоматически определяется по какой оси проставлять размер. Такой способ называется автоматическим нанесением ординатных размеров.

Запросы:

Specify feature location:

(Укажите положение элемента:)

Specify leader endpoint or [Xdatum/ Ydatum/ Mtext/ Text/ Angle]: - указать конечную точку выноски или координату, которую необходимо изменить, а следовательно, изменить ориентацию выносок и размерного текста.

(Конечная точка выноски или [Xзначение/ Yзначение/ Мтекст/ Текст/ Угол]:)

Dimension text = измеренное значение.

*Выноска* – линия, соединяющая на рисунке пояснительную надпись с объектом, к которому она относится. Команда БВЫНОСКА (QLEADER)/

Запросы:

Select first leader point or [Settings]<Settings>:

(Первая точка выноски или [Параметры] <Параметры>:)

Select next point:

(Следующая точка:)

Select next point:

(Следующая точка:)

Select text width <0>:

Ширина текста <0>:

Enter first line of annotation text <Mtext>:

(Первая строка текста пояснения <Мтекст>:

Enter next line of annotation text:

(Следующая строка текста пояснения:)

Enter next line of annotation text: - нажать клавишу Enter для завершения команды.

### 5.5.5. Служебные средства

*Лимиты* – это ограниченная область рисунка. Практически лимиты представляют пару двумерных точек в мировой системе координат: координаты левого нижнего и правого верхнего угла, определяющие прямо-

угольную область. Команда вызывается из падающего меню «Формат» → «Ограничения». По умолчанию координаты левого нижнего угла совмещаются с началом координат и равны 0,0. Координаты правого верхнего угла зависят от размеров рабочей области рисунка. В пределах лимитов устанавливается вспомогательная сетка.

В экранном меню предусмотрены специальные режимы рисования: ШАГ, СЕТКА, ОРТО, ПОЛЯР, ВЫРВ, СЛЕД, ВЕСЛИН, МОДЕЛ.

Кнопка ШАГ позволяет включать и выключать режим привязки к точкам сетки с определенным настраиваемым шагом или к угловой привязке (перемещение тогда осуществляется по прямолинейным сегментам с заданным шагом).

Кнопка СЕТКА позволяет включать или выключать отображаемую в зоне лимитов сетку из точек с настраиваемым шагом.

Кнопка ОРТО включает или выключает режим ортогональности. Если этот режим включен, то AutoCad начинает исправлять вновь строящиеся сегменты отрезков и полилиний до вертикальности и горизонтальности.

ПОЛЯР является расширением режима ОРТО на углы с некоторым настраиваемым шагом.

Кнопка ВЫРВ позволяет включать или выключать режим постоянного действия заданных функций объектов привязки. При указании точки на объекте, система вычисляет соответствующую функцию объектной привязки к этому объекту (то есть конечную точку или среднюю точку).

При включении кнопки СЛЕД режима отслеживания объектной привязки программа подсказывает, какая функция объектной привязки могла бы быть использована вблизи показываемого примитива.

Кнопка ВЕСЛИН включает или выключает режим отображения весов элементов чертежа. Вес линии – это ширина, с которой линия будет выводиться на внешнее устройство.

Кнопка МОДЕЛ позволяет переключаться между пространствами модели и листа.

Чтобы обновить изображение на экране монитора, его можно перерисовать или регенерировать. При перерисовке экран просто очищается от разного рода «мусора» – временных элементов, появляющихся на экране. При регенерации, кроме перерисовки изображения текущего видового экрана, производится пересчет экранных координат всех объектов базы данных рисунка. Иногда в процессе работы возникает необходимость полной регенерации рисунка с пересчетом экранных координат всех объектов.

Команда ОБНОВИТЬ (REDRAW) вызывается из падающего меню «Вид».

Для регенерации рисунка используется команда РЕГЕН (REGEN)

Панорамирование – динамическое перемещение рисунка по экрану без изменения его масштаба. Режим панорамирования в реальном времени используется по умолчанию при вызове команды ПАН (PAN).

Зумирование ПОКАЗАТЬ (ZOOM) – это изменение масштаба чертежа, отображаемого на экране. При зумировании абсолютные размеры рисунка остаются прежними, изменяется лишь размер его части, отображаемой в графической области.

### 5.5.6. Редактирование чертёжесей

*Выбор объектов.* Большинство команд редактирования предлагает пользователю указать объекты для работы с ним. Выбранная группа объектов называется набором. Набор объектов можно создавать как до, так и после вызова команды редактирования. Имеется возможность добавлять объекты в набор и удалять их оттуда. Выбранные примитивы отмечаются подсветкой. Для управления процессом выбора объектов необходимо загрузить из падающего меню «Инструменты» → «Опции ...» диалоговое окно «Параметры». Задание метода сортировки объектов осуществляется в закладке Настройки пользователя. Режим выбора объектов и размер прицела определяется в закладке Выбор.

*Удаление объектов* осуществляется командой СТЕРЕТЬ (ERASE), которая вызывается из падающего меню «Изменить» → «Стереть» или щелчком мыши по пиктограмме «Стереть» панели инструментов Изменить. После вызова команды в ответ на запрос «Выберите объекты:» любым способом указать объекты, которые необходимо стереть на чертеже, и закончить выполнение команды нажатием правой клавиши мыши или клавиши Enter.

*Перемещение объектов* выполняется командой ПЕРЕНЕСТИ (MOVE), которая вызывается из падающего меню «Изменить» или щелчком мыши по пиктограмме «Переместить» панели инструментов «Изменить».

После вызова команды и выбора объектов появляется запрос:

Specify base point or displacement:

(Определите базовую точку или перемещение:)

Specify second point of displacement or <use first point as displacement>:

(Вторая точка перемещения или <считать перемещением первую точку>:)

Команда ПОВЕРНУТЬ (ROTATE) осуществляет поворот объектов и находится в меню «Изменить» и на панели инструментов «Изменить». После вызова команды и выбора объектов появляется запрос:

Specify base point

(Определите базовую точку:)

Specify rotation angle or [Reference]:

(Определите угол поворота или <Ссылка>:)

Опция Reference используется для поворота относительно существующего угла. При этом выдаются запросы:

Specify the reference angle <0>:

(Определите ссылаемый угол <0>:)

Specify the new angle:

(Определите новый угол:)

Копирование объектов производится командой КОПИРОВАТЬ (COPY), которая вызывается из падающего меню «Изменить» или соответствующей панели инструментов.

Запросы:

Select objects:

(Выберите объекты:)

Select objects: - нажать клавишу Enter для окончания выбора объектов

Specify base point or displacement, or [Multiple]:

(Базовая точка или перемещение, или [Несколько]:)

Specify second point or displacement, or <use first point as displacement>^

(Вторая точка перемещения или <считать перемещением первую точку>:)

Опция Multiple используется для создания множества копий объектов. При его применении последний запрос, требующий указания точки смещения задается многократно. Каждое смещение определяется относительно исходной базовой точки. После получения нужного числа копий в ответ на запрос необходимо нажать клавишу Enter.

Команды Снятие фасок ФАСКА (CHAMFER) и рисование скруглений СОПРЯЖЕНИЕ (FILLET) вызываются из падающего меню «Изменить» или одноименной панели инструментов.

Процесс снятия фасок заключается в соединении двух непараллельных объектов с помощью их удлинения или обрезки до пересечения либо друг с другом, либо с линией фаски.

Запросы команды CHAMFER:

(TRIM mode)Current Dist1=10.0, Dist2=10.0

(Режим с обрезкой) Параметры фаски: Длина 1=10.0, Длина 2 = 10.0)

Select first line or [Polyline/ Distance/ Angle/ Trim/ Method]:

(Выберите первый отрезок или [Полилиния/ Длина/ Угол/ Обрезка/ Метод]:)

Select second line:

(Выберите второй отрезок:)

Опции команды фаска:

Polyline – снятие фасок вдоль всей полилинии, то есть в каждом пересечении ее сегментов. При этом обрабатываются только те сегменты, длины которых превосходят длину фаски;

Distance – производит настройку длины фаски;

Angle – задает длину для первой фаски и угол относительно первой линии для подрезания второй линии;

Trim – определяет, обрезать ли линии до снятия фаски;

Method – выбирает один из методов задания размеров фасок: либо расстояниями, либо расстоянием и углом.

СОПРЯЖЕНИЕ осуществляет плавное скругление объектов, т.е.

плавное соединение двух объектов дугой заданного радиуса.

Запросы:

Current setting: Mode = TRIM, Radius=10.0

(Текущие настройки: Режим = с обрезкой, радиус = 10.0)

Select first object or [Poliline/ Radius/ Trim]:

(Выберите первый объект или [полилиния/ Радиус/ Обрезка]:)

Select second object:

(Выберите второй объект:)

При первом запуске команды устанавливается нужный радиус сопряжения, а при втором запуске выбираются два сопрягаемых объекта.

### 5.5.7. Выход из графического редактора

Для выхода можно использовать одно из следующих действий:

- щелкнуть мышью на значок «закрывать» – крестик в правом верхнем углу экрана;
- набрать в командной строке слово Quit и нажать Enter;
- выбрать в меню ФАЙЛ пункт Выход.

Команда позволяет сохранить или проигнорировать сделанные в рисунке изменения и выйти из программы. Если информация в чертеже не была сохранена, то AutoCAD предложит ее сохранить.

Пример создания графических документов приведен на рис. 5.2.

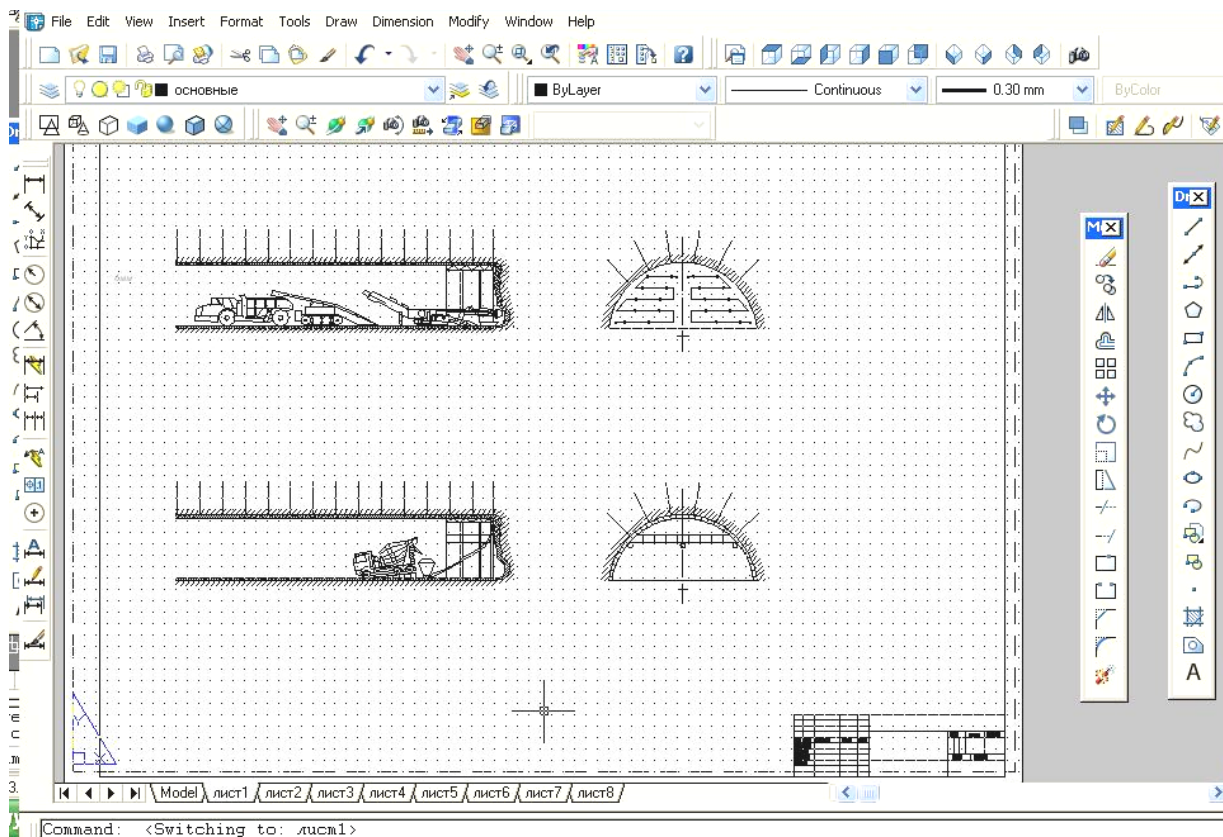


Рис. 5.2. Создание графического документа в AutoCAD

## 5.6. Понятие геоинформационных систем (ГИС)

В последние десятилетия активно развивается новая разновидность систем интерактивной компьютерной графики – *географические информационные системы* (ГИС). В них используются методы и алгоритмы многих наук и информационных технологий. В первую очередь – это алгоритмы и методы компьютерной графики, достижения технологий баз данных, кроме того, в них заложены многие методы и алгоритмы математики, физики, геодезии, топологии, картографии, навигации.

Географическая Информационная Система или ГИС – это компьютерная система, позволяющая показывать данные на электронной карте. Карты, созданные с помощью ГИС, можно назвать картами нового поколения. На карты ГИС можно нанести не только географические, но и статистические, демографические, технические и многие другие виды данных и применять к ним разнообразные аналитические операции. ГИС обладает уникальной способностью выявлять скрытые взаимосвязи и тенденции, которые трудно или невозможно заметить, используя привычные бумажные карты.

Электронная карта, созданная в ГИС, поддерживается мощным арсеналом аналитических средств, богатым инструментарием создания и редактирования объектов, а также базами данных, специализированными устройствами сканирования, печати и другими техническими решениями, средствами Интернет и даже космическими снимками и информацией со спутников.

Системы типа ГИС зачастую требуют значительных мощностей компьютера как в плане работы с базами данных, так и для визуализации объектов, которые находятся на поверхности Земли. Причем, визуализацию необходимо делать с различной степенью детализации – как для Земли в целом, так и в границах отдельных участков. В настоящее время заметно стремление разработчиков ГИС повысить реалистичность изображений пространственных объектов и территорий.

Типичными для любой ГИС являются такие операции – ввод и редактирование объектов с учетом их расположения на поверхности Земли, формирование разнообразных цифровых моделей, запись в базы данных, выполнение разнообразных запросов к базам данных. Важной операцией является анализ с учетом пространственных, топологических отношений множества объектов, расположенных на некоторой территории.

Одним из примеров использования ГИС можно назвать управление территориально-распределенными комплексами. К таким комплексам относятся современные горные предприятия. Для решения этой задачи требуется наличие надежных средств хранения и обработки больших (до 10 Гигабайт) массивов информации, и соответствующих средств визуализа-

ции графических данных. Внедрение компьютеров в данную область позволило в целом решить основные проблемы управления за счет:

- повышения степени достоверности информации, устранения дублирования и противоречивости данных;
- увеличения степени надежности хранения и обработки информации;
- обеспечения представления информации в различной, удобной для пользователя форме;
- сокращения времени на получение необходимой информации.

Основными формами представления информации в геоинформационных системах являются цифровые карты (метрическая и топологическая информация), семантические базы данных (семантическая информация) и служебные базы данных, содержащие информацию о картографических проекциях цифровых карт, годах состояния местности и т.д. С точки зрения программных средств цифровые карты, семантические и служебные базы данных представляются в виде файлов. Внутренняя структура файлов, способ кодирования информации в них зависят от конкретной программной реализации ГИС.

Вся совокупность информации, хранящейся в ГИС на данную территорию, называется *картографическим банком данных*. Картографический банк данных содержит цифровые карты, семантические и служебные базы данных. Для работы с информацией, хранящейся в банке данных ГИС, предусматривается специальный набор функциональных средств, называемых *системой управления картографическим банком данных*. К основным функциям системы управления относятся: добавление, удаление и модификация содержащейся в банке информации, изменение структуры банка, контроль целостности информации, средства ограничения доступа и т.д.

Цифровой банк данных является самой важной частью ГИС, поскольку он является основным источником сведений о моделируемой территории. От точности, полноты и корректности хранимой в нем информации, в конечном итоге, зависит результат работы всей ГИС. По этой причине к разработке структуры и требований к банку данных следует подходить с особой тщательностью.

Основой картографического банка данных являются цифровые карты, содержащие данные, выраженные в единой системе координат и с определенной точностью. Сам банк данных может содержать несколько цифровых карт и связанных с ними баз данных семантической информации.

## *Контрольные вопросы*

1. Для решения каких задач используется компьютерная графика?
2. Каковы приложения компьютерной графики?
3. Что такое САПР? Где они используются?
4. Дайте понятия векторной и растровой графики.
5. Какие форматы графических файлов вы знаете?
6. Какие профессиональные пакеты используются для создания графической документации в горном деле?
7. Какие специализированные программы базируются на платформе AutoCAD? Для чего они могут применяться в горном деле?
8. Как в AutoCAD создаются графические изображения?
9. Какие геометрические примитивы используются для построения графических объектов в AutoCAD?
10. Как редактируются изображения в AutoCAD?
11. Назовите известные геоинформационные системы. Для каких целей они используются?



## **6. ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ**

---

### **6.1. Алгоритм и его свойства**

Алгоритмы могут описывать процессы преобразования самых разных объектов. Широкое распространение получили вычислительные алгоритмы, которые описывают преобразование числовых данных. Само слово «алгоритм» происходит от *algorithmi* – латинской формы написания имени выдающегося математика IX века аль-Хорезми, который сформулировал правила выполнения арифметических операций.

**Алгоритм** – это строго определенная последовательность действий, описывающая процесс преобразования объекта из начального состояния в конечное, записанная с помощью понятных исполнителю команд.

Алгоритмы состоят из отдельных команд, которые исполнитель выполняет одну за другой в определенной последовательности. Разделение информационного процесса в алгоритме на отдельные команды является важным свойством алгоритма и называется *дискретностью*.

Каждый исполнитель обладает определенным набором, системой команд, которые он может выполнить. Алгоритм должен быть понятен исполнителю, то есть должен содержать только те команды, которые входят в систему его команд.

Запись алгоритма должна быть такова, чтобы, выполнив очередную команду, исполнитель точно знал, какую команду необходимо исполнять следующей. Это свойство алгоритма называется *детерминированностью*.

Должны быть определены начальное состояние объекта и его конечное состояние (цель преобразования). Алгоритм должен обеспечивать преобразование объекта из начального состояния в конечное за конечное число шагов. Такое свойство алгоритма называется *результативностью*.

Представление информационного процесса в форме алгоритма позволяет поручить его автоматическое исполнение различным техническим устройствам, среди которых особое место занимает компьютер. При этом говорят, что компьютер исполняет программу (последовательность команд), реализующую алгоритм.

Алгоритм, записанный на «понятном» компьютеру языке программирования, называется **программой**.

### **6.2. Развитие языков программирования**

Информацию в компьютере обрабатывает процессор, следовательно, алгоритм должен быть записан на языке, «понятном» для процессора, то есть на машинном языке, представляющем собой логические последовательности нулей и единиц.

На заре компьютерной эры, в 50-е годы XX века, программы писались на машинном языке и представляли собой очень длинные последовательности нулей и единиц. Составление и отладка таких программ было чрезвычайно трудоемким делом.

В 60-70-е гг. для облегчения труда программистов начали создаваться языки программирования высокого уровня, формальные языки, кодирующие алгоритмы в привычном для человека виде (в виде предложений). Такие языки программирования строились на основе использования определенного алфавита и строгих правил построения предложений (синтаксиса).

Наиболее широко распространенным типом языков программирования высокого уровня являются процедурные языки. В таких языках широко используются управляющие конструкции (операторы), которые позволяют закодировать различные алгоритмические структуры (линейную, ветвление, цикл).

Одним из первых процедурных языков программирования был известный всем Бейсик (Basic), созданный в 1964 году. В течение последующего времени Бейсик развивался, появлялись его различные версии (MSX-Basic, Бейсик-Агат, QBasic и др.). Другим широко распространенным языком программирования алгоритмического типа является Pascal.

### **6.3. Основы объектно-ориентированного визуального программирования**

Приложения на языках объектно-ориентированного программирования Visual Basic и Visual Basic for Applications (VBA) строятся из объектов, подобно тому, как из блоков и различных деталей строятся дома. Программные библиотеки готовых объектов входят в эти системы программирования, причем языки Visual Basic и VBA различаются между собой, главным образом, составом программных библиотек.

Системы объектно-ориентированного программирования дают возможность визуализировать процесс создания графического интерфейса разрабатываемого приложения, то есть позволяют создавать объекты и задавать значения их свойств с помощью диалоговых окон системы программирования.

Взаимодействие программных объектов между собой и их изменения описываются с помощью программного кода. Создание программного кода в объектно-ориентированном программировании базируется на использовании алгоритмических структур различных типов (линейной, ветвления, цикла), исполнителями которых выступают программные объекты.

#### **6.3.1. Объекты: свойства, методы, события**

Основной единицей в объектно-ориентированном программировании является программный *объект*, который объединяет в себе как описывающие его данные (*свойства*), так и средства обработки этих данных (*мето-*

ды). Если говорить образно, то объекты – это «существительные», свойства объекта – это «прилагательные», а методы объекта – это «глаголы».

Программные объекты обладают свойствами, могут использовать методы и реагируют на события.

**Свойства объектов (Properties).** Каждый объект обладает определенным набором свойств, первоначальные значения которых можно установить с использованием диалогового окна системы программирования.

Значения свойств объектов можно изменять в программном коде. Для присвоения свойству объекта нового значения в левой части строки программного кода необходимо указать имя объекта и затем название свойства, которые в соответствии с правилами точечной нотации разделяются между собой точкой. В правой части строки (после знака равенства) необходимо записать конкретное значение свойства:

**Объект.Свойство = ЗначениеСвойства**

Например, установим в выделенном фрагменте текста (*объект Selection*) для первого символа (*объект Characters (1)*) начертание полужирный (*свойство Bold*).

Свойство **Bold** может быть установлено (значение **True** свойства) или не установлено (значение **False** свойства). Значения **True** и **False** являются ключевыми словами языка и поэтому выделяются полужирным начертанием.

Присвоим свойству **Bold** значение **True**:

```
Selection.Characters(1).Bold = True
```

Объект обычно имеет несколько свойств. С помощью конструкции **With . . . End With** можно задать значения сразу нескольких свойств объекта. Синтаксис установки значения нескольких свойств объекта:

**With**

```
Объект.Свойство1 = ЗначениеСвойства1 .Свойство2 = ЗначениеСвойства2  
.СвойствоN = ЗначениеСвойстваN
```

**End With**

Например, для придания выделенному фрагменту текста, состоящему из 10 символов, начертания «*полужирный курсив*» можно использовать следующий программный код:

```
For I = 1 To 10  
With Selection.Characters(I)  
.Bold = True  
.Italic = True End With Next I
```

**Методы объектов (Methods).** Для того чтобы объект выполнил какую-либо операцию, необходимо применить метод, которым он обладает. Многие методы имеют аргументы, которые позволяют задать параметры

выполняемых действий. Для присваивания аргументам конкретных значений используется двоеточие и знак равенства, а друг от друга аргументы отделяются запятой.

Обратиться к методу объекта можно так же, как и к свойству объекта, с использованием *точечной нотации*. Чтобы определить, для какого объекта вызывается метод, перед именем метода указывается имя объекта, отделенное точкой:

### Объект.Метод арг1:=значение, арг2:=значение

Так, сохранение на диске открытого в приложении Word документа реализуется методом Save, без аргументов:

```
Documents ("Проба.doc") .Save
```

Операция открытия в приложении **Word** документа Проба.doc должна содержать не только название метода Open, но и указание пути к открываемому файлу (аргументу FileName метода Open необходимо присвоить конкретное значение):

```
Documents() .Open FileName := "C:\Документы\Проба.doc"
```

**События (Events).** Событие представляет собой действие, распознаваемое объектом. Событие может создаваться пользователем (например, щелчок **мышью** или нажатие клавиши) или быть результатом воздействия других программных объектов. В качестве реакции на события вызывается определенная процедура, которая может изменять значения свойств объекта, вызывать его методы и так далее.

Например, объект Document (*Документ*) реагирует на события Open (*Открытие*), New (*Создание*) и Close (*Закрытие*), а объект Selection (*Выделенный фрагмент документа*) реагирует на события Cut (*Вырезка*), Copy (*Копирование*), Paste (*Вставка*), Delete (*Удаление*) и так далее.

### 6.3.2. Графический интерфейс и событийные процедуры

*Графический интерфейс.* Графический интерфейс необходим для реализации интерактивного диалога пользователя с работающим приложением. Основой для создания графического интерфейса разрабатываемого приложения являются *форма* (в Visual Basic — класс объектов Form, в VBA — класс объектов UserForm), представляющая собой окно, в котором размещаются управляющие элементы. Необходимо отметить, что графический интерфейс проекта может включать в себя несколько форм.

**Форма** — это объект, представляющий собой окно на экране, в котором размещаются управляющие элементы.

Визуальное конструирование графического интерфейса приложения состоит в том, что на форму с помощью мыши помещаются и «рисуются» те или иные *управляющие элементы*.

Классы *управляющих элементов* (Controls) имеют различное назначение в графическом интерфейсе приложения. *Текстовые поля* (TextBox), *метки* (Label) и *списки* (ListBox) обычно используются для ввода и вывода данных, *графические окна* (PictureBox) — для вывода графики, *командные кнопки* (CommandButton), *переключатели* (CheckBox) и *флажки* (OptionsButton) – для организации диалога и так далее.

На форму может быть помещено несколько экземпляров одного класса управляющих элементов, например, несколько кнопок Command1, Command2, Command3 и так далее, каждая из которых обладает индивидуальными значениями свойств (надпись, размеры и др.).

**Управляющие элементы** – это объекты, являющиеся элементами графического интерфейса приложения и реагирующие на события, производимые пользователем или программными объектами.

*Форма и управляющие элементы* обладают определенными наборами свойств, методов и событий (табл. 6.1).

Таблица 6.1

Примеры классов объектов, их свойств, методов и событий

Класс объектов	Свойства	Методы	События
Form (форма) UserForm (форма)	Name (Имя) Caption (Надпись) Font (Шрифт) Height (Высота) Width (Ширина)	Show (Показать) Move (Переместить)	Load (Загрузка)
CommandButton (командная кнопка)	Name (Имя) Caption (Надпись) Font (Шрифт) Height (Высота) Width (Ширина)	Move (Переместить)	Click (Щелчок)
TextBox (текстовое поле)	Name (Имя) Text (Текст) Font (Шрифт) Height (Высота) Width (Ширина)	Move (Переместить)	DblClick (Двойной щелчок)

**Событийные процедуры.** Для каждого события можно запрограммировать *отклик*, то есть реакцию объекта на произошедшее событие. Если пользователь производит какое-либо воздействие на элемент графического интерфейса (например, щелчок), в качестве отклика выполняется некоторая последовательность действий (событийная процедура).

Имя процедуры включает в себя имя объекта и имя события.

Объект Событие()

**Событийная процедура** представляет собой подпрограмму, которая начинает выполняться после реализации определенного события.

В событийной процедуре может участвовать несколько объектов. Например, само событие происходит с первым объектом (Объект1), в результате второй (Объект2) изменяет значение своего свойства, а третий (Объект3) реализует какой-либо метод и так далее.

Каждая процедура представляет собой отдельный программный модуль, который реализует определенный алгоритм. В терминологии процедурного программирования такие процедуры соответствуют подпрограммам, поэтому каждая из событийных процедур начинается с ключевого слова **Sub** (subroutine — подпрограмма) и заканчивается ключевыми словами **End Sub**.

## 6.4. Интегрированная среда программирования Visual Basic

Интегрированная среда разработки языка Visual Basic предоставляет пользователю удобный графический интерфейс в процессе разработки приложения. Рассмотрим интерфейс среды Visual Basic (рис. 6.1) и назначение каждой из панелей (окон).

**1. Главное меню.** Предназначено для управления проектом, т. е. выполнения команд создания, открытия, сохранения, печати, запуска, останова проекта, отмены или возврата действий, настройки интерфейса приложения и др. операций с проектом. Стандартно настроенное главное меню среды Visual Basic показано на рис. 6.2.

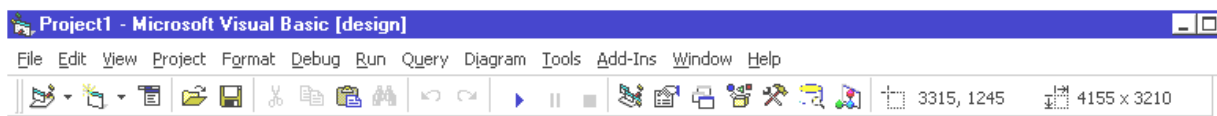


Рис. 6.2. Главное меню

**2. Панель инструментов.** В левой части окна интегрированной среды разработки Visual Basic располагается *Панель инструментов (Tool-Box)*, содержащая пиктограммы *управляющих элементов* (рис. 6.3). Стандартный набор управляющих элементов включает в себя 21 класс объектов: *CommandButton* (*командная кнопка*), *TextBox* (*текстовое поле*), *Label* (*надпись*) и т. д. Существует возможность дополнить *панель инструментов* новыми классами управляющих элементов *RichTextBox* (*усовершенствованное текстовое поле*), *ImageList* (*список изображений*) и др.

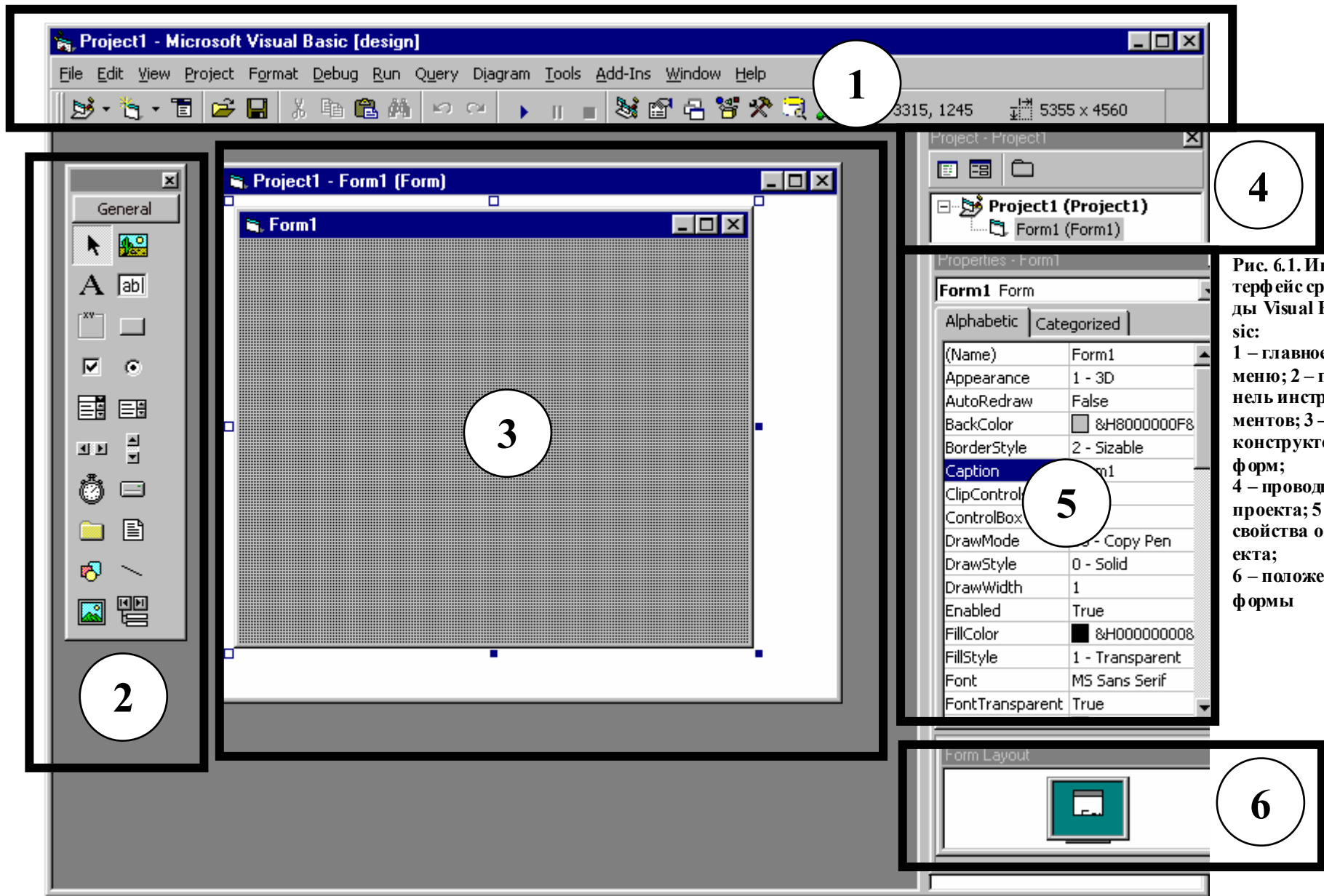


Рис. 6.1. Интерфейс среды Visual Basic:  
1 – главное меню; 2 – панель инструментов; 3 – конструктор форм; 4 – проводник проекта; 5 – свойства объекта; 6 – положение формы

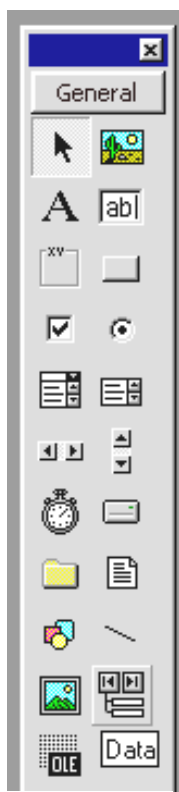


Рис. 6.3. Панель инструментов

Выбрав щелчком мышью на *Панели инструментов* нужный элемент, мы можем поместить его на форму проектируемого приложения. Процесс размещения на форме управляющих элементов аналогичен рисованию графических примитивов с использованием графического редактора.

Фактически мы размещаем на форме экземпляры определенных классов объектов. Например, выбрав класс `CommandButton`, мы можем разместить на форме неограниченное количество экземпляров этого класса, то есть командных кнопок `Command1`, `Command2`, `Command3` и так далее.

**3. Окно Конструктор форм.** Окно *Конструктор форм* (рис. 6.4) является основным рабочим окном и расположено в центре окна интегрированной среды разработки языка Visual Basic. По умолчанию проекту присваивается имя *Project1*. Именно в этом окне происходит визуальное конструирование графического интерфейса разрабатываемого приложения.

В окне *конструктор форм* располагается сама форма `Form1`, которая является также объектом и принадлежит классу объектов `Form`. Размеры формы можно менять, перетаскивая мышью правую или нижнюю границу формы.

Первоначально форма пуста, в дальнейшем, в процессе создания графического интерфейса приложения, на ней размещаются элементы управления.

**4. Окно Проводник проекта.** Окно *Проводник проекта (Project)* располагается в верхнем правом углу (рис. 6.5).

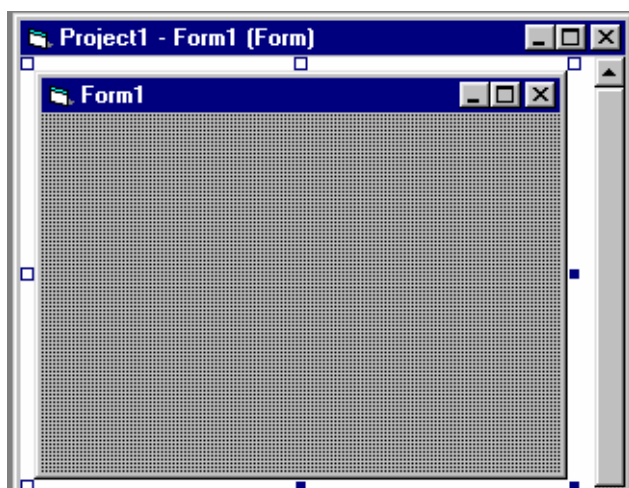


Рис. 6.4. Конструктор форм

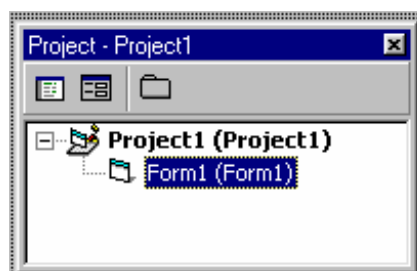
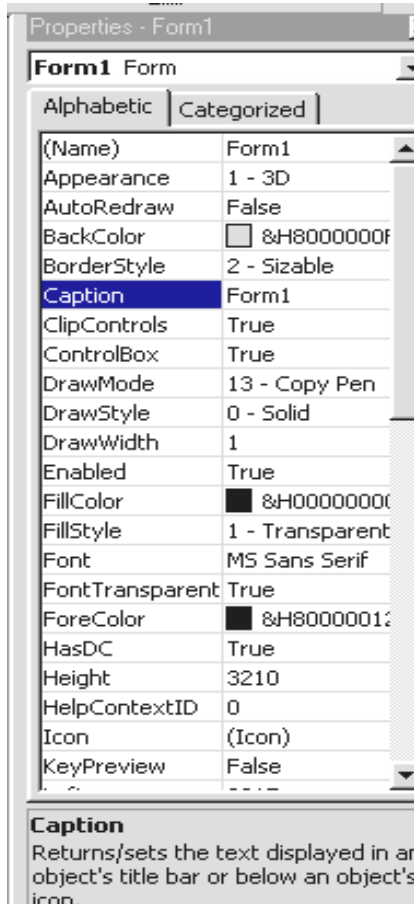


Рис. 6.5. Проводник проекта

Оно отображает в виде иерархического каталога все составные части текущего проекта (в данном случае *Project 1*) и позволяет переключаться между ними (по форме и по функциям оно аналогично окну *Проводник Windows*).



Проект хранится в файле с расширением vbr (в данном случае в файле project1.vbr). Кроме того, входящие в проект формы хранятся в отдельных файлах с расширением frm (в данном случае форма, входящая в состав проекта, хранится в файле Form1.frm).



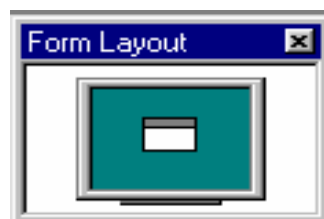
**Рис. 6.6. Окно Свойства объектов**

**5. Окно *Свойства объекта*.** Справа располагается окно *Свойства объекта (Properties)* – рис. 6.6. Окно содержит список объектов и список свойств, относящихся к выбранному объекту (форме или управляющему элементу на форме). На рисунке выбран объект Form1 класса Form.

Список свойств разделен на две колонки. В левой находятся имена свойств, в правой – их значения. Установленные по умолчанию значения могут быть изменены. Свойством объекта является количественная или качественная характеристика этого объекта (размеры, цвет, шрифт и др.).

Для некоторых свойств предусмотрена возможность выбора значений из раскрывающегося списка, например, из списка можно выбрать значение цвета фона формы (свойства BackColor).

**6. Окно *Расположение формы*.** В нижнем правом углу находится окно *Расположение формы (Form Layout)* – рис. 6.7. Оно показывает, где будет располагаться окно формы на экране монитора в период выполнения программы. Положение формы можно изменять перетаскиванием мышью.



**Рис. 6.7. Окно Расположение формы**

**Этапы разработки приложения.** Создание приложения в среде Visual Basic можно условно разделить на несколько этапов:

1. Создание графического интерфейса будущего приложения. В окне *Конструктор форм* на форму помещаются управляющие элементы, которые должны обеспечить взаимодействие приложения с пользователем.

2. Задание значений свойств объектов графического интерфейса. С помощью окна *Свойства объекта* задаются значения свойств управляющих элементов, помещенных ранее на форму.

3. Создание программного кода. В окне *Редактор кода* производится ввод и редактирование программного кода процедур.

4. Сохранение проекта. Так как проекты включают в себя несколько файлов (в том числе несколько файлов форм), рекомендуется для каждого проекта создать отдельную папку на диске. Сохранение проекта производится с помощью меню *File*. Сначала необходимо сохранить форму и связанный с ней программный модуль с помощью пункта меню *Save Form1.frm As...* По умолчанию для файла формы предлагается имя, заданное в качестве значения свойства Name и расширение frm.

Далее, необходимо сохранить файл проекта с помощью пункта меню *Save Project As...* В соответствии с соглашением об именах объектов целесообразно сохранить проект под именем с префиксом prj, например prjVBI.vbp.

5. Компиляция проекта в приложение. Сохраненный проект может выполняться только в самой системе программирования Visual Basic. Для того чтобы преобразовать проект в приложение, которое может выполняться непосредственно в среде операционной системы, необходимо сохранить проект в исполняемом файле (типа exe). Для компиляции проекта в исполняемый файл используется команда [File-Make ...]

## 6.5. Пример разработки приложения в среде Visual Basic

**Задание.** Разработать Windows-приложение для расчета необходимого числа шпуров для взрывания породного забоя, определяемого формулой

$$N = 12,7 \cdot \frac{q \cdot S \cdot \eta}{\gamma \cdot d^2 \cdot \rho},$$

где  $q$  – удельный расход ВВ,

$$q = q_1 \cdot f_1 \cdot v \cdot e;$$

здесь  $q_1$  – нормальный удельный расход ВВ,  $q_1 = 0,1f$ ;

$f$  – крепость взрываемых пород по шкале проф. Протодьяконова;

$f_1$  – коэффициент структуры породы,  $f_1 = 1,3 \div 1,4$ ;

$v$  – коэффициент зажима породы,  $v = \frac{6,5}{\sqrt{S}}$ ;

- $S$  – площадь поперечного сечения выработки,  $\text{м}^2$ ;
- $\eta$  – коэффициент использования шпура, принять равным:
- при крепости пород  $f < 4$   $\eta = 0,9$ ;
  - при крепости пород  $10 < f < 4$   $\eta = 0,85$ ;
  - при крепости пород  $f > 10$   $\eta = 0,8$ .
- $\gamma$  – коэффициент заполнения шпура,  $\gamma = 0,4 \div 0,5$ ;
- $d$  – диаметр патрона ВВ, см;
- $\rho$  – гравиметрическая плотность ВВ в патронах,  $\text{г}/\text{см}^3$ , принимаемая согласно приведенной таблицы в зависимости от типа ВВ;
- $e$  – коэффициент работоспособности ВВ, принимаемый согласно следующей таблице в зависимости от типа ВВ:

Тип ВВ	Коэффициент $e$	$\rho$ , $\text{г}/\text{см}^3$
Аммонит скальный №1	0,8	1,10
Детонит М	0,82	1,15
Аммонит АП-5ЖВ	1,17	1,15
Аммонит № 6ЖВ	1,00	1,20

Предусмотреть вывод результатов на экран и в файл. В качестве результатов вывести наименование выбранного пользователем ВВ, его основные характеристики и рассчитанные удельный расход ВВ и число шпуров на забой.

Разработку приложения для решения поставленной задачи производим в следующей последовательности:

**1-й этап** – создание графического интерфейса приложения. Интерфейс определяет внешний вид программы и содержит все необходимые объекты, позволяющие вводить исходные данные и управлять проектом.

Поскольку задача имеет небольшой объем исходных данных и результатов расчета, достаточно разработать только одну форму, на которой разместить все необходимые объекты. Для решения более сложных задач обычно требуется разработка нескольких экранных форм, которые в ходе выполнения программы сменяют друг друга. О создании многооконного приложения читайте в [6].

Исходя из поставленной задачи, определяем основные блоки, которые войдут в интерфейс программы:

- блок исходных данных;
- блок результатов расчета;
- блок элементов управления проектом.

**Формирование блока исходных данных.** При составлении перечня исходных данных, выносимых в интерфейс приложения, необходимо руководствоваться следующими рекомендациями:

- количество исходных данных должно быть минимально возможным, т.е. не следует запрашивать параметры, однозначно зависящие друг от друга. Например, в рассматриваемой задаче достаточно запросить только

ко крепость взрываемых пород, а значения коэффициента использования шпуров, а также нормального удельного расхода ВВ могут определяться в программе в зависимости от крепости, поэтому последние две величины выносить на экранную форму и запрашивать у пользователя не следует, так как это загромождает форму и делает программу неудобной для пользователя;

– на форму рекомендуется выносить только те исходные данные, значения которых могут изменяться в различных вариантах расчета. В данном примере это крепость пород, площадь сечения выработки, тип ВВ. Значения постоянных коэффициентов (например, коэффициент заполнения шпура, коэффициент структуры породы в рамках данной задачи), а также физические и математические константы выносить на форму не имеет смысла. Такие значения лучше ввести в программном коде с помощью операторов присвоения;

– размерность вводимых величин должна быть удобной для пользователя, т.е. соответствовать общепринятым для данного параметра единицам, например, диаметр патрона принято указывать в мм, крепость пород в безразмерных единицах по шкале проф. Протоdjeяконова и т.д. Если в формулы данные значения должны подставляться в других единицах измерениях, лучше этот перевод предусмотреть внутри программного кода (включить в формулу), а ввод данных организовать в «привычных» для пользователя единицах. Это снизит вероятность ошибки пользователем и сделает использование приложения более удобным. В любом случае на форме рядом с вводимой величиной рекомендуется указывать ее размерность;

– если возможные значения вводимых величин ограничены рядом стандартных значений, (например, диаметр патрона ВВ может принимать значения 28, 32, 36 или 45 мм), то такие данные целесообразно вводить с помощью специальных инструментов Option Button, Check Box или Combo Box, которые позволяют пользователю выбрать необходимую величину из заранее сформированного ряда значений или списка;

– при формировании большого списка выбора значений, удобнее всего использовать для ввода данных инструмент Combo Box (выпадающий список), который позволяет сэкономить место на форме.

Учитывая вышеизложенные рекомендации, составляем минимально возможный список исходных данных, которые необходимо вынести на форму приложения.

Для удобства все исходные данные, их единицы измерения и пределы допустимых значений сводим в табл. 6.2. Пределы значений, которые могут принимать вводимые переменные, указываем для того, чтобы затем в программном коде отсечь заведомо неверные значения, которые по ошибке может ввести пользователь. В этой же таблице указываем, какой из инструментов будет использоваться для ввода того или иного данного. Таким образом, из 12 переменных, входящих в расчетные формулы, в интерфейс приложения достаточно вынести всего 4 значения исходных данных (табл. 6.2), все остальные будут определены или заданы непосредственно в программном коде.

Таблица 6.2

## Исходные данные для расчета, вводимые пользователем

Вводимый параметр	Единица измерения	Пределы допустимых значений	Инструмент для ввода
Крепость пород	относ. ед.	1 – 20	текстовое окно
Площадь поперечного сечения выработки	м <sup>2</sup>	6 – 20	текстовое окно
Диаметр патрона	мм	28; 32; 36; 45	кнопки-переключатели
Тип ВВ	–	задаются списком выбора	выпадающий список

Выбор инструмента для ввода исходных данных определялся из следующих соображений:

– крепость пород и площадь сечения выработки могут принимать любые, в том числе и дробные значения, поэтому пользователю целесообразно предоставить пустое текстовое окно (Text Box) для ввода произвольного значения;

– диаметр патрона может принимать только 4 различных значения, поэтому во избежание ввода несуществующего диаметра принят инструмент «кнопки-переключатели» (Option Button), который позволяет выбрать только один из предложенных диаметров;

– тип ВВ представляет собой символьную (строковую) переменную, состоящую из набора определенных букв и цифр. Во избежание ошибочного написания переменной и для удобства пользователя применяем инструмент «выпадающий список» (Combo Box), который также позволяет выбрать нужное ВВ из заранее сформированного списка.

*Формирование блока результатов.* Перечень выводимых результатов расчета формируется исходя из условий каждой конкретной задачи, при этом наиболее важные и конечные результаты расчета целесообразно выводить непосредственно на форму, а полный перечень результатов, а также исходных данных сохранять в отдельном текстовом файле. Выводимые результаты приведены в табл. 6.3.

Таблица 6.3

## Выводимые результаты расчета

Выводимый параметр	Единица измерения	Способ (инструмент) вывода
Удельный расход ВВ	кг/м <sup>3</sup>	Текстовое окно + вывод в файл
Количество шпуров на забой	шт.	Текстовое окно + вывод в файл
Крепость пород	относ. ед.	Вывод в файл
Площадь поперечного сечения выработки	м <sup>2</sup>	Вывод в файл
Тип ВВ	–	Вывод в файл
Диаметр патрона	мм	Вывод в файл
Коэффициент работоспособности ВВ	–	Вывод в файл
Плотность патронирования ВВ	кг/м <sup>3</sup>	Вывод в файл

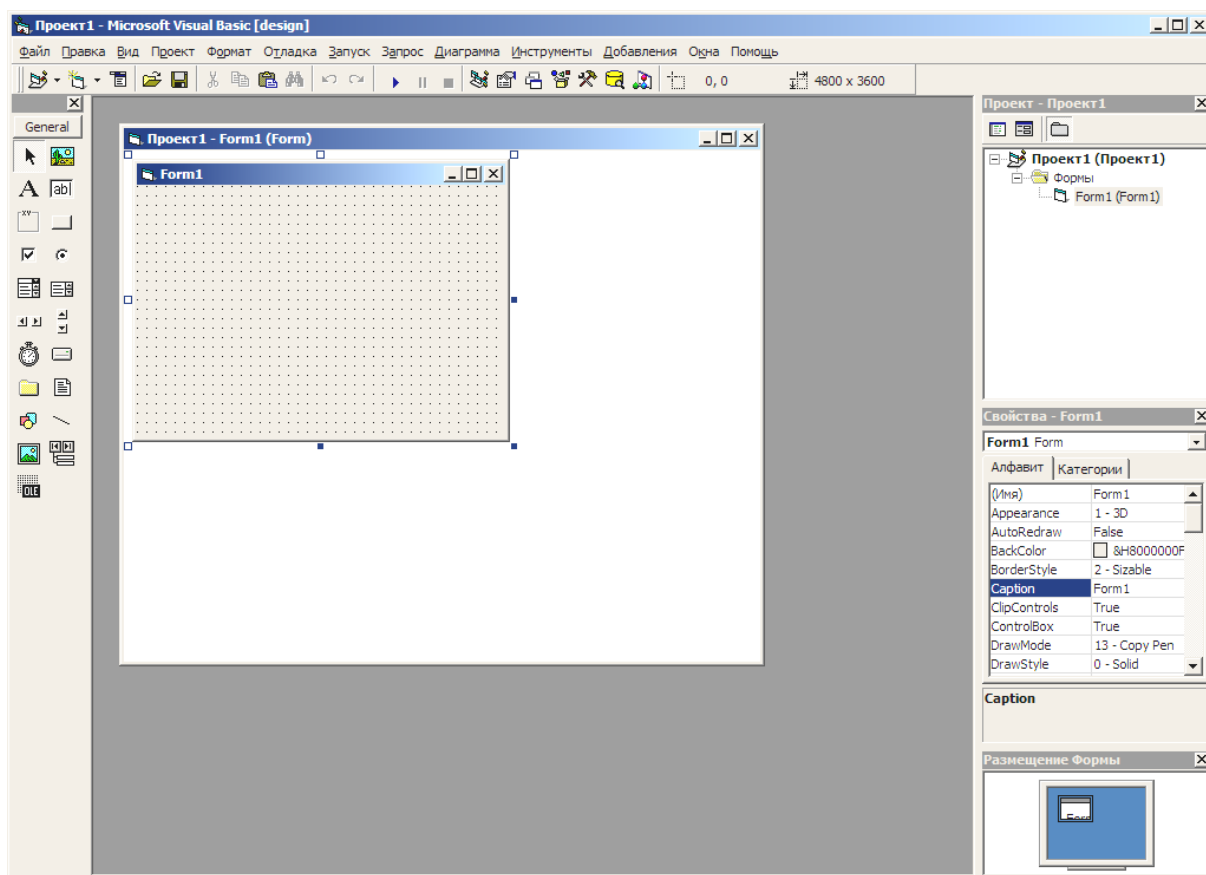
*Формирование блока элементов управления проектом.* В этот блок входят, как правило, командные кнопки (Command Button), которые позволяют пользователю производить необходимые действия: запускать программу для выполнения расчета, сохранять результаты в файл, выходить из программы и т.д.

В данном проекте будут использованы всего 4 командные кнопки:

- «Результат» – для запуска расчета;
- «Сохранить» – для запроса необходимости сохранения результатов в файл;
- «ОК» – для подтверждения введенных параметров (диска, имени файла, фамилии пользователя) для сохранения;
- «Выход» – для закрытия окна приложения и выхода из программы.

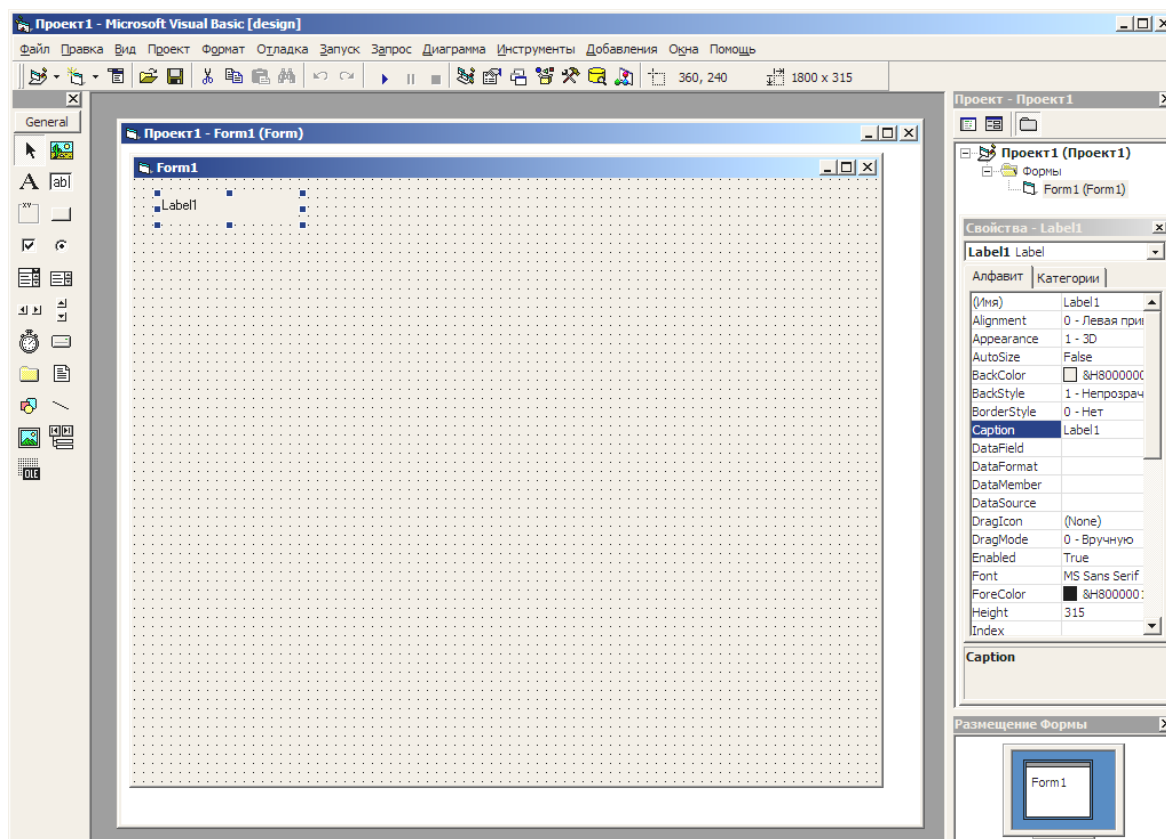
Определившись со всеми объектами приложения, переходим к их нанесению на форму, т.е. к непосредственному формированию интерфейса программы.

При запуске Visual Basic и выборе нового проекта «Стандартный exe» автоматически открывается проект с именем Проект1, имеющий пустую форму, которой по умолчанию присваивается имя Form1 (рис. 6.8). Поле этой формы – это и есть место для формирования интерфейса будущего приложения. Изменить размеры окон проекта или формы можно, «потянув» окно мышью за стороны или угол окна.



**Рис. 6.8.** Вид окна Visual Basic после запуска нового проекта

Для нанесения объектов на форму необходимо мышью выбрать на панели инструментов нужный объект и «растянуть» его в нужном месте на поле формы до необходимых размеров. Например, необходимо нанести надпись «Исходные данные». Для этого выбираем инструмент label (метка, на панели инструментов он обозначен иконкой **A** ) и наносим этот объект в левом верхнем углу формы (рис. 6.9).



**Рис. 6.9. Нанесение на форму первого объекта Label1**

Для внесения надписи внутрь объекта и задания его основных свойств (точных размеров, положения на форме, типа и цвета шрифта надписи, цвета фона, выравнивания надписи в форме, видимости и др.) необходимо использовать окно свойств, расположенное по умолчанию в правой стороне экрана. Так, для внесения надписи «Исходные данные» необходимо найти свойство Caption (заголовок) и справа от него в колонке свойств ввести соответствующую надпись, которая одновременно появится и на поле метки Label1. Аналогично можно изменить и другие свойства. Основные свойства объектов и их краткая характеристика приведены в табл. 6.4. Выбором цвета фона, цвета, размера и свойств (полужирный, курсив) шрифта можно выделять однородные блоки или надписи.



Ниже метки Label1 с надписью «Исходные данные» наносим метку Label2 с надписью «Крепость пород», а рядом с ней – текстовое окно Text1, в которое пользователь будет вводить необходимое значение крепости.


Текстовые окна наносятся на форму с помощью иконки **abl** .

Некоторые свойства объектов

Свойство	Характеристика
Alignment	Выравнивание надписи внутри объекта по левому, правому краю или по центру
AutoSize	При значении этого свойства True (истина) размеры объекта автоматически сжимаются до размеров внесенной в него надписи, при значении False (ложь) размеры объекта остаются такими, какими их задал программист
BackColor	Задание цвета фона объекта
BorderStyle	Задание стиля рамки объекта
Caption	Ввод надписи или заголовка
DataFormat	Выбор формата данных (общий, числовой, формат даты, времени и др.)
Font	Выбор типа и размера шрифта надписи
ForeColor	Задание цвета надписи
Height	Задание высоты объекта
Left	Задание положения объекта на форме относительно ее левой рамки
MousePointer	Выбор вида курсора при наведении мыши на данный объект
Picture	Вставка рисунка
Top	Задание положения объекта на форме относительно ее верхней рамки
Visible	Задание видимости (True) или невидимости (False) объекта на форме
Width	Задание ширины объекта

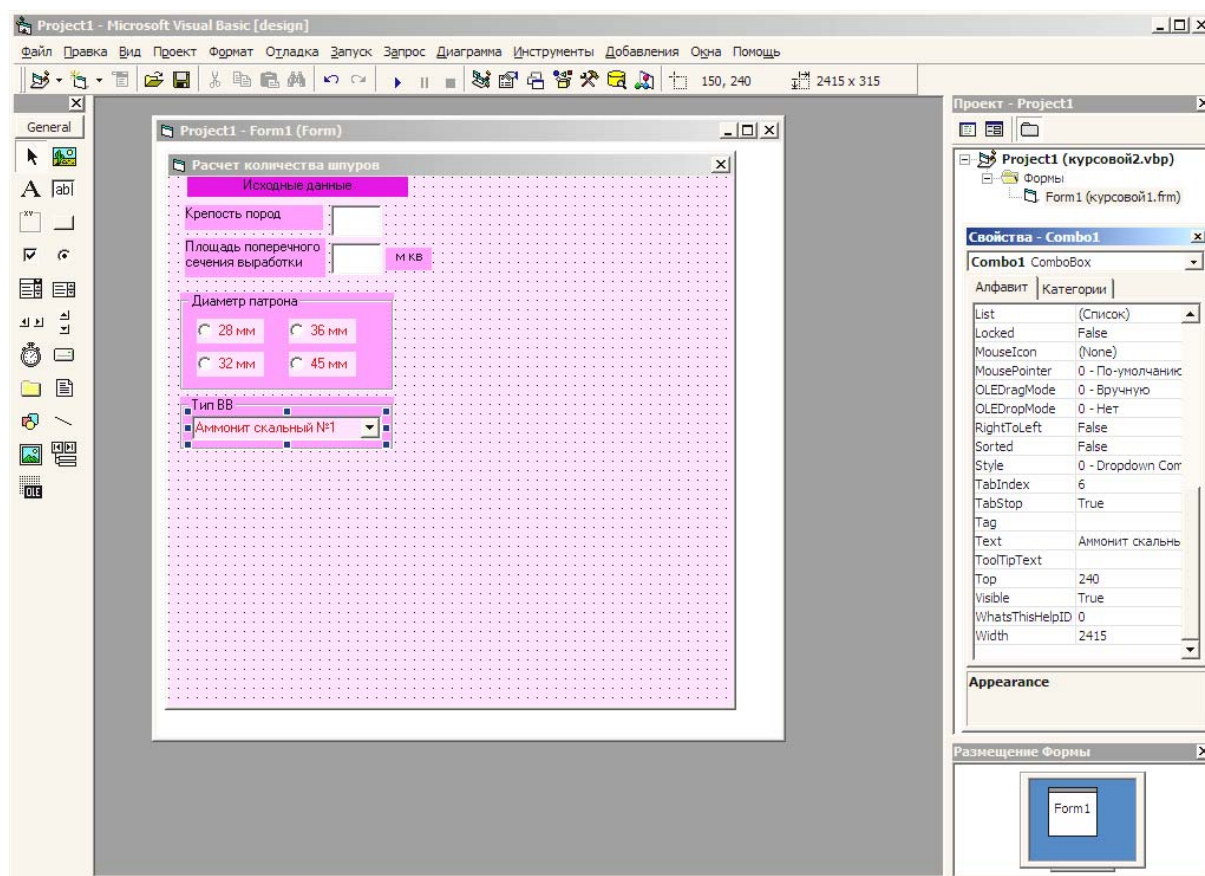
Далее на форму наносим метку Label3 с надписью «Площадь поперечного сечения выработки», рядом помещаем текстовое окно Text2 для ввода пользователем необходимого значения площади, а затем метку Label4 с указанием единицы измерения «м кв.».

Как было рассмотрено выше, диаметр патрона задаем с помощью кнопок-переключателей (Option Button). Для их логического объединения сначала наносим на форму рамку (Frame), которая на панели инструментов обозначается иконкой . В свойстве рамки Caption вводим текст «диаметр патрона». Размеры рамки определяем, исходя из размещения внутри нее 4 кнопок-переключателей, каждая из которых определяет один из возможных диаметров. Внутрь рамки с помощью инструмента  последовательно вносим 4 элемента Option Button и в свойстве Caption каждого из них задаем необходимый текст: «28 мм» для первого объекта, «32 мм» – для второго и т.д.

Последним из исходных данных, выносимым на форму (см. табл. 6.2), является тип ВВ, который удобнее ввести, используя инструмент «выпадающий список» (ComboBox), обозначаемый на панели инструментов иконкой . Для удобства пользователя предварительно нанесем на форму рамку (Frame) с заголовком (Caption) «Тип ВВ», а внутри нее разместим «выпадающий список». В свойство Text этого объекта можно внести наименование первого ВВ, которое будет приниматься по умолчанию, например, «аммонит скальный №1». Формирование полного списка для выбора из него нужного ВВ производится на стадии написания программного кода.




После нанесения на форму всех исходных данных, она примет вид, показанный на рис. 6.10.




**Рис. 6.10. Форма с нанесенными объектами для ввода исходных данных**

Аналогично вышеописанному порядку наносим все необходимые объекты для вывода результатов расчета. Это метки и текстовые окна, для вывода удельного расхода ВВ и количества шпуров на забой.

Остается нанести на форму последний блок объектов, с помощью которых пользователь будет управлять проектом. Это командные кнопки (Command Button), которые наносятся с помощью инструмента .

Для нанесения на командные кнопки надписей используется их свойство Caption. Вводим последовательно на кнопки надписи «Результат», «Сохранить», «Выход». Кроме этих кнопок, на форму наносим ряд дополнительных объектов, определяющих параметры сохранения результатов в файл. Это метки с надписями «Диск», «Имя файла», «Ф.И.О. студента», а также выпадающий список для выбора диска, на который будет производиться сохранение, текстовые окна для ввода имени файла и фамилии студентов.

Для того, чтобы разнообразить интерфейс программы, на форму можно наносить различные изображения или рисунки. Для наглядности процесса бурения шпуров в забое нанесем на форму рисунок с изображением бурильной установки, при этом используем объект PictureBox, наносимый на форму с помощью инструмента .

Готовый интерфейс приложения показан на рис. 6.11.

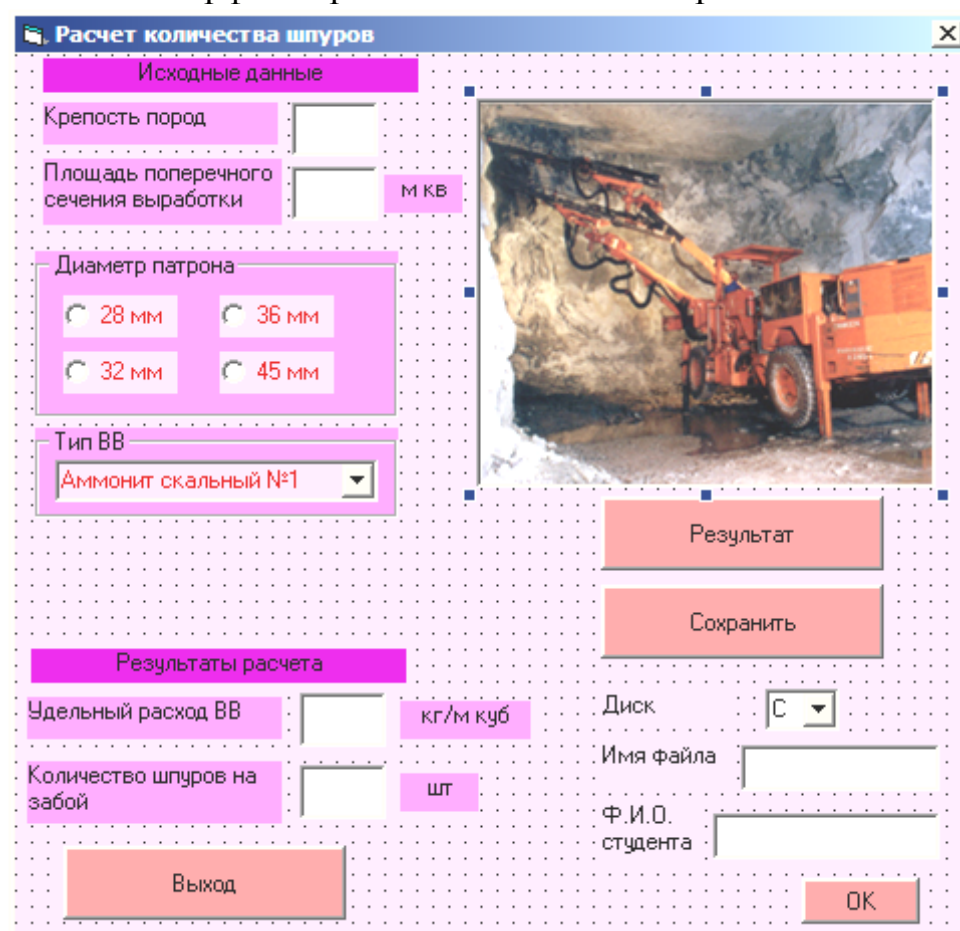


Рис. 6.11. Интерфейс приложения

**2-й этап** – разработка алгоритма. До начала написания программы программист должен четко определить последовательность всех выполняемых действий по вводу, преобразованию исходных данных, выводу и сохранению результатов, т.е. составить алгоритм (см. п. 6.1).

При составлении алгоритма очень важно следить за тем, чтобы все параметры, входящие в расчетную формулу были введены или определены ранее. Если значение некоторой переменной не задано, по умолчанию ей присваивается 0, что приводит к ошибочным результатам или сбою в выполнении программы.

Алгоритм рассматриваемого процесса заключается в следующем:

- считывание значений крепости пород  $F$  и площади поперечного сечения  $S$ , введенных пользователем в текстовые окна;
- определение диаметра патрона  $d$  в зависимости от выбранной пользователем кнопки-переключателя;
- определение коэффициента работоспособности  $e$  и плотности патронирования  $\rho$  в зависимости от выбранного пользователем из выпадающего списка типа ВВ;

– присвоение значений постоянным коэффициентам: структуры породы  $f_1$  и заполнения шпура  $\gamma$ ;

– проверка корректности исходных данных, введенных пользователем. Этот шаг необходим для исключения заведомо неверных значений, которые может ввести пользователь. Например, при задании крепости пород, меньшей 1 или большей 20, программа должна выдать сообщение об ошибке и обнулить введенные результаты. Аналогично при вводе площади сечения, меньшей 6 или большей 20 м<sup>2</sup>, программа должна выдать предупреждение пользователю о том, что данный алгоритм на заданную площадь не рассчитан. Во избежание сбоя программы, необходимо также выдать сообщение об ошибке в том случае, если пользователь не задал ни один из предложенных диаметров патрона;

– определение значения коэффициента использования шпура  $\eta$  в зависимости от крепости пород  $f$ ;

– расчет коэффициента зажима  $v = \frac{6,5}{\sqrt{S}}$ ;

– расчет нормального удельного расхода ВВ  $q_1 = 0,1f$ ;

– расчет удельного расхода ВВ  $q = q_1 \cdot f_1 \cdot v \cdot e$ ;

– расчет количества шпуров на забой  $N = 12,7 \cdot \frac{q \cdot S \cdot \eta}{\gamma \cdot d^2 \cdot \rho}$ ;

– вывод результатов расчета на экран (в текстовые поля) и при дополнительном запросе пользователя – в файл;

– выход из программы с запросом подтверждения о выходе.

**3-й этап** – составление таблицы идентификаторов и выбор типов переменных. Данный этап необходим для упорядочивания всех переменных, которые будут использованы в программе и фиксирования имен переменных (идентификаторов). Используемые символы для обозначений переменных в математических формулах не всегда допустимы или удобны в программном коде. Например, имя переменной не может содержать греческих букв, которые часто встречаются в формулах, а также верхних и нижних индексов. Одна и та же заглавная и строчная буква воспринимается программой как одна переменная, поэтому такие переменные нужно обозначать по-разному. При составлении списка идентификаторов удобно сразу определяться и с типом переменной.

Пример таблицы идентификаторов применительно к рассматриваемой задаче приведен в табл. 6.5. Здесь сопоставлены обозначения величин в формулах и принятые для них имена переменных. Вспомогательные переменные необходимы для управления проектом, выдачи сообщений об ошибках или запросов подтверждения выхода. Назначение вспомогательных переменных будет понятно в дальнейшем из программного кода.

Таблица 6.5

Таблица идентификаторов

Определяемый параметр	Обозначение в формуле	Имя переменной	Тип переменной
Основные переменные			
Крепость пород	$f$	F	Integer
Площадь поперечного сечения выработки	$S$	S	Single
Диаметр патрона ВВ	$d$	D	Integer
Коэффициент структуры породы	$f_1$	F1	Single
Коэффициент заполнения шпура	$\gamma$	GAMMA	Single
Коэффициент использования шпура	$\eta$	KISH	Single
Коэффициент работоспособности	$e$	E	Single
Плотность патронированного ВВ	$\rho$	RO	Single
Коэффициент зажима	$\nu$	NU	Single
Нормальный удельный расход ВВ	$q_1$	Q1	Single
Удельный расход ВВ	$q$	Q	Single
Количество шпуров на забой	$N$	N	Integer
Тип ВВ	–	TIPVV	String
Вспомогательные переменные			
Сообщение о недопустимо низкой крепости пород	–	a1	Integer
Сообщение о недопустимо высокой крепости пород	–	a2	Integer
Сообщение о недопустимой для данного алгоритма площади сечения	–	a3	Integer
Сообщение об отсутствии выбранного диаметра патрона	–	a4	Integer
Запрос подтверждения выхода из программы	–	a5	Integer
Имя диска для сохранения результатов	–	DISK	String
Имя файла для сохранения результатов	–	NAMEFILE	String
Фамилия, инициалы студента	–	FIO	String

Выбор типа переменной зависит от того, какие значения эта переменная будет принимать. Если возможны только целые значения (например, число шпуров, диаметр патрона), целесообразно задать тип Integer (целочисленный); если возможны дробные значения (площадь сечения, КИШ, плотность патрона, удельный расход ВВ и т.д.), то необходимо задавать тип Single (вещественный); если же переменная будет принимать не числовое, а буквенное значение (тип ВВ, фамилия студента, имя файла), то для нее нужно задать тип String (строковый). Все данные можно объединить одним общим типом – Variant. Подробно обо всех типах переменных в Visual Basic, их описании и преобразовании – см. в [6].

**4-й этап** – написание программного кода. Для «оживления» объектов, нанесенных на форму, т.е. для обеспечения функционирования командных кнопок и других объектов приложения необходимо в соответствии с разработанным алгоритмом написать программный код.

Целью настоящего пособия не является подробное рассмотрение всех (широчайших!) возможностей среды программирования Visual Basic, всех ее команд, операторов, процедур и т.д. Поэтому покажет только некоторые (наиболее простые и часто используемые) возможности среды и ограничимся краткими пояснениями, которые будут необходимы по ходу написания программы для решения этой конкретной задачи.

Итак, наш программный код будет состоять из отдельных подпрограмм, каждая из которых выполнится при определенном событии (действии пользователя). В данном примере таких событий будет 5, таким же будет количество подпрограмм.

*Первое событие* – это загрузка формы, т.е. самое начало работы приложения. Еще до того, как пользователь начнет вводить данные и нажимать на командные кнопки, форма полностью должна быть готова к этому, т.е. должны быть сформированы «выпадающие списки», должны быть видимыми нужные объекты и, наоборот, «спрятаны» пока ненужные.

Для того, чтобы начать писать подпрограмму, выполняемую при загрузке формы, необходимо дважды щелкнуть мышью в любом месте формы, свободном от объектов, при этом откроется окно ввода программного кода, в котором уже будут две готовые строки – первая и последняя:

```
Private Sub Form_Load()
```

```
End Sub
```

Что это означает? `Private Sub` – подпрограмма данного модуля, `Form` – имя объекта, в данном случае формы, `Load` – загрузка (событие), `End Sub` – конец подпрограммы, т.е. все действия, расположенные между этими строками, выполняются сразу при загрузке (`Load`) данной формы (`Form`).

При загрузке формы нам необходимо сформировать 2 «выпадающих списка» – `Combo1`, содержащий перечень ВВ для выбора, и `Combo2`, содержащий список дисков компьютера, на которые можно сохранять результаты. Для этого после имени списка `Combo1` через точку нужно поставить свойство `AddItem` (добавить) и затем в кавычках указать наименование ВВ, которое будет стоять в списке первым (по умолчанию). В следующей строке аналогично указывается второе ВВ списка и т.д. Во втором выпадающем списке `Combo2` также с помощью свойства `AddItem` добавляются имена дисков «С» и «D».

Данный фрагмент подпрограммы будет выглядеть следующим образом:

```
Private Sub Form_Load()  
Combo1.AddItem "Аммонит скальный №1"  
Combo1.AddItem "Детонит М"  
Combo1.AddItem "Аммонит АП-5ЖВ"  
Combo1.AddItem "Аммонит БЖВ"  
Combo2.AddItem "С"  
Combo2.AddItem "D"
```

Кроме формирования выпадающих списков с формы при ее загрузке необходимо убрать «лишние» объекты. Например, до нажатия пользователем командной кнопки «Сохранить», все объекты, определяющие место сохранения, имя файла, фамилию студента, на форме не нужны. Их можно сделать невидимыми с помощью свойства `Visible`, установив его значение для данных объектов равным `False`:

```
...
Label10.Visible = False
Label11.Visible = False
Label12.Visible = False
Combo2.Visible = False
Text5.Visible = False
Text6.Visible = False
Command4.Visible = False
End Sub
```

*Второе событие* – щелчок пользователя по командной кнопке `Command1` («Результат»). Подразумевается, что щелчок мышью по кнопке «Результат» пользователь произведет после того, как будут введены все необходимые исходные данные: задана крепость пород, площадь сечения выработки, выбраны диаметр патрона и тип ВВ. Поэтому данная подпрограмма должна предусматривать считывание исходных данных, проверку их корректности, все расчетные действия и вывод результатов (значений удельного расхода и количества шпуров на забой) в соответствующие текстовые блоки на форме.

Для начала написания программного кода, выполняемого при данном событии необходимо дважды щелкнуть на командную кнопку `Command1`, при этом опять открывается окно для ввода программного кода и появляются первая и последняя строки этой подпрограммы. На этот раз они будут выглядеть так:

```
Private Sub Command1_Click()

End Sub
```

т.е. данная подпрограмма (`Private Sub`) выполнится при однократном щелчке мыши (`Click`) по командной кнопке `Command1`.

Сначала необходимо объявить все переменные, которые будут использоваться в программе, для этого используется оператор `DIM`. (Следует отметить, что большое количество операторов и ключевых слов `Visual Basic` унаследовал от более старых версий Бэйсика, в частности от `Quick Basic`, подробные сведения о котором можно найти в [7,8]).

При объявлении переменных задаем их типы в соответствии с ранее обозначенными в табл. 6.5:

```
Rem описание переменных
Dim F, D, N, a1, a2, a3, a4, a5 As Integer
Dim S, F1, GAMMA, KISH, E, RO, NU, Q1, Q As Single
Dim TIPVV, DISK, NAMEFILE, FIO As String
```

Затем производим считывание исходных данных (крепости пород и площади сечения выработки), введенных пользователем в текстовые блоки Text1 и Text2. Для этого используем оператор Val, который преобразует текстовую информацию в числовые значения:

```
Rem ввод исходных данных
F = Val (Text1.Text)
S = Val (Text2.Text)
```

Таким образом, в результате выполнения данных строк переменная F получит значение, введенное пользователем в текстовый блок Text1, а переменная S – значение, введенное в Text2.

Для определения нужного значения диаметра патрона, который пользователь задаст с помощью кнопок-переключателей Option1 – Option4, используем их свойство Value, которое принимает значение True, если данная кнопка выбрана пользователем, и значение False – в противном случае. Проверку значения свойства Value производим с помощью строчного оператора IF – THEN (ЕСЛИ – ТО):

```
If Option1.Value = True Then D = 2.8
If Option2.Value = True Then D = 3.2
If Option3.Value = True Then D = 3.6
If Option4.Value = True Then D = 4.5
```

Таким образом, если пользователем выбрана кнопка Option1, то переменная D принимает значение 2,8, если Option2 – то значение 3,2 и т.д. (Здесь величина диаметра сразу задается в см, т.е. тех единицах, которые необходимо подставлять в расчетную формулу).

И последнее исходное данное, задаваемое пользователем, – это тип ВВ, который выбирается из выпадающего списка Combo1. В зависимости от этого программа должна определить значения коэффициента работоспособности ВВ и его плотности патронирования. Ввод этих данных осуществляем путем проверки значения свойства Text выпадающего списка Combo1, при этом используем блочную структуру оператора IF:

```
If Combo1.Text = "Аммонит скальный №1" Then
    E = 0.8
    RO = 1.1
    TIPVV = "Аммонит скальный №1"
ElseIf Combo1.Text = "Детонит М" Then
    E = 0.82
    RO = 1.15
    TIPVV = "Детонит М"
ElseIf Combo1.Text = "Аммонит АП-5ЖВ" Then
    E = 1.17
    RO = 1.15
    TIPVV = "Аммонит АП-5ЖВ"
ElseIf Combo1.Text = "Аммонит 6ЖВ" Then
    E = 1
    RO = 1.2
    TIPVV = "Аммонит АП-5ЖВ"
End If
```

Блочный IF работает следующим образом. Сначала проверяется первое условие, и если в выпадающем списке Combo1 пользователь оставил Text «Аммонит скальный №1», то выполнится блок операторов, следующий за данным условием, т.е. переменная E получит значение 0,8, переменная RO – значение 1,1, а строковая переменная тип ВВ – значение «Аммонит скальный №1». Если первое условие не выполняется, то программа пропускает следующий за ней блок операторов и переходит к проверке следующего условия, которое начинается с ключевого слова ElseIf. При выполнении этого условия (т.е. при выборе пользователем детонита М), переменные E, RO и TIPVV получают значения, соответствующие выбранному ВВ и т.д.

Заканчивается блочный IF оператором окончания проверки END IF.

Далее, согласно алгоритму, присваиваем значения постоянным коэффициентам:

F1 = 1.3

ГАММА = 0.5

Блок ввода исходных данных завершен. Произведем проверку корректности введенных пользователем данных. При этом используем пределы значений, оговоренных ранее для всех вводимых данных (см. табл. 6.2). В принципе, отсутствие данной проверки при корректности пользователя никак не скажется на ходе выполнения программы и полученных результатах. Однако опытные программисты всегда защищают свою программу от некорректных действий пользователя, т.е. выполняют «защиту от дурака».

В данном случае возможны только 3 ошибки пользователя:

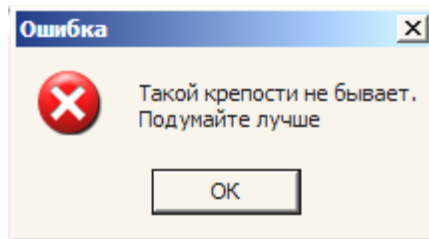
- неверное задание значения (слишком низкое или высокое) или вообще отсутствие задания крепости пород;
- задание площади выработки, для которой данный алгоритм не подходит (т.е. меньше 6 или больше 20 м<sup>2</sup>);
- отсутствие выбора диаметра ВВ (ни одна из кнопок-переключателей не выбрана).

Предусмотрим защиту программы от этих ошибок и организуем вывод сообщений пользователю о допущенных ошибках. Опять используем условный оператор проверки IF и команду вывода сообщений MsgBox. Например для исключения значений крепости ниже 1 (в том числе и полного отсутствия значения крепости) используем такие строки:

```
Rem проверка корректности исходных данных
If F < 1 Then
    a1 = MsgBox("Такой крепости не бывает." & Chr(13) & Chr(10)
    & "Подумайте лучше", 16, "Ошибка")
    Text1.Text = ""
End If
```

Это значит, что при значении крепости F, меньшем 1, текстовое поле Text1 обнулится (пустые кавычки после Text1.Text) и появляется следующее окошко сообщения MsgBox









Организация вывода сообщений с помощью системного окна MsgBox осуществляется следующим образом:

Имя переменной = MsgBox("текст сообщения", A+B+C, "заголовок окна"),

A – аргумент, определяющий, какая пиктограмма будет отображена на окне:

Значение A		Вид сообщения	Пиктограмма
числовое	буквенное		
16	vbCritical	Критическое сообщение	
32	vbQuestion	Вопрос	
48	vbExclamation	Предупреждение	
64	vbInformation	Информация	

B – аргумент, определяющий набор кнопок:

Значение B		Отображаемые кнопки
числовое	буквенное	
0	vbOKOnly	ОК
1	vbOKCancel	ОК, отмена
2	vbAbortRetry Ignore	Стоп, повтор, пропустить
3	vbYesNoCancel	Да, нет, отмена
4	vbYesNo	Да, нет
5	vbRetry Cancel	Повтор, отмена

C – аргумент, определяющий, какая кнопка принимается по умолчанию:

Значение C		Отображаемые кнопки
числовое	буквенное	
0	vbDefaultButton1	Первая кнопка – по умолчанию
256	vbDefaultButton2	Вторая кнопка – по умолчанию
2	vbDefaultButton3	Третья кнопка – по умолчанию

Применительно к вышеприведенному примеру:  
«Текст сообщения» – это «Такой крепости не бывает. Подумайте лучше»  
(символы Chr(13) и Chr(10) обозначают завершение строки и перенос части сообщения на следующую строку, значок & объединяет все части сообщения в единое целое):

$A + B + C = 16 + 0 + 0 = 16$  (т.е. будет отображаться пиктограмма критического сообщения ( $A=16$ ), одна кнопка ОК ( $B=0$ ) и она же принимается по умолчанию ( $C=0$ ).

«Заголовок окна» – в него выводится комментарий сообщения, в данном случае слово «Ошибка».

Кроме числовых значений переменные  $A$ ,  $B$  и  $C$  могут принимать буквенные, например:

$D = \text{MsgBox}(\text{"Сохранить результаты?"}, \text{vbQuestion} + \text{vbYesNoCancel} + \text{vbDefaultButton2}, \text{"Вопрос"})$ .

В этом случае на сообщении появится пиктограмма «Вопрос», три кнопки (Да, Нет и Отмена) и вторая кнопка (Нет) будет приниматься по умолчанию.

В зависимости от того, какая из предложенных кнопок будет нажата пользователем, переменная, которой присвоено сообщение `MsgBox` (в последнем случае переменная  $D$ , в первом примере – переменная  $a1$ ), будет принимать одно из следующих значений

Значение переменной сообщения		Выбранная кнопка
числовое	буквенное	
1	vbOK	ОК
2	vbCancel	отмена
3	vbAbort	Стоп
4	vbRetry	Повтор
5	vbIgnore	Пропустить
6	vbYes	Да
7	vbNo	Нет

Продолжим проверку корректности исходных данных. Если пользователь введет значение крепости пород, большее 20 (что при проведении горных выработок практически не встречается), программа выдаст следующее сообщение:

```
If F > 20 Then
    a2 = MsgBox(" Действительно ли это крепость породы, " &
Chr(13) & Chr(10) & "а не напиток?", 36, "Вопрос")
    If a2 = 7 Then
        Text1.Text = ""
    End If
```

В данном случае значения  $A + B + C = 32 + 4 + 0 = 36$ , поэтому на окне сообщения отобразится пиктограмма «Вопрос» и две кнопки «Да» и «Нет», при этом по умолчанию будет приниматься первая из них. В случае нажатия на кнопку «Нет» (т.е. признания пользователем своей ошибки) переменная  $a2$  принимает значение 7 (см. последнюю таблицу) и в этом случае текстовое окно `Text1` обнуляется. Если же пользователь нажал кнопку «Да», т.е. подтвердил заданное значение крепости свыше 20, то окно автоматически закрывается и расчет производится с этим значением крепости.

Следующее сообщение об ошибке появится при неверном задании площади поперечного сечения. Здесь мы используем сложное условие, объединенное логическими операторами OR (ИЛИ) и AND (И):

```
If S <> 0 And (S < 6 Or S > 20) Then
  a3 = MsgBox("Для указанной площади сечения" & Chr(13) &
  Chr(10) & "данная программа не рассчитана", 48, "предупрежде-
  ние")
  Text2.Text = ""
End If
```

Данное сообщение появится только в том случае, если площадь S будет меньше 6 ИЛИ (OR) больше 20, И (AND) при этом вообще будет задана (т.е. значение S не равно нулю). На сообщении будет пиктограмма «Предупреждение» (A=48) и одна кнопка ОК (B=0).

И последнее сообщение появится в том случае, если пользователь забыл выбрать диаметр патрона, т.е. не выбран ни один из элементов Option:

```
If Option1.Value = False And Option2.Value = False And
Option3.Value = False And Option4.Value = False Then
  a4 = MsgBox("Не задан диаметр патрона!", 16, "Ошибка")
End If
```

После проверки корректности исходных данных приступаем к непосредственному расчету требуемых параметров и выводу результатов на форму.

По значению крепости определяем КИШ:

```
Rem расчет
If F <= 4 Then KISH = 0.9
If F > 4 And F <= 10 Then KISH = 0.85
If F > 10 Then KISH = 0.8
```

и последовательно рассчитываем коэффициент зажима, нормальный удельный расход ВВ, удельный расход ВВ и количество шпуров на забой:

```
NU = 6.5 / Sqr(S)
Q1 = 0.1 * F
Q = Q1 * F1 * NU * E
N = 12.7 * Q * S * KISH / (GAMMA * D ^ 2 * RO)
```

Последние два параметра выводим на форму в подготовленные для них текстовые окна Text3 и Text4:

```
Rem Вывод результатов на экран
Text3.Text = Format(Q, "0.000")
Text4.Text = Format(N, "###")
End Sub
```

При выводе по формату в скобках задается имя выводимой переменной и через запятую в кавычках собственно формат вывода – сколько знаков выделяется до и после запятой. Так, для удельного расхода Q выделен 1 знак для целой и 3 знака для дробной части, при этом в случае появления

нуля в целой части или в конце дробной части он будет отображаться. Если вместо символа «0» в формате указать символ «#», то нули в вышеназванных позициях будут опущены. Так, для переменной N, определяющей число шпуров выделено 3 знака в целой части и в случае получения в результате одно- или двузначного числа первые нули отображаться не будут.

Оператор End Sub закрывает подпрограмму, которая выполнится при однократном щелчке мыши по командной кнопке Command1.

*Третье событие* – щелчок пользователя по командной кнопке Command2 (Выход). При этом необходимо выполнить запрос на подтверждение выхода из программы, который можно организовать с помощью уже известного окна сообщения MsgBox:

```
Private Sub Command2_Click()  
    a4 = MsgBox("Неужели Вы хотите выйти" & Chr(13) & Chr(10) &  
    "из этой прекрасной программы?", 36, "Вопрос")  
    If a4 = 6 Then Unload Me  
End Sub
```

В данном случае при нажатии пользователем на кнопку «Да» переменная a4 принимает значение 6 и происходит выгрузка формы (Unload Me), в противном случае – сообщение закрывается, а выполнение программы продолжается.

*Четвертое событие* – щелчок пользователя по командной кнопке Command3 («Сохранить»). В этом случае у пользователя нужно запросить дополнительную информацию о том, на какой диск и в файл с каким именем следует сохранить исходные данные и результаты, а также фамилию и инициалы пользователя для их последующего вывода в файл.

Для этого следует отобразить все ранее «спрятанные» (при загрузке формы) объекты:

```
Private Sub Command3_Click()  
Label10.Visible = True  
Label11.Visible = True  
Label12.Visible = True  
Combo2.Visible = True  
Text5.Visible = True  
Text6.Visible = True  
Command4.Visible = True  
End Sub
```

На форме появятся окна для ввода всех необходимых данных и командная кнопка ОК для подтверждения сохранения результатов.

*Пятое событие* – щелчок пользователя по командной кнопке Command4 («ОК»). В результате этого события все исходные данные и результаты должны быть сохранены в файл. Но сначала дублируем задание ос-

НОВНЫХ ИСХОДНЫХ ДАННЫХ И РАСЧЕТ ПОКАЗАТЕЛЕЙ, КОТОРЫЕ ЗАТЕМ БУДУТ ВЫВОДИТЬСЯ В ФАЙЛ И СОХРАНЯТЬСЯ:

```
Private Sub Command4_Click()

Dim F, D, N As Integer
Dim S, F1, GAMMA, KISH, E, RO, NU, Q1, Q As Single
Dim TIPVV, DISK, NAMEFILE, FIO As String

Rem ввод исходных данных
If Option1.Value = True Then D = 2.8
If Option2.Value = True Then D = 3.2
If Option3.Value = True Then D = 3.6
If Option4.Value = True Then D = 4.5

If Combo1.Text = "Аммонит скальный №1" Then
    E = 0.8
    RO = 1.1
    TIPVV = "Аммонит скальный №1"
ElseIf Combo1.Text = "Детонит М" Then
    E = 0.82
    RO = 1.15
    TIPVV = "Детонит М"
ElseIf Combo1.Text = "Аммонит АП-5ЖВ" Then
    E = 1.17
    RO = 1.15
    TIPVV = "Аммонит АП-5ЖВ"
ElseIf Combo1.Text = "Аммонит 6ЖВ" Then
    E = 1
    RO = 1.2
    TIPVV = "Аммонит АП-5ЖВ"
End If
```

Затем производим считывание данных о том, куда и под каким именем будет производиться сохранение: сначала проверяем из выпадающего списка имя диска (С или D), из текстового окна Text5 считываем имя файла и из текстового окна Text5 – фамилию и инициалы пользователя:

```
If Combo2.Text = "C" Then DISK = "C"
If Combo2.Text = "D" Then DISK = "D"
NAMEFILE = Text5.Text
FIO = Text6.Text
```

После этого производим непосредственное сохранение результатов, предварительно открыв нужный файл с помощью оператора OPEN, после которого указывается путь к файлу (в данном случае это только выбранный пользователем диск) и заданное имя файла, к которому программа добавит расширение txt (текстовый). Если к моменту сохранения такого файла не окажется на указанном диске, он будет автоматически создан. В конце строки указывается, для чего и под каким номером открыт файл. В данном случае – For Output (для вывода) As #1 (под номером 1):

```

Rem вывод результатов в файл
Open DISK + ":\\" + NAMEFILE + ".txt" For Output As #1
Print #1, Tab(20), "Расчет количества шпуров на забой"
Print #1,
Print #1, Tab(20), "Исходные данные"
Print #1,
Print #1, Tab(20), "Крепость пород - "; Text1.Text
Print #1, Tab(20), "Площадь поперечного сечения -"; Text2.Text;" м кв."
Print #1, Tab(20), "Характеристика ВВ:"
Print #1, Tab(50), "- тип ВВ - "; TIPVV
Print #1, Tab(50), "- диаметр патрона - "; 10 * D; " мм"
Print #1, Tab(50), "- коэффициент работоспособности - "; E
Print #1, Tab(50), "- плотность патронирования - "; RO; " кг/м куб."
Print #1,
Print #1, Tab(20), "Результаты расчета"
Print #1,
Print #1, Tab(20), "Удельный расход ВВ - "; Text3.Text; " кг/м куб"
Print #1, Tab(20), "Количество шпуров на забой - "; Text4.Text;" шт."
Print #1,
Print #1, Tab(20), "Расчет выполнил студент "; FIO
Close #1
End Sub

```

Формирование строк вывода внутри файла осуществляется оператором Print #1, после которого указываются строки и переменные вывода. Вспомогательный оператор Tab показывает, с какой позиции от начала строки необходимо начинать вывод.

Если все вышеописанные фрагменты слить воедино, получится следующий программный код, который обеспечит работу всего приложения:

```

Private Sub Form_Load()
Combo1.AddItem "Аммонит скальный №1"
Combo1.AddItem "Детонит М"
Combo1.AddItem "Аммонит АП-5ЖВ"
Combo1.AddItem "Аммонит 6ЖВ"
Combo2.AddItem "С"
Combo2.AddItem "D"
Label10.Visible = False
Label11.Visible = False
Label12.Visible = False
Combo2.Visible = False
Text5.Visible = False
Text6.Visible = False
Command4.Visible = False
End Sub

Private Sub Command1_Click()
Rem описание переменных
Dim F, D, N, a1, a2, a3, a4, a5 As Integer
Dim S, F1, GAMMA, KISH, E, RO, NU, Q1, Q As Single
Dim TIPVV, DISK, NAMEFILE, FIO As String

Rem ввод исходных данных
F = Val(Text1.Text)
S = Val(Text2.Text)

```

```

If Option1.Value = True Then D = 2.8
If Option2.Value = True Then D = 3.2
If Option3.Value = True Then D = 3.6
If Option4.Value = True Then D = 4.5
If Combol.Text = "АММОНИТ скальный №1" Then
    E = 0.8
    RO = 1.1
    TIPVV = "АММОНИТ скальный №1"
ElseIf Combol.Text = "Детонит М" Then
    E = 0.82
    RO = 1.15
    TIPVV = "Детонит М"
ElseIf Combol.Text = "Аммонит АП-5ЖВ" Then
    E = 1.17
    RO = 1.15
    TIPVV = "Аммонит АП-5ЖВ"
ElseIf Combol.Text = "Аммонит 6ЖВ" Then
    E = 1
    RO = 1.2
    TIPVV = "Аммонит АП-5ЖВ"
End If
F1 = 1.3
ГАММА = 0.5

Rem проверка корректности исходных данных
If F < 1 Then
    a1 = MsgBox("Такой крепости не бывает." & Chr(13) & Chr(10) & "Подумайте
лучше", 16, "Ошибка")
    Text1.Text = ""
End If
If F > 20 Then
    a2 = MsgBox("Действительно ли это крепость породы," & Chr(13) & Chr(10)
& "а не напитка?", 36, "Вопрос")
    If a2 = 7 Then
        Text1.Text = ""
    End If
End If
If S <> 0 And (S < 6 Or S > 20) Then
    a3 = MsgBox("Для указанной площади сечения" & Chr(13) & Chr(10) & "дан-
ная программа не рассчитана", 48, "предупреждение")
    Text2.Text = ""
End If
If Option1.Value = False And Option2.Value = False And Option3.Value =
False And Option4.Value = False Then
    a4 = MsgBox("Не задан диаметр патрона!", 16, "Ошибка")
End If

Rem расчет
If F <= 4 Then KISH = 0.9
If F > 4 And F <= 10 Then KISH = 0.85
If F > 10 Then KISH = 0.8
NU = 6.5 / Sqr(S)
Q1 = 0.1 * F
Q = Q1 * F1 * NU * E
N = 12.7 * Q * S * KISH / (ГАММА * D ^ 2 * RO)

Rem Вывод результатов на экран
Text3.Text = Format(Q, "0.000")
Text4.Text = Format(N, "###")

End Sub

```

```

Private Sub Command2_Click()
    a4 = MsgBox("Неужели Вы хотите выйти" & Chr(13) & Chr(10) & "из этой пре-
красной программы?", 36, "Вопрос")
    If a4 = 6 Then Unload Me
End Sub

Private Sub Command3_Click()
Label10.Visible = True
Label11.Visible = True
Label12.Visible = True
Combo2.Visible = True
Text5.Visible = True
Text6.Visible = True
Command4.Visible = True
End Sub

Private Sub Command4_Click()

Dim F, D, N As Integer
Dim S, F1, GAMMA, KISH, E, RO, NU, Q1, Q As Single
Dim TIPVV, DISK, NAMEFILE, FIO As String

Rem ввод исходных данных
If Option1.Value = True Then D = 2.8
If Option2.Value = True Then D = 3.2
If Option3.Value = True Then D = 3.6
If Option4.Value = True Then D = 4.5

If Combo1.Text = "Аммонит скальный №1" Then
    E = 0.8
    RO = 1.1
    TIPVV = "Аммонит скальный №1"
ElseIf Combo1.Text = "Детонит М" Then
    E = 0.82
    RO = 1.15
    TIPVV = "Детонит М"
ElseIf Combo1.Text = "Аммонит АП-5ЖВ" Then
    E = 1.17
    RO = 1.15
    TIPVV = "Аммонит АП-5ЖВ"
ElseIf Combo1.Text = "Аммонит 6ЖВ" Then
    E = 1
    RO = 1.2
    TIPVV = "Аммонит АП-5ЖВ"
End If

If Combo2.Text = "С" Then DISK = "С"
If Combo2.Text = "D" Then DISK = "D"
NAMEFILE = Text5.Text
FIO = Text6.Text

Rem вывод результатов в файл
Open DISK + ":\\" + NAMEFILE + ".txt" For Output As #1
Print #1, Tab(20), "Расчет количества шпуров на забой"
Print #1,
Print #1, Tab(20), "Исходные данные"
Print #1,
Print #1, Tab(20), "Крепость пород - "; Text1.Text
Print #1, Tab(20), "Площадь поперечного сечения - "; Text2.Text; " м кв."
Print #1, Tab(20), "Характеристика ВВ:"
Print #1, Tab(50), "- тип ВВ - "; TIPVV

```



```

Print #1, Tab(50), "- диаметр патрона - "; 10 * D; " мм"
Print #1, Tab(50), "- коэффициент работоспособности - "; E
Print #1, Tab(50), "- плотность патронирования - "; RO; " кг/м куб."
Print #1,
Print #1, Tab(20), "Результаты расчета"
Print #1,
Print #1, Tab(20), "Удельный расход ВВ - "; Text3.Text; " кг/м куб"
Print #1, Tab(20), "Количество шпуров на забой - "; Text4.Text; " шт."
Print #1,
Print #1, Tab(20), "Расчет выполнил студент "; FIO
Close #1
End Sub

```

Пример окна выполняемого приложения приведен на рис. 6.12.

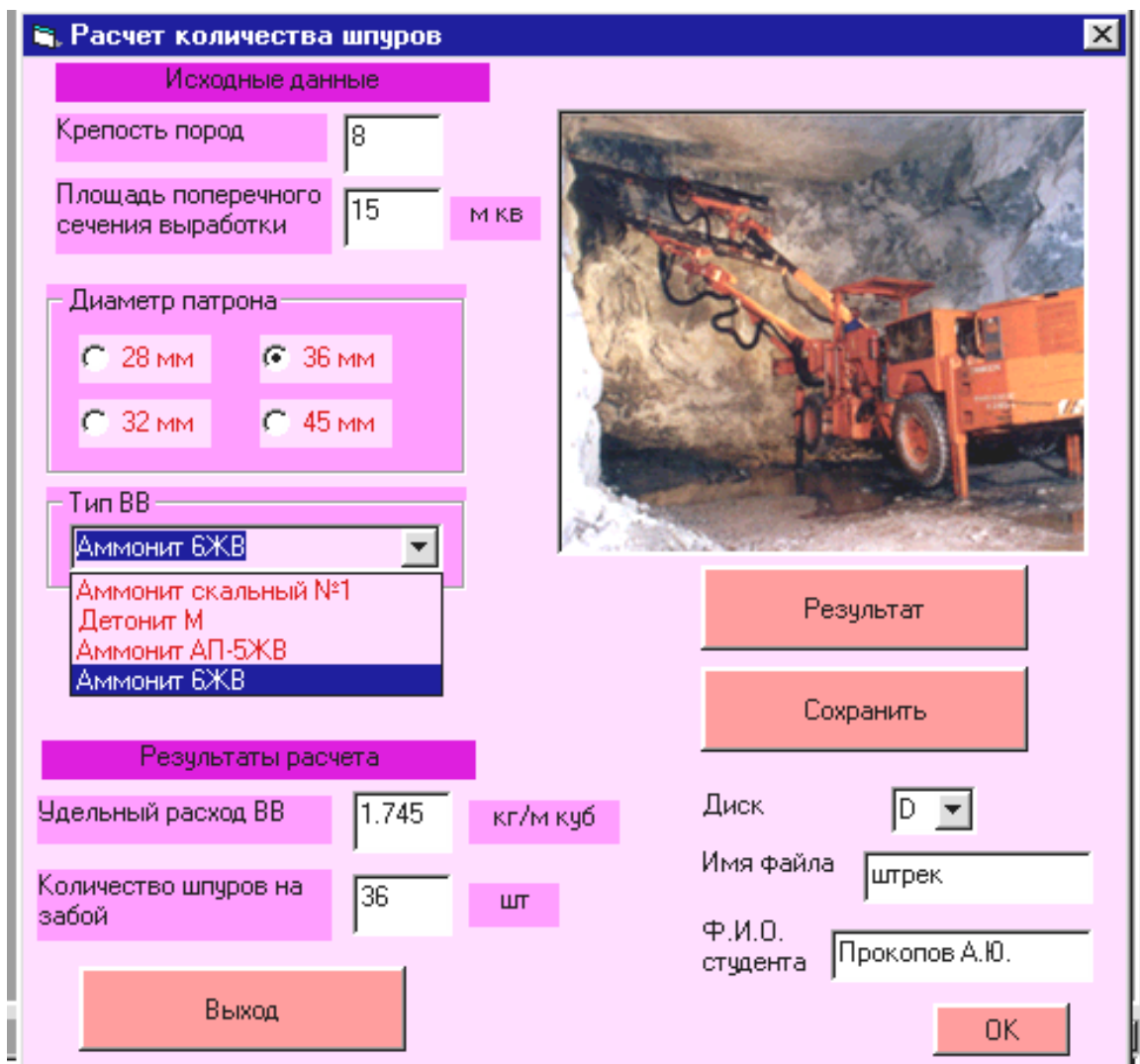


Рис. 6.12. Пример работы приложения

### Контрольные задания

Разработать Windows-приложения по определению заданных параметров. Программный код должен обеспечивать расчет для любых вариантов исходных данных. Предусмотреть возможность вывода результатов выполнения программы на экран и в файл.

1. По заданной производственной мощности шахты определить нормативную продолжительность подготовительного периода строительства согласно приведенной таблице:

Производственная мощность шахты, млн т / год	0,6	0,9	1,2	1,5	1,8	2,1	2,4	3,0	3,6
Нормативная продолжительность подготовительного периода, мес.	12	12	15	15	15	16	16	16	16

Для случая задания пользователем производственной мощности, отличной от типовой, предусмотреть выдачу сообщения о некорректности исходных данных и возможность повторного выбора производственной мощности шахты.

2. Проверить соблюдение запаса прочности каната подъемной машины по условию:

$$Z_g = \frac{\Sigma F_p}{Q_\Gamma + Q_c + m_k \cdot H_0} \geq Z_1,$$

где  $Z_1$  – запас прочности каната, принять  $Z_1 = 10$ ;

$\Sigma F_p$  – суммарное разрывное усилие всех проволок в канате, Н;

$Q_\Gamma$  – масса породы и воды в бадье, кг,

$$Q_\Gamma = V_b \cdot \gamma_\Pi + \left( V_b - \frac{V_b}{k_p} \right) \cdot \gamma_v \cdot k_3;$$

$V_b$  – вместимость бадьи, м<sup>3</sup>;

$\gamma_\Pi$  – плотность породы в разрыхленном состоянии, кг/м<sup>3</sup>;

$k_p$  – коэффициент разрыхления породы,  $k_p = 1,8 \div 2$ ;

$\gamma_v$  – плотность воды, кг/м<sup>3</sup>;

$k_3$  – коэффициент заполнения пустот водой,  $k_3 = 0,5$ ;

$m_k$  – масса 1 м каната, определяемая из соотношения, кг,

$$m_k = \frac{Q_\Gamma + Q_c}{\frac{\sigma}{z_1 \cdot \gamma_0} - H_0};$$

$\sigma$  – предел прочности материала проволок каната при растяжении, МПа,  $\sigma = 1500$ ;

$\gamma_0$  – фиктивный объемный вес материала каната, кН/м<sup>3</sup>,  $\gamma_0 = 90$ ;

$H_0$  – максимальная длина отвеса каната, м,  $H_0 = H_{ст} + h_k$ ;

$H_{ст}$  – конечная глубина ствола, м;

$h_k$  – высота копра, м;

$Q_c$  – масса бадьи с направляющей рамой и прицепным устройством, кг,

$Q_c = Q_b + Q_{н.р} + Q_{п.у}$ , определяемая по следующей таблице в зависимости от типа бадьи:

Тип бадьи	Вместимость $V$ , $\text{м}^3$	Масса бадьи $Q_6$ , кг	Масса направ- ляющей рамы $Q_{н.р.}$ , кг	Масса прицепного устройства $Q_{п.у.}$ , кг
БПС-1	1,0	356	394	92
БПС-1,5	1,5	605	590	92
БПС-2	2,0	730	590	92
БПС-2,5	2,5	878	600	97
БПС-3	3,0	938	600	97
БПС-4	4,0	1465	835	97
БПС-5	5,0	1696	1000	118

Расчет произвести для следующих данных:  $\Sigma F_p = 268000 \text{ Н}$ ,  $H_{ст} = 450 \text{ м}$ ,  $h_k = 25 \text{ м}$ ,  $\gamma_n = 1600 \text{ кг/м}^3$ . Предусмотреть ручной ввод типа применяемой бадьи. На экран вывести тип бадьи, концевую нагрузку, величину запаса прочности и сообщение о соблюдении или несоблюдении требуемого запаса прочности.

3. Произвести выбор вместимости проходческой бадьи  $V_6$ ,  $\text{м}^3$ , по продолжительности полного цикла подъема бадьи по стволу и производительности подъемной установки, используя формулу

$$V_6 = \frac{P_n \cdot t_{ц}}{3600 \cdot k_6},$$

где  $P_n$  – необходимая производительность подъемной установки,  $\text{м}^3/\text{ч}$ ,  
 $P_n = \min\{P_{п1}; P_{п2}\}$ ;

$P_{п1}$  – производительность подъемной установки, определяемая исходя из заданной технической скорости проходки ствола,

$$P_{п1} = \frac{v_T \cdot S_{пр} \cdot k_n \cdot k_p}{t \cdot m};$$

$v_T$  – заданная скорость проходки ствола,  $\text{м/мес}$ ;

$S_{пр}$  – площадь сечения ствола в проходке,  $\text{м}^2$ ,  $S_{пр} = \frac{\pi \cdot D_{пр}^2}{4}$ ;

$k_n$  – коэффициент неравномерности работы подъема,  $k_n = 1,15$ ;

$k_p$  – коэффициент разрыхления породы,  $k_p = 1,8 \div 2$ ;

$t$  – продолжительность работы подъема в сутки по выдаче породы, ч;

$m$  – число рабочих дней в месяце по проходке ствола;

$P_{п2}$  – производительность подъемных установок, определяемая исходя из возможной производительности погрузочных машин в забое  $P_p$ ,  $\text{м}^3/\text{ч}$ ,  $P_{п2} = 1,15 \cdot P_p$ ;

$t_{ц}$  – продолжительность полного цикла подъема бадьи по стволу, с;

$k_6$  – коэффициент заполнения бадьи,  $k_6 = 0,9$ .

По расчетной вместимости бадьи принять ближайшую большую вместимость бадьи из следующего ряда значений:

0,5; 0,75; 1,0; 1,5; 2,0; 2,5; 3,0; 4,0; 5,0; 6,5

Расчет произвести для следующих исходных данных:  $t_{ц} = 400$  с;  $t = 14$  ч;  $m = 25$ . Предусмотреть ручной ввод параметров  $v_{т}$ ,  $D_{пр}$  и  $P_{р}$ . На печать вывести расчетное и принятое из типового ряда значения вместимости бабды и значения производительности подъема  $P_{п1}$  и  $P_{п2}$ .

4. Рассчитать производительность погрузочной машины  $Q$ , м<sup>3</sup>/ч, при различных технологических схемах транспортирования горной массы из забоя и определить схему, при которой производительность будет максимальной:

$$Q = \max \{Q_{в}, Q_{п}, Q_{к}\},$$

где  $Q_{в}$ ,  $Q_{п}$ ,  $Q_{к}$  – производительность погрузочной машины соответственно при погрузке породы в одиночные вагонетки, при применении перегружателей и при транспортировании породы конвейером, м<sup>3</sup>/ч. Данные параметры определяются следующими формулами:

$$Q_{в} = \frac{1}{\varphi \cdot \alpha \cdot \left( \frac{1}{Q_{т}} + \frac{t_3}{V_{в} \cdot k_3} \right) + \frac{(1 - \alpha) \cdot \varphi}{n_{р} \cdot P_{п}}};$$

$$Q_{п} = \frac{1}{\varphi \cdot \alpha \cdot \left( \frac{1}{Q_{т}} + \frac{t_c}{V_{в} \cdot k_3 \cdot n_c} \right) + \frac{(1 - \alpha) \cdot \varphi}{n_{р} \cdot P_{п}}};$$

$$Q_{к} = \frac{1}{\frac{\varphi \cdot \alpha}{Q_{т}} + \frac{(1 - \alpha) \cdot \varphi}{n_{р} \cdot P_{п}}},$$

где  $\varphi$  – коэффициент, учитывающий технологические простои машины,  $\varphi = 1,15 \div 1,20$ ;

$\alpha$  – доля объема породы первой фазы,  $\alpha = 0,85 \div 0,90$ ;

$Q_{т}$  – техническая производительность погрузочной машины, м<sup>3</sup>/ч;

$V_{в}$  – объем вагонетки, м<sup>3</sup>;

$k_3$  – коэффициент заполнения вагонетки,  $k_3 = 0,9$ ;

$t_3$  – время замены груженной вагонетки на порожнюю, ч;

$t_c$  – время замены груженной партии вагонеток на порожнюю, ч;

$n_c$  – число вагонеток, установленных под перегружателем;

$n_{р}$  – число рабочих, занятых на подкидке породы;

$P_{п}$  – производительность рабочего на подкидке породы, м<sup>3</sup>/ч, принимаемая по следующей таблице в зависимости от крепости пород:

$f$	< 4	4 ÷ 7	> 7
$P_{п}$ , м <sup>3</sup> /ч	1,1	0,9	0,8

Расчет выполнить для следующих исходных данных:  $t_3 = 0,02$  ч;  $t_c = 0,03$  ч;  $n_c = 5$ ;  $n_p = 2$ . Предусмотреть ручной ввод параметров  $Q_T, V_B, f$ . На печать вывести три значения производительности и сообщение о способе транспортировки породы, при котором получена максимальная производительность.

5. Сравнить сменную производительность скреперной установки, при разгрузке горной массы на конвейер  $Q_K$  и в одиночные вагонетки  $Q_B$  и выбрать схему транспорта, обеспечивающую максимальную производительность  $Q$ , м<sup>3</sup>/смен:

$$Q = \max \{ Q_K, Q_B \},$$

где  $Q_K$  и  $Q_B$  определяются формулами:

$$Q_K = \frac{(T - t_{п.з} - t_{л}) \cdot V_B \cdot k_{3.с}}{\left( \frac{l}{v_{Г}} + \frac{l}{v_{П}} + t_{3.р} \right) \cdot k_p},$$

$$Q_B = \frac{(T - t_{п.з} - t_{л}) \cdot V_B \cdot k_{3.В}}{\frac{V_B \cdot k_{3.В}}{V \cdot k_p} \cdot \left( \frac{l}{v_{Г}} + \frac{l}{v_{П}} + t_{3.р} \right) + \frac{2 \cdot L}{v_B} + t_p};$$

$T$  – продолжительность смены, мин,  $T = 360$ ;

$t_{п.з}$  – продолжительность подготовительно-заключительных операций, мин,  $t_{п.з} = 40$ ;

$t_{л}$  – продолжительность опробования лебедки, мин,  $t_{л} = 10$ ;

$V$  – объем скрепера, м<sup>3</sup>;

$k_{3.с}$  – коэффициент заполнения скрепера, принимаемый равным:  
для крупнокусковатой горной массы –  $0,5 \div 0,7$ , для средней –  $0,7 \div 0,8$ , для мелкой –  $0,8 \div 1$ ;

$l$  – расстояние скреперования, м;

$v_{Г}$  и  $v_{П}$  – соответственно скорости движения груженого и порожнего скрепера, м/мин;

$t_{3.р}$  – время загрузки и разгрузки скрепера с учетом пауз на переключение и неравномерности хода, мин;

$k_p$  – коэффициент разрыхления породы,  $k_p = 1,8 \div 2$ ;

$V_B$  – объем вагонетки, м<sup>3</sup>;

$k_{3.В}$  – коэффициент заполнения вагонетки,  $k_{3.В} = 0,9$ ;

$L$  – длина транспортировки породы в вагонетке, м;

$v_B$  – скорость движения вагонетки, м/мин;

$t_p$  – время разгрузки вагонетки, мин.

Расчет произвести для следующих исходных данных:  $l = 30$  м,  $v_{Г} = 66$  м/мин,  $v_{П} = 90$  м/мин,  $t_{3.р} = 0,7$  мин,  $L = 50$  м,  $v_B = 120$  м/мин,  $t_p =$

1 мин. Организовать ручной ввод параметров  $V$ ,  $V_B$  и степень кусковатости горной массы (крупная, средняя, мелкая). На печать вывести значения производительности для обеих схем транспортирования горной массы и сообщение о более эффективной схеме.

6. Определить необходимое количество воздуха  $Q_{з.п}$ , м<sup>3</sup>/мин, для проветривания тупиковой выработки по всем указанным факторам. В качестве расчетного принять максимальное из полученных значений

$$Q_{з.п} = \max \{Q_1, Q_2, Q_3, Q_4, Q_5\},$$

где  $Q_1$  – расход воздуха в забое по выделению метана, м<sup>3</sup>/мин,

$$Q_1 = \frac{100 \cdot I_{з.п.}}{C - C_0};$$

$I_{з.п.}$  – суммарное метановыделение с обнаженной поверхности угольного пласта и из отбитого угля, м<sup>3</sup>/мин;

$C$  – максимально допустимое по ПБ содержание метана в исходящей вентиляционной струе, %,  $C = 1$ ;

$C_0$  – содержание метана в поступающей вентиляционной струе, %;

$Q_2$  – расход воздуха в забое по максимальному числу людей, одновременно работающих в забое выработки, м<sup>3</sup>/мин,

$$Q_2 = 6 \cdot n_{л};$$

$n_{л}$  – максимальное число людей, одновременно работающих в забое выработки;

$Q_3$  – расход воздуха в забое по минимальной скорости движения воздуха, м<sup>3</sup>/мин,

$$Q_3 = 60 \cdot V_{min} \cdot S_{св};$$

$V_{min}$  – минимально допустимая скорость движения воздуха в призабойном пространстве, м/с,  $V_{min} = 0,15$ ;

$S_{св}$  – площадь поперечного сечения выработки в свету, м<sup>2</sup>;

$Q_4$  – расход воздуха в забое по тепловому фактору, м<sup>3</sup>/мин,

$$Q_4 = 20 \cdot V_{n.min} \cdot S_{св};$$

$V_{n.min}$  – минимально допустимая скорость движения воздуха в выработке по тепловому фактору, м/с;

$Q_5$  – расход воздуха в забое по количеству одновременно взрывающегося ВВ, м<sup>3</sup>/мин,

$$Q_5 = \frac{2,25}{T} \cdot \sqrt[3]{\frac{V_{ВВ} \cdot S_{св}^2 \cdot l_{п}^2 \cdot K_{обв}}{K_{ут.тр}^2}};$$

$T$  – время проветривания выработки после взрывания, мин;

$V_{ВВ}$  – объем вредных газов, образующихся после взрывания, л,

$$V_{ВВ} = 100 \cdot B_{уг} + 40 \cdot B_{пор};$$

$B_{уг}$ ,  $B_{пор}$  – масса одновременно взрывааемых ВВ по углю и породе, соответственно, кг;

$l_{п}$  – длина тупиковой выработки, м, для горизонтальных и наклонных выработок протяженностью 500 м и более вместо  $l_{п}$  подставляется ее критическая длина  $l_{кр} = 500$ ;

$K_{обв}$  – коэффициент, учитывающий обводненность выработки и принимаемый равным 0,8 – при проведении выработки по сухим породам, 0,6 – по влажным и 0,3 – по обводненным или с применением водяных завес;

$K_{ут.тр}$  – коэффициент утечек воздуха в вентиляционных трубопроводах.

Расчет произвести для следующих исходных данных:  $I_{з.п} = 1 \text{ м}^3/\text{мин}$ ,  $C_0 = 0,05\%$ ,  $n_{п} = 6$  чел.,  $V_{нmin} = 0,5 \text{ м/с}$ ,  $T = 20$  мин,  $K_{ут.тр} = 1,41$ .

Организовать ручной ввод параметров  $l_{п}$ ,  $S_{св}$ ,  $B_{уг}$ ,  $B_{пор}$ , степени обводненности проходимой выработки (сухая, влажная, обводненная). На печать вывести значения расхода воздуха по всем факторам, принятое значение  $Q_{з.п}$  и сообщение об определяющем факторе.

7. Рассчитать толщину  $B_{п}$ , м, и количество ступеней  $N$  тампонажной подушки, сооружаемой при цементации пород из забоя ствола, если

$$B_{п} = \frac{\lambda \cdot P \cdot (r_{пр}^2 + \eta^2)^2}{4 \cdot r_{пр}^2 \cdot \eta \cdot m \cdot R_6},$$

где  $\lambda$  – коэффициент перегрузки,  $\lambda = 1,1 \div 1,2$ ;

$P$  – давление нагнетания цементного раствора, МПа,  $P = P_{г} + P_{и}$ ;

$P_{г}$  – гидростатический напор подземных вод в интервале цементации, МПа;

$P_{и}$  – избыточное давление цементного раствора, МПа;

$r_{пр}$  – радиус ствола в проходке, м;

$\eta$  – высота сферической поверхности тампонажной подушки, м;

$m$  – коэффициент условий работы,  $m = 0,7 \div 0,8$ ;

$R_6$  – расчетное сопротивление бетона в раннем возрасте на сжатие, МПа,  
 $R_6 = n_6 \cdot R_{28}$ ;

$n_6$  – коэффициент относительной прочности бетона в раннем возрасте;

$R_{28}$  – расчетное сопротивление бетона на сжатие в 28-суточном возрасте, МПа.

Расчет произвести для следующих исходных данных:  $P_{г} = 3$  МПа;  $P_{и} = 5$  МПа;  $\eta = 1,5$  м;  $n_6 = 0,82$ ;  $R_{28} = 20$  МПа. Значение параметра  $r_{пр}$  задать самостоятельно.

Если расчетное значение толщины подушки  $B_{п} > 2,5$  м, необходимо предусмотреть многоступенчатую тампонажную подушку, число ступеней

которой определить из выражения

$$N = \frac{B_{\text{п}}}{B_{\text{с}}},$$

где  $B_{\text{с}}$  – толщина одной ступени, принимаемая равной 1,5÷2,5 м. Если значение  $N$  получается дробным, его необходимо округлить до ближайшего большего целого.

Для рассчитанной толщины и конструкции подушки определить напряжение сжатия бетона  $\sigma_{\text{с}}$ , МПа, по формуле

$$\sigma_{\text{с}} = \frac{0,01 \cdot \lambda \cdot P \cdot (r^2 + \eta^2)^2}{4 \cdot r^2 \cdot \eta \cdot m \cdot N \cdot B_{\text{с}}},$$

где  $r$  – радиус ствола в свету, м.

Проверить конструкцию по условию прочности  $\sigma_{\text{с}} \leq R_{\text{б}}$ .

На печать вывести значения толщины тампонажной подушки, количество ступеней, напряжение сжатия бетона и сообщение о выполнении (невыполнении) условия прочности. В случае невыполнения условия прочности предусмотреть возможность изменения параметров  $R_{28}$  или  $n_{\text{б}}$ .

8. Рассчитать приближенную длину цементационной заходки  $l$ , м, при нагнетании тампонажного раствора с земной поверхности, используя уравнение

$$g \cdot (\gamma_{\text{р}} - \gamma_{\text{в}}) \cdot l^2 + 2 \cdot [h_{\text{н}} + g \cdot \gamma_{\text{р}} \cdot h_{\text{ст}} + g \cdot (\gamma_{\text{р}} - \gamma_{\text{в}}) \cdot h_{\text{п}}] \cdot l - 2 \cdot K \cdot \frac{Q_{\text{ск}}}{q} = 0,$$

где  $\gamma_{\text{р}}$ ,  $\gamma_{\text{в}}$  – плотность соответственно раствора и воды, кг/м<sup>3</sup>;

$h_{\text{ст}}$  – расстояние от статического уровня подземных вод до манометра на скважине, м;

$h_{\text{п}}$  – расстояние от кровли водоносного горизонта до статического уровня подземных вод, м;

$Q_{\text{ск}}$  – поглощающая способность скважины, м<sup>3</sup>/с;

$K$  – коэффициент учета увеличения сопротивления скважины и трещин при переходе от течения в них воды к течению раствора;

$q$  – удельное водопоглощение горных пород, м<sup>3</sup>/(с·м·Па);

$h_{\text{н}}$  – начальное давления нагнетания, Па, определяемое по следующей таблице в зависимости от величины раскрытия трещин  $\delta$ :

$\delta$ , мм	До 1	1 ÷ 5	5 ÷ 20	Более 20
$h_{\text{н}}$ , Па	$7 \cdot 10^5$	$5 \cdot 10^5$	$3 \cdot 10^5$	$2 \cdot 10^5$

Расчет произвести для следующих исходных данных:  $\gamma_{\text{р}} = 1150$  кг/м<sup>3</sup>,  $h_{\text{ст}} = 20$  м,  $h_{\text{п}} = 10$  м,  $Q_{\text{ск}} = 0,15$  м<sup>3</sup>/с,  $K = 1,2$ ,  $q = 1,2 \cdot 10^{-7}$  м<sup>3</sup>/(с·м·Па). Предусмотреть ручной ввод параметра  $\delta$ . На печать вывести величину начального давления нагнетания и значение длины цементационной заходки.



9. Произвести расчет пневматической сети участка шахты, используя выражение

$$V = k_{\text{ут}} \cdot \psi \cdot \sum_{i=1}^n m_i \cdot q_i \cdot \xi_i,$$

где  $k_{\text{ут}}$  – коэффициент, учитывающий потери воздуха от утечек через неплотности трубопровода,  $k_{\text{ут}} = 1,2$ ;

$\psi$  – коэффициент, учитывающий износ оборудования,  $\psi = 1,1$ ;

$n$  – число однотипных групп воздухоприемников;

$m_i$  – число однотипных ( $i$ -х) потребителей;

$q_i$  – расход воздуха одним ( $i$ -м) потребителем, м<sup>3</sup>/мин;

$\xi_i$  – коэффициент неравномерности работы потребителей, принимаемый равным согласно следующей таблице:

$m_i$	1	2÷4	5÷10	11÷30
$\xi_i$	1	0,95	0,90	0,80

Расчет произвести для следующих исходных данных:

$i$	Наименование воздухоприемников	$m_i$	$q_i$ , м <sup>3</sup> /мин
1	Перфоратор ПП-30	12	3,0
2	Оборудование для очистки шпуров	5	1,0
3	Пневмолоток	5	1,2
4	Отбойный молоток МО-10	4	1,15
5	Стволовая погрузочная машина КС-2у/40	2	50,0
6	Насос забойный Н-1м	2	6,0
7	Пневмотельфер	1	8,0

Результаты выполнения программы вывести на печать в форме таблицы, содержащей столбцы исходных данных и столбец значений расхода воздуха по каждому виду оборудования. Ниже последнего столбца поместить значение общего расхода воздуха на участке шахты.

10. По заданным значениям относительной влажности  $P$ , %, и скорости движения воздуха  $V$ , м/с, определить допустимую температуру воздуха  $T$ , °С, в забое, регламентируемую Правилами безопасности:

Скорость воздуха, м/с	Допустимая температура, °С, при относительной влажности, %,		
	до 75	76 – 90	свыше 90
До 0,25	24	23	22
0,26 – 0,5	25	24	23
0,51 – 1,0	26	25	24
Более 1,0	26	26	25

Предусмотреть ручной ввод значений скорости движения и влажности воздуха. На печать вывести значения всех параметров.

11. Проверить соответствие Правилам Безопасности содержания газов в шахтной атмосфере, руководствуясь следующими требованиями:

Газ	Допустимая концентрация газа, % (объемная доля)
Кислород	$\geq 20$
Углекислый газ:	
– в исходящих струях выемочных участков и тупиковых выработок;	$\leq 0,5$
– в выработках с исходящей струей крыла, горизонта, шахты в целом;	$\leq 0,75$
– при проведении и восстановлении выработок по завалу	$\leq 1$
Водород	$\leq 0,5$
Оксид углерода (IV)	$\leq 0,0017$
Оксиды азота	$\leq 0,00026$
Сернистый ангидрид	$\leq 0,00038$
Сероводород	$\leq 0,00071$

Предусмотреть возможность выбора газа и при необходимости места замера концентрации. Организовать ручной ввод значений измеренных концентраций. На печать вывести наименование выбранного газа (газов), значение измеренной концентрации и сообщение о превышении (не превышении) предельно допустимой концентрации.

12. Определить категорию шахты по метану, присваиваемую в зависимости от относительной газообильности  $q_i$ , м<sup>3</sup>/т, и вида выделения метана согласно следующей таблице:

Категория шахты по метану	Относительная метанообильность, м <sup>3</sup> /т
I	До 5
II	От 5 до 10
III	От 10 до 15
Сверхкатегорные	15 и более; шахты, опасные по суфлярным выделениям

Относительную метанообильность определить по формуле

$$q_i = \frac{1440 \cdot \sum_{i=1}^n I_i \cdot N_i}{\sum_{i=1}^n A_i},$$

где  $n$  – число месяцев работы объекта в году;

$I_i$  – расход газа на шахте в  $i$ -м месяце, м<sup>3</sup>/мин;

$N_i$  – число фактически отработанных дней в месяц по добыче угля;

$A_i$  – добыча на объекте за каждый месяц в истекшем году, т.

Исходные данные для расчета принять согласно следующей таблице:

<i>i</i>	1	2	3	4	5	6	7	8	9	10	11	12
$I_{i_s}$ М <sup>3</sup> /МИН	6,5	6,8	7,6	8,0	7,9	7,1	7,0	7,0	6,6	6,9	6,3	6,0
$N_i$	26	26	29	29	27	29	30	29	29	25	27	30
$A_{i_s}$ ТЫС. Т	57,5	54,2	61,0	60,5	52,6	55,0	61,5	60,0	52,0	41,5	49,5	66,0

На печать вывести относительную метанообильность шахты и сообщение о присвоении соответствующей категории по метану.

13. Рассчитать толщину и произвести выбор (монолитный бетон или набрызгбетон) крепи вертикальной выработки  $\delta_k$ , мм, при использовании бетонов различных классов по формуле

$$\delta_k = m_y \cdot r_{св} \cdot \left( \sqrt{\frac{0,85 \cdot R_{пр}}{0,85 \cdot R_{пр} - 2 \cdot K_k \cdot P}} - 1 \right) - \delta_{пб},$$

где  $r_{св}$  – радиус вертикальной выработки в свету, мм;

$m_y$  – коэффициент условий работы крепи,  $m_y = 1,25$ ;

$K_k$  – коэффициент концентрации напряжений в конструкции крепи,

$K_k = 1$  на протяженных участках ствола (при  $z > 20$  м) и

$K_k = 2 - 0,05 \cdot z$  в районе сопряжений;

$z$  – расстояние от узла сопряжения до рассматриваемого сечения, м;

$P$  – горизонтальное давление на крепь ствола, МПа;

$\delta_{пб}$  – толщина породобетонной оболочки, образующейся за счет проникновения бетона в окружающие нарушенные породы, принимаемая равной 500 мм для набрызгбетона и нулю для монолитной бетонной крепи;

$R_{пр}$  – расчетное сопротивление бетона сжатию, МПа, принимаемое по следующей таблице в зависимости от класса бетона:

Класс бетона	B10	B15	B20	B25	B30	B40	B50
$R_{пр}$ , МПа	4,6	7,0	9,0	11,0	13,5	17,5	21,5

Расчет произвести для всех классов бетона, приведенных в таблице. Организовать ручной ввод параметров  $r_{св}$ ,  $P$ ,  $z$ , а также предполагаемого типа крепи (монолитная бетонная или набрызгбетонная). При составлении программы следует учесть, что толщина набрызгбетонной крепи не должна превышать 150 мм. Результаты выполнения программы вывести на печать в форме таблицы, содержащей классы бетона и соответствующие им значения толщины и типа крепи.

14. Рассчитать лобовую горизонтальную силу  $P^л$ , Н, действующую на проводник в динамической системе «подъемный сосуд – армировка», используя формулу

$$P^{\text{л}} = \frac{2 \cdot \pi^2 \cdot \delta \cdot K_p^2 \cdot m \cdot V^2}{h^2} \cdot n^{\text{л}},$$

где  $K_p$  – коэффициент влияния типа рабочих направляющих подъемного сосуда, равный 1 – при жестких направляющих скольжения (принять для рельсовых и деревянных проводников) и 0,85 – при упругих роликовых направляющих (принять для коробчатых проводников);

$m$ ,  $V$  – соответственно масса, кг, и скорость, м/с, груженого подъемного сосуда;

$n^{\text{л}}$  – коэффициент, учитывающий жесткость проводников и эксцентриситет центра масс груженого сосуда, принять равным для рельсовых проводников 1,25, для коробчатых 1,15, для деревянных 1,35;

$h$  – шаг армировки, м;

$\delta$  – зазор на сторону между рабочими или предохранительными направляющими скольжения и проводником, м, принимаемый по таблице в зависимости от типа проводников:

Тип проводников	Рельсовые	Коробчатые	Деревянные
$\delta$ , м	0,010	0,015	0,020

Расчет произвести для следующих значений шага армировки  $h$ , м:

- для рельсовых проводников – 3,126; 4,168; 6,252;
- для коробчатых проводников – 3; 4; 5; 6;
- для деревянных проводников – 2; 3; 4.

Предусмотреть ручной ввод массы и скорости движения груженого подъемного сосуда. Результаты выполнения программы вывести на экран в виде таблицы, содержащей тип проводника, шаг армировки и соответствующее им значение горизонтальной силы на проводник.

## 7. МАТЕРИАЛЬНОЕ И КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ

---

### 7.1. Понятия модели и моделирования. Сущность моделирования

Модели и моделирование используются человечеством давно. С помощью моделей и модельных отношений развились разговорные языки, письменность, графика. Наскальные изображения наших предков, затем картины и книги – это модельные, информационные формы передачи знаний об окружающем мире последующим поколениям. Модели применяются при изучении сложных явлений, процессов, конструировании новых сооружений. Хорошо построенная модель, как правило, доступнее для исследования, нежели реальный объект. Более того, некоторые объекты вообще не могут быть изучены непосредственным образом: недопустимы, например, эксперименты с экономикой страны в познавательных целях; принципиально неосуществимы эксперименты с прошлым или, скажем, с планетами Солнечной системы и т. п.

Модель позволяет научиться правильно работать с объектом, апробируя различные варианты управления на его модели. Экспериментировать в этих целях с реальным объектом в лучшем случае бывает неудобно, а зачастую просто вредно или вообще невозможно в силу ряда причин (большой продолжительности эксперимента во времени, риска привести объект в нежелательное и необратимое состояние и т. п.)

**Модель** – это материальный или мысленно представляемый объект, замещающий в процессе изучения объект-оригинал, и сохраняющий значимые для данного исследования типичные его черты. Основным достоинством моделей является возможность экспериментировать путем легкого вмешательства с целью изменения (варьирования) сравнительно ограниченного числа входных переменных и быстрого получения выходных результатов. Процесс построения модели называется моделированием.

Другими словами, **моделирование** – это процесс изучения строения и свойств оригинала с помощью модели.

Смысл *моделирования* заключается в том, чтобы по результатам опытов на модели можно было судить о явлениях, происходящих в натурном объекте.

**В основе моделирования лежат информационные процессы, поскольку само создание модели базируется на информации о реальном объекте или процессе.**

Модель является представлением действительности. Но модель не может быть полностью тождественной натурному объекту, ибо в этом случае она не выполняла бы своей основной функции и представляла бы собой лишь копию сложного реального объекта. Поэтому при моделировании неизбежна

определенная степень упрощения, хотя и стремятся, чтобы модель достаточно хорошо отображала исследуемую сторону функционирования натурального объекта.

## 7.2. Классификация моделей

Различают **материальное** и **идеальное** моделирование. Материальное моделирование, в свою очередь, делится на физическое и аналоговое моделирование.

*Физическим* принято называть моделирование, при котором реальному объекту противопоставляется его увеличенная или уменьшенная копия, допускающая исследование (как правило, в лабораторных условиях) с помощью последующего перенесения свойств изучаемых процессов и явлений с модели на объект на основе теории подобия. Примерами моделей такого рода служат: в астрономии – планетарий, в архитектуре – макеты зданий, в самолетостроении – модели летательных аппаратов и т. п.

*Аналоговое моделирование* основано на аналогии процессов и явлений, имеющих различную физическую природу, но одинаково описываемых формально (одними и теми же математическими уравнениями).

От материального моделирования принципиально отличается **идеальное моделирование**, которое основано не на материальной аналогии объекта и модели, а на аналогии идеальной, мыслимой. Основным типом идеального моделирования является знаковое моделирование.

*Знаковым* называется моделирование, использующее в качестве моделей знаковые преобразования какого-либо вида: схемы, графики, чертежи, формулы, наборы символов.

Важнейшим видом знакового моделирования является *математическое моделирование*, при котором исследование объекта осуществляется посредством модели, сформулированной на языке математики. Классическим примером математического моделирования является описание и исследование законов механики Ньютона средствами математики.

По степени полноты изучаемого объекта модели подразделяются на полные, неполные и приближенные. В основе *полных* моделей лежит полное подобие, которое проявляется как во времени, так и в пространстве. Полные модели существуют только в производственно-экспериментальном методе.

Для *неполных* моделей характерно моделирование и соблюдение требований подобия для ограниченного числа характеристик объекта. В основе *приближенного моделирования* лежит подобие, при котором некоторые стороны функционирования натурального объекта не моделируются совсем. В этом случае используют те характеристики процесса, которые известны, или исключают из рассмотрения характеристики, влияние которых незначительно. Степень приближения в каждом конкретном случае

моделирования устанавливают при сопоставительном анализе и сравнении получаемых результатов исследований модели с данными измерений в натуре для отобранных контрольных точек.

### 7.3. Принципы и схема процесса моделирования

Для построения моделей используют два принципа: **дедуктивный** (от общего к частному) и **индуктивный** (от частного к общему). При первом подходе рассматривается частный случай общеизвестной фундаментальной модели, которая приспособливается к условиям моделируемого объекта с учетом конкретных обстоятельств. Второй способ предполагает выдвижение гипотез, декомпозицию сложного объекта, анализ, а затем синтез. Здесь широко используется подобие, поиск аналогий, умозаключение с целью формирования каких-либо закономерностей в виде предположений о поведении системы. Ниже представлена схема процесса моделирования:



Технология моделирования требует от исследователя умения корректно формулировать проблемы и задачи, прогнозировать результаты, проводить разумные оценки, выделять главные и второстепенные факторы для построения моделей, находить аналогии и выражать их на языке математики.

### 7.4. Методы материального моделирования в горном деле

В материальных моделях, используемых в горном деле, существенные свойства натурного объекта представлены самими этими свойствами, но, как правило, в ином масштабе, поэтому их называют *моделями геометрического подобия*. Наглядные модели внешне похожи на реальный объект, но отличаются от него размерами, представляя собой образы или копии этого объекта.

Реальные вещи можно изобразить наглядно в виде трехмерной модели: глобус, модель горной машины или ее узла, макет подземного сооружения и т.п. Эти же объекты можно изобразить в виде двухмерных моделей: фотография, эскиз, план, чертеж. Наглядные модели служат для того, чтобы создать четкий зрительный образ объекта или процесса.

В наглядных физических моделях, называемых *моделями физического подобия*, воспроизводят физические процессы, протекающие в натурном

объекте. С помощью методов теории подобия размерные физические величины объединяют в безразмерные комбинации. Благодаря введению безразмерных комбинаций число аргументов сокращается, что упрощает исследование физического процесса.

Понятие подобия распространяется на любые физические процессы. Обязательной предпосылкой подобия физических явлений должно быть *геометрическое подобие*. Большинство физических процессов, подлежащих изучению в горном деле, описывается условиями подобия, которые могут быть разделены на три группы: механические (силовые), гидромеханические и тепловые.

В основе *механического* подобия лежит общий закон подобия Ньютона. Понятие подобия физических явлений применимо только к явлениям одного и того же рода, которые качественно одинаковы и аналитически описываются одинаковыми уравнениями как по форме, так и по содержанию.

Из второго закона механики Ньютона для двух явлений (в натуре и на модели) имеет место:

$$P_1 = m_1 a_1; P_2 = m_2 a_2,$$

где  $P_1, P_2$  – силы;  $m_1, m_2$  – массы;  $a_1, a_2$  – ускорения соответственно в натуре и на модели.

Если массу выразить через объем и плотность, а ускорение – через линейные размеры и время, то условием механического подобия будет выражение

$$\frac{\sigma_n}{\rho_n l_n^2 v_n^2} = \frac{\sigma_m}{\rho_m l_m^2 v_m^2} = Ne = idem, \quad (7.1)$$

где  $\sigma_n, \sigma_m$  – напряжения соответственно в натуре и на модели, Па;  
 $\rho_n, \rho_m$  – плотности соответственно в натуре и на модели, кг/м<sup>3</sup>;  
 $l_n, l_m$  – линейные размеры соответственно в натуре и на модели, м;  
 $v_n, v_m$  – скорости соответственно в натуре и модели, м/с;  
 $Ne$  – некоторое безразмерное число, называемое общим критерием подобия Ньютона.

Определив параметры, при которых удовлетворяется условие подобия (7.1), получают модель, механические процессы в которой будут подобны соответствующим процессам в натуре.

Моделирование на физических моделях механических явлений, протекающих в массиве горных пород, разработано и осуществлено в 1936-37 гг. Г.Н. Кузнецовым. Такое моделирование получило название «*метода эквивалентных материалов*» и применяется для исследования проявлений горного давления в подземных капитальных горных выработках, при изучении пучения пород, сдвижения массивов и других физических процессов, происходящих в массиве в связи с проведением в них горных работ.

Модели метода эквивалентных материалов нашли широкое применение в исследованиях ВНИМИ и ЛИИЖТа при проектировании станци-



онных и перегонных туннелей метрополитена, безлюдной добычи угля в очистном забое. Сущность метода заключается в следующем. Модель породного массива создается из искусственных материалов, прочность и модуль деформации которых уменьшены в определенном соотношении с натурными величинами. Выражая в формуле (7.1) значения квадратов скоростей через ускорения и длины, получим

$$\frac{\sigma_{\text{н}}}{\rho_{\text{н}} a_{\text{н}} l_{\text{н}}^3} = \frac{\sigma_{\text{м}}}{\rho_{\text{м}} a_{\text{м}} l_{\text{м}}^3} = Ne. \quad (7.2)$$

Поскольку деформации и разрушения горных пород происходят за счет действия сил тяжести, величина  $a$  в формуле (7.2) имеет значение ускорения свободного падения  $g$ .

$$\text{Тогда имеем} \quad \rho_{\text{м}} a_{\text{м}} = \rho_{\text{м}} g = \gamma_{\text{м}}; \quad \rho_{\text{н}} a_{\text{н}} = \rho_{\text{н}} g = \gamma_{\text{н}}, \quad (7.3)$$

где  $\gamma_{\text{м}}$ ,  $\gamma_{\text{н}}$  – удельные веса материала природы и модели.

Введем в формулу (7.2) обозначения отношений сил  $P$  к площадям  $l^2$  через  $N$ , под которыми подразумеваются величины, имеющие размерность напряжений: пределы прочности материала при растяжении, сжатии, сдвиге, а также модуль деформации. После подстановки с учетом соотношений (7.3) получим

$$N_{\text{м}} / (\gamma_{\text{м}} l_{\text{м}}) = N_{\text{н}} / (\gamma_{\text{н}} l_{\text{н}}) = \text{idem}. \quad (7.4)$$

Из этого выражения получаем

$$N_{\text{м}} = \frac{l_{\text{м}} \gamma_{\text{м}}}{l_{\text{н}} \gamma_{\text{н}}} N_{\text{н}}. \quad (7.5)$$

Зная механические характеристики материала природы  $N_{\text{н}}$  и задавшись соотношением  $l_{\text{м}}/l_{\text{н}}$  и  $\gamma_{\text{н}}/\gamma_{\text{м}}$ , получим значения механических свойств материала модели  $N_{\text{м}}$ , которые необходимы для удовлетворения требований подобия модели и природы. Такие искусственные материалы, которые удовлетворяют соотношению (7.5), называются *эквивалентными*. Поскольку линейные размеры в модели значительно (в 10 – 100 раз) меньше, чем в натуре, а удельные веса материалов одного порядка, то для удовлетворения соотношения (7.4) эквивалентные материалы должны иметь прочностные характеристики также значительно меньшие, чем натурные породы. В качестве эквивалентных материалов чаще всего применяют песчано-гипсовые бетоны, песок, замешанный на вазелине, и др.

Для размещения эквивалентного материала применяют плоские и объемные стенды. На рис. 7.2 приведена плоская модель из эквивалентных материалов, на которой можно изучать область неупругих деформаций, концентрацию напряжений и перемещения пород.



A-A

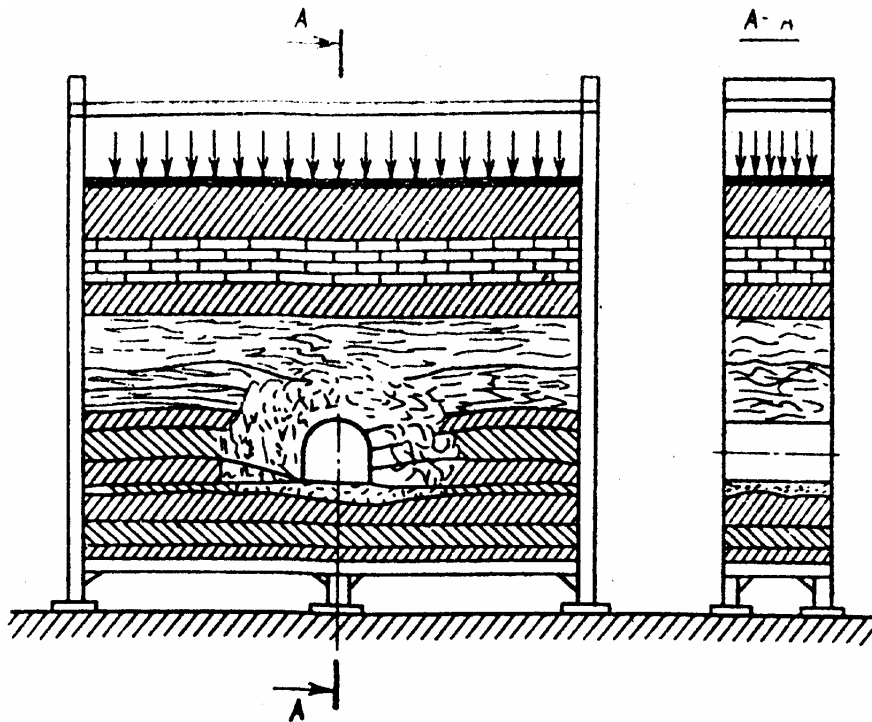


Рис. 7.1. Плоская модель породного массива из эквивалентных материалов

Для исследования напряжений в горных породах вокруг горизонтальной выработки, осадки сооружений, устойчивости откосов и решения других задач применяют метод *центробежного моделирования*. Его также используют для исследования процессов взрывного разрушения горных пород и сейсмических колебаний.

Сущность метода заключается в том, что масштабную модель выделенной области породного массива помещают на центрифугу, с помощью которой создается механическое подобие сил, действующих в натуре. Иными словами, благодаря действию центробежных сил вес модели увеличивается, и при определенной частоте вращения достигается механическое подобие в соответствии с принятым масштабом  $\mu_l$  линейных размеров в натуре и на модели.

Для соблюдения условий подобия необходимо, чтобы в соответствии с соотношением (7.4) при  $N_m = N_n$  удельный вес материала модели  $\gamma_m$  был во столько раз больше удельного веса породы в натуре  $\gamma_n$ , во сколько раз линейные размеры в натуре больше размеров в модели. Это достигается за счет инерционных сил, действующих на модель при ее вращении с ускорением  $a$ , превышающим естественное ускорение свободного падения  $g$  в  $\mu_l$  раз.

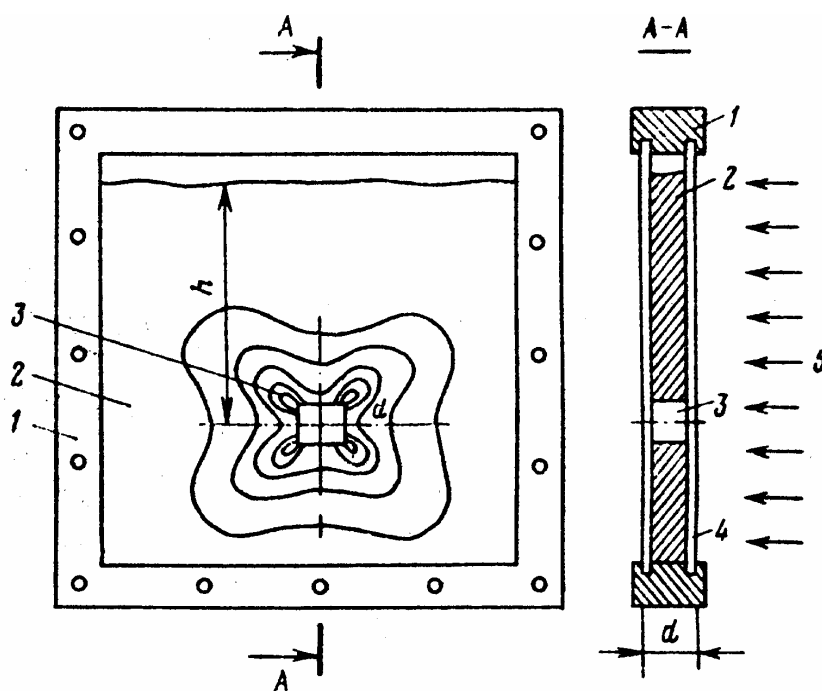
Наглядным методом моделирования механических процессов в породных массивах, окружающих горные выработки, является *оптический (оптико-поляризационный)* или *метод фотомеханики*. Его применяют для определения условий устойчивости породных массивов и элементов обделки подземного сооружения, установления закономерностей взаимодей-

ствия породных массивов и подземных сооружений, а также для изучения степени влияния подземных сооружений на окружающие породные массивы.

Возникающие под действием горных работ и сил тяжести механические явления моделируются так же, как при методе эквивалентных материалов специально подобранными в соответствии с критериями механического подобия материалами. В отличие от эквивалентных, эти материалы прозрачны для света и обладают оптической чувствительностью к деформациям и механическим напряжениям. К таким материалам относятся стекло, целлулоид, бакелит, желатин, эпоксидная смола и др. Оптическая чувствительность определяется разностью хода  $\Gamma$  двух плоскополяризованных лучей, которая пропорциональна действующим напряжениям и легко регистрируется визуально или с помощью фотоаппарата.

Модель породного массива представляет собой плоскую пластинку из оптически чувствительного материала толщиной  $d$ , два других размера которой в соответствии с требованиями геометрического подобия определяют глубину и ширину исследуемого участка в натуре. Отверстия в пластинке моделируют горную выработку (рис. 7.2).

Для соблюдения требований механического подобия при больших значениях линейного масштаба используют специально подобранные оптически чувствительные материалы на желатин-глицериновой основе с весьма низким модулем упругости. Такой материал под действием собственного веса растекается, поэтому для его удержания модель породного массива помещают в рамку с прозрачными стенками.



**Рис. 7.2. Оптико-поляризационная модель:**

1 – рамка; 2 – оптически чувствительный материал; 3 – модель горной выработки; 4 – стекло; 5 – поляризованный луч ( $h$  – глубина расположения выработки)

Поляризованный луч создают на специальных оптических установках. Величина разности хода лучей на модели определяется по методам сопоставления цветов, полос, компенсации. При первом методе регистрируют цвета участков модели и по специальным таблицам определяют разность хода лучей  $\Gamma$ . Методом полос определяется порядковый номер полосы света в исследуемой области модели и учитывается степень влияния каждой полосы. По методу компенсации к искомой разности хода, возникающей в исследуемой области модели, добавляется компенсирующая разность хода лучей, вследствие чего в измеряемой точке модели наблюдается затемнение.

Главные касательные напряжения в данной точке модели, для которой известно значение  $\Gamma$ , определяются по формуле

$$\tau_{\max} = \frac{\sigma_1 - \sigma_2}{2} = \frac{\Gamma}{2C_y d},$$

где  $\sigma_1, \sigma_2$  – наибольшее и наименьшее главные напряжения;  $C_\sigma$  – оптический коэффициент чувствительности, постоянный для данного материала модели;  $d$  – толщина модели.

Соединяя точки равных главных касательных напряжений линиями, получают так называемые изохромы. В результате исследований определяют напряжения на контуре выработки и в породном массиве, устанавливают опасные сечения, дают рекомендации по укреплению породного массива.

*Гидромеханическое моделирование* получило распространение при изучении газодинамических процессов, происходящих в горных выработках. На моделях исследуют процессы переноса газов в атмосфере горной выработки, их разбавление воздухом. Изучаются явления загрязнения рудничной атмосферы автотранспортом, взрывными работами. Значительное число работ по гидромеханическому моделированию выполнено применительно к тупиковым выработкам, в частности к камерам.

Гидромеханическое моделирование осуществляется на гидравлических или воздушных моделях. Более распространены модели первого типа, что объясняется быстрым развитием процесса во времени и трудностью его фиксации на воздушных (аэродинамических) моделях. Газообразная примесь (обычно газы взрывчатых веществ) имитируется на таких моделях раствором кислоты (соли) или краской, а определение ее концентрации выполняется электрическим способом, основанным на измерении электропроводности раствора, или колориметрическим способом, основанным на измерении оптической плотности раствора.

*Моделирование на основе теплового подобия* применяется для исследования процесса образования ледопородного ограждения вокруг замораживающих колонок при строительстве подземных сооружений. Достоинством тепловых моделей является их наглядность и простота основных критериев подобия, в качестве которых выступает линейный масштаб и

время  $t$ , масштаб которого определяется из соотношения  $\mu_t = 1/\mu_l^2$  ( $\mu_l$  – линейный масштаб). Масштабы температур хладоносителя и горной породы равны единице.

Простота критериев, однако, приводит к необходимости иметь довольно сложную экспериментальную установку. Для регистрации температур в замерной плоскости модели породного массива устанавливаются термопары. Измерение ЭДС термопар осуществляют с помощью самопишущих потенциометров. Для автоматизации регистрации и обработки данных измерений применяют мини-ЭВМ.

Тепловое моделирование газодинамических процессов основано на сходстве механизма переноса вещества и тепла в турбулентном потоке, в результате чего концентрационные и температурные поля оказываются подобными. Тепловое моделирование имеет ряд достоинств, к числу которых относится возможность использования высокоточной измерительной аппаратуры.

## 7.5. Понятия компьютерного и имитационного моделирования

В современном мире все шире применяется процесс *компьютерного моделирования*, подразумевающий использование вычислительной техники для проведения экспериментов с моделью.

*Компьютерная модель* – это модель реального процесса или явления, реализованная компьютерными средствами. Если состояние системы меняется со временем, то модели называют динамическими, в противном случае – статическими.

Процессы в системе могут протекать по-разному в зависимости от условий, в которых находится система. Следить за поведением реальной системы при различных условиях бывает трудно, а иногда и невозможно. В таких случаях, построив модель, можно многократно возвращаться к начальному состоянию и наблюдать за ее поведением. Этот метод исследования систем называется *имитационным* моделированием.

Моделирование событий реального мира может производиться многими способами. Явления макромира достаточно хорошо описываются моделями, построенными на математике бесконечного и непрерывного. События же, происходящие в микромире, плохо поддаются описанию подобным способом и требуют применения других принципов моделирования.

Еще в 1970 году известным математиком А. Н. Колмогоровым давался прогноз, что с *«развитием современной вычислительной техники будет во многих случаях разумно вести изучение реальных явлений, избегая промежуточный этап их стилизации в духе математики бесконечного и непрерывного, переходя прямо к дискретным моделям»*. Сейчас уже можно с

уверенностью сказать, что этот прогноз сбылся, так как появилось большое количество разнообразных математических систем, основанных на принципе мелкозернистого параллелизма, и, самое главное, появились программные и аппаратные комплексы, способные моделировать работу таких систем.

## **7.6. Компьютерное моделирование в горном деле**

Горное производство является одним из самых трудоемких, сложных и опасных видов промышленного производства. Одним из объектов исследований в горном деле является породный массив с капитальными, подготовительными и очистными выработками. Спецификой их изучения является большая продолжительность исследований во времени, значительные размеры объектов в пространстве, недоступность непосредственного изучения традиционными методами и средствами. Для проведения исследований в натуральных условиях требуются значительные затраты, а иногда осуществить желаемый эксперимент или наблюдение технически очень сложно, опасно или принципиально невозможно.

В таких случаях на помощь исследователям приходят методы математического или компьютерного моделирования. В настоящее время разработано большое количество математических моделей массива, полученных на основе строгих аналитических решений соответствующих задач теории упругости. Эти модели подробно описаны в работах [1,2]. Однако в ряде случаев (неоднородный массив, наличие трещин, сопряжения и пересечения выработок, требующие постановки объемной задачи и др.) аналитических решений не имеется. В этих случаях целесообразно использование приближенных методов решения задач и численных моделей подземных сооружений.

Одним из таких методов, получающим все большее распространение в моделировании и вычислительных программных комплексах, является метод конечных элементов (МКЭ). Этот метод рассматривает ограниченную область (плоскую или объемную), которая разбивается на конечное число элементов (в плоской задаче обычно принимаются треугольные элементы), при этом стыковка элементов осуществляется только в вершинах. Таким образом, условия равновесия и совместности деформаций соблюдаются только в общих узлах элементов. Для определения неизвестных усилий в узлах и смещений узлов по заданным усилиям или перемещениям на границе области составляются уравнения равновесия и совместности деформаций, число которых соответствует числу узлов расчетной схемы.

Для решения таких задач используются современные программно-вычислительные комплексы (ПВК) Cosmos/Design Star, MSC/Nastran (NAsa STRuctural ANalysis) for Windows, ANSYS, Structure Cad, «Липа-Windows», «Мираж», «Мономах» и др.

Анализ конструкций с использованием МКЭ является фактическим мировым стандартом для прочностных и других видов расчетов конструкций. Основой этого служит универсальность МКЭ, позволяющая единым способом рассчитывать различные конструкции с разными свойствами материалов и находить параметры их напряженно-деформированного состояния (НДС).

### **7.7. Назначение и структура программного комплекса «Лира-Windows»**

*Программный комплекс Лира-Windows (ПК ЛИРА)* – это многофункциональный программный комплекс для расчета, исследования и проектирования конструкций различного назначения.

ПК ЛИРА с успехом применяется в расчетах объектов строительства, горной промышленности, машиностроения, мостостроения, атомной энергетики, нефтедобывающей промышленности и во многих других сферах, где актуальны методы строительной механики.

Программные комплексы семейства ЛИРА имеют более чем 40-летнюю историю создания, развития и применения в научных исследованиях и практике проектирования конструкций. Программные комплексы семейства ЛИРА непрерывно совершенствуются и приспособляются к новым операционным системам и графическим средам. Новейшим представителем семейства ЛИРА является ПК ЛИРА версии 9.0.

Кроме общего расчета модели объекта на все возможные виды статических нагрузок, температурных, деформационных и динамических воздействий (ветер с учетом пульсации, сейсмические воздействия и т.п.) ПК ЛИРА автоматизирует ряд процессов проектирования: определение расчетных сочетаний нагрузок и усилий, назначение конструктивных элементов, подбор и проверка сечений стальных и железобетонных конструкций с формированием эскизов рабочих чертежей колонн и балок.

ПК ЛИРА позволяет исследовать общую устойчивость рассчитываемой модели, проверить прочность сечений элементов по различным теориям разрушений. ПК ЛИРА предоставляет возможность производить расчеты объектов с учетом физической и геометрической нелинейностей, моделировать процесс возведения сооружения с учетом монтажа и демонтажа элементов.

ПК ЛИРА состоит из следующих взаимосвязанных информационных систем.

*Система ЛИРА-ВИЗОР* – это единая графическая среда, которая располагает обширным набором возможностей и функций для формирования адекватных конечно-элементных и супер-элементных моделей рассчитываемых объектов, их подробного визуального обследования и корректи-

ровки, для задания физико-механических свойств материалов, связей, разнообразных нагрузок, характеристик различных динамических воздействий, а также взаимосвязей между загружениями для определения их наиболее опасных сочетаний.

Возможности, предоставляемые по результатам расчета при отображении напряженно-деформированного состояния объекта, позволяют произвести детальный анализ полученных данных по полям перемещений и напряжений, по эпюрам усилий и прогибов, по мозаикам разрушения элементов, по главным и эквивалентным напряжениям и по многим другим параметрам. ЛИР-ВИЗОР предоставляет исчерпывающую информацию по всему объекту и по его элементам.

В системе ЛИР-ВИЗОР реализована возможность визуализации схемы и ее напряженно-деформированного состояния в графике OpenGL.

ЛИР-ВИЗОР позволяет вести общение с комплексом на русском и английском языках, причем замена языка может осуществляться на любой стадии работы с комплексом. ЛИР-ВИЗОР дает возможность использовать любую действующую систему единиц измерения как при создании модели, так и при анализе результатов расчета.

*Система ЛИР-КС (Конструктор сечений)* позволяет в специализированной графической среде сформировать сечения произвольной конфигурации, вычислить их осевые, изгибные, крутильные и сдвиговые характеристики. Кроме того, предоставляется возможность вычисления секториальных характеристик сечений, координат центров изгиба и кручения, моментов сопротивления, а также определения формы ядра сечения. При наличии усилий в заданном сечении производится отображение картины распределения текущих, главных и эквивалентных напряжений, соответствующих различным теориям прочности.

*РАСЧЕТНЫЙ ПРОЦЕССОР* реализует современные усовершенствованные методы решения систем уравнений, обладающие высоким быстродействием и позволяющие решать системы с очень большим числом неизвестных. В расчетном процессоре содержится обширная *БИБЛИОТЕКА КОНЕЧНЫХ ЭЛЕМЕНТОВ*, которая позволяет создавать адекватные расчетные модели практически без ограничений на описание реальных свойств рассчитываемых объектов. При этом возможны задание линейных и нелинейных законов деформирования материалов, учет геометрической нелинейности с нахождением формы изначально изменяемых систем, а также учет конструктивной нелинейности. Реализованы законы деформирования различных классов железобетона. При расчетах нелинейных задач производится автоматический выбор шага нагружения с учетом его истории. Возможности процессора позволяют смоделировать поведение сооружения в процессе возведения при многократном изменении расчетной схемы. Полный состав библиотеки конечных элементов.



*Система УСТОЙЧИВОСТЬ* дает возможность произвести проверку общей устойчивости рассчитываемого сооружения с определением коэффициента запаса и формы потери устойчивости.

*Система ЛИТЕРА* реализует вычисление главных и эквивалентных напряжений по различным теориям прочности.

*Система ФРАГМЕНТ* позволяет определить силы воздействия одного фрагмента рассчитываемого сооружения на другой как нагрузку. В частности, могут быть определены нагрузки, передаваемые наземной частью расчетной схемы на фундаменты.

*Конструирующая система ЛИР-АРМ* реализует подбор площадей сечения арматуры колонн, балок, плит и оболочек по первому и второму предельным состояниям в соответствии с действующими в мире нормативами. Существует возможность задания произвольных характеристик бетона и арматуры, что имеет большое значение при расчетах, связанных с реконструкцией сооружений. Система позволяет объединять несколько однотипных элементов в конструктивный элемент, что позволяет производить увязку арматуры по длине всего конструктивного элемента. Система может функционировать в локальном режиме, осуществляя как подбор арматуры, так и проверку заданного армирования для одного элемента. По результатам расчета формируются чертежи балок и колонн, а так же производится создание dxf-файлов чертежей.

*Конструирующая система ЛИР-СТК* работает в двух режимах – подбора сечений элементов стальных конструкций, таких, как фермы, колонны и балки, и проверки заданных сечений в соответствии с действующими в мире нормативами. Допускается объединение нескольких однотипных элементов в конструктивный элемент. Система может функционировать в локальном режиме, позволяя проверить несколько вариантов при конструировании требуемого элемента.

*Система ЛИР-РС*, которая информационно связана с системой ЛИР-СТК, позволяет производить редактирование используемой сортаментной базы прокатных и сварных профилей.

*Система ДОКУМЕНТАТОР* предназначена для формирования отчетов по результатам работы с комплексом. При этом вся информация может быть представлена как в табличном, так и в графическом виде. Табличный и графический разделы необходимой для отчета информации могут быть размещены совместно на специально организуемых для этой цели листах и снабжены комментариями и надписями. Кроме того, табличная информация может быть передана в Microsoft Excel, а графическая – в Microsoft Word. Реализован вывод таблиц в формате HTML.

*Протокол расчета задачи.* В этом файле содержится следующая информация:

- ориентировочный прогноз времени счета;
- ошибки и предупреждения, обнаруженные при контроле исходных данных;

- требуемые объемы виртуальной и дисковой памяти;
- информация об объеме задачи, количестве неизвестных и элементов, о ширине ленты системы уравнений до и после оптимизации профиля матрицы;
- о наличии геометрической изменяемости;
- о контроле решения системы уравнений;
- о формировании файлов результатов.

*Шаговый нелинейный процессор* предназначен для решения физически и геометрически нелинейных, а также контактных задач.

В линейных задачах существует прямая пропорциональность между нагрузками и перемещениями вследствие малости перемещений, а также между напряжениями (усилиями) и деформациями вследствие линейного закона Гука. Поэтому для линейных задач справедлив принцип суперпозиции и независимости действия сил.

В физически нелинейных задачах отсутствует прямая пропорциональность между напряжениями и деформациями. Материал конструкции подчиняется нелинейному закону деформирования. Закон деформирования может быть и несимметричным – с различными пределами сопротивления растяжению и сжатию.

В геометрически нелинейных задачах отсутствует прямая пропорциональность между деформациями и перемещениями. На практике наибольшее распространение имеет случай больших перемещений при малых деформациях.

В задачах конструктивной нелинейности имеет место изменение расчетной схемы по мере деформирования конструкции – например, в момент достижения некоторой точкой конструкции определенной величины перемещения возникает контакт этой точки с опорой.

Для решения таких задач шаговый нелинейный процессор организует процесс пошагового нагружения конструкции и обеспечивает решение линеаризованной системы уравнений на каждом шаге для текущего приращения вектора узловых нагрузок, сформированного для конкретного нагружения.

Шаговый нелинейный процессор позволяет получить напряженно-деформированное состояние для мономатериальных и для биматериальных, в частности железобетонных, конструкций. Моделирование физической нелинейности производится с помощью конечных элементов, оперирующих библиотекой законов деформирования материалов.

Моделирование геометрической нелинейности производится с помощью конечных элементов, учитывающих изменение геометрии конструкции и возникновение мембранной группы напряжений (усилий), что позволяет рассчитывать мембранные и вантовые конструкции.

Моделирование конструктивной нелинейности обеспечивается наличием специальных конечных элементов односторонних связей.

Матрица жесткости линейризованной физически нелинейной системы формируется на основании переменных интегральных жесткостей, получаемых в точках интегрирования как по сечению, так и по конечному элементу при решении линейной задачи на каждом шаге. Сечение конечного элемента в точках интегрирования дробится на ряд элементарных подбластей, в центрах которых определяются новые значения жесткостных характеристик в соответствии с заданной диаграммой деформирования. На каждом шаге решается линейризованная задача с формированием векторов перемещений, усилий (напряжений) и новых жесткостей по касательному модулю деформации для следующего шага.

При расчете геометрически нелинейных систем считается, что закон Гука соблюдается. На каждом шаге происходит учет мембранной группы усилий (для стержней – учет продольной силы) при построении матрицы жесткости. Для решения нелинейных задач необходимо задавать информацию о количестве шагов и коэффициентах к нагрузке. Схема может содержать несколько нагружений, из которых может быть сформирована последовательность (история) нагружений. Для решения геометрически нелинейных задач реализован автоматический выбор шага нагружения.

*Процессор МОНТАЖ* предназначен для прочностного расчета конструкций с учетом процесса возведения и предусматривает два вида монтажных операций – монтаж и демонтаж элементов. Изменение условий закрепления конструкции или сопряжения элементов между собой моделируется с помощью специальных КЭ, например связи конечной жесткости, упругой связи между узлами, односторонней связи и т.п.).

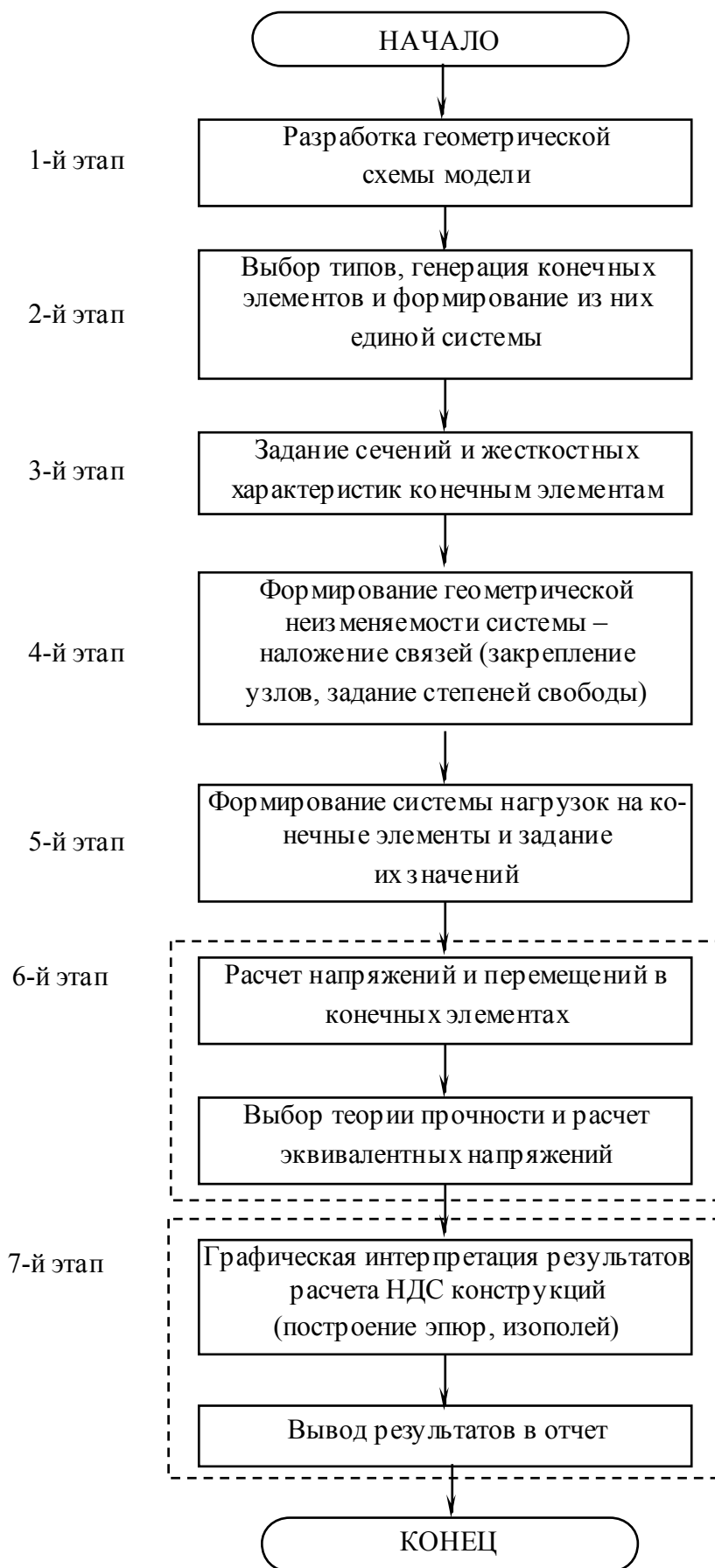
В результате работы процессора МОНТАЖ вычисляются усилия и напряжения в элементах, накапливаемые в процессе возведения, а также расчетные сочетания усилий (PCY) в них.

По умолчанию перемещения узлов в процессе счета не накапливаются, а их значения выдаются независимо для каждой стадии. Если же перемещения необходимо в процессе расчета накопить, то в таблице моделирования нелинейных нагружений вводится признак учета предыстории нагружения.

Результаты работы процессора МОНТАЖ могут быть импортированы в конструирующие системы ЛИР-АРМ и ЛИР-СТК.

## **7.8. Алгоритм построения компьютерных моделей в ПК «Лири-Windows»**

Рассмотрим основные этапы построения конечно-элементной модели с помощью ПК «Лири-Windows». Схематически последовательность решения геомеханических задач МКЭ с помощью ПК «Лири-Windows» представлено на рис. 7.3.



**Рис. 7.3. Последовательность решения задачи методом конечных элементов в ПК «Лира-Windows»**

*Первый этап* представляет собой геометрическое моделирование и заключается в выборе системы координат (координатной плоскости), начала координат и геометрических параметров моделируемой системы (ее размеров и расположения элементов в пространстве).

*Второй этап* включает выбор типа конечных элементов (стержни, трех- или четырехугольные пластины, объемные элементы различной конфигурации и др.) из библиотеки КЭ, генерацию регулярных фрагментов (рам, плит, балок-стенок и др.), ферм, поверхностей вращения и т.д. в зависимости от типа поставленной задачи, а также синтез отдельных стандартных фрагментов в единую систему, являющуюся геометрической основой будущей конечно-элементной модели.

*Третий этап* заключается в задании каждому КЭ или их группе соответствующих выбранному типу сечений, которые выбираются из стандартных или же задаются пользователем самостоятельно, и жесткостных характеристик (модуль Юнга, коэффициент Пуассона, плотность, коэффициенты постели (Винклера), которые будут определять параметры напряжений и деформаций элементов при их работе под нагрузкой.

*Четвертый этап* включает выбор узлов, для которых устанавливаются ограничения в перемещениях по отдельным осям или вращению. Это делается для создания геометрически неизменяемой системы, что является одним из обязательных условий расчета по методу конечных элементов.

*Пятый этап* заключается в выборе типа, места и направления действия нагрузок, которые выбираются в соответствии с назначенными типами КЭ. ПВК «Лира-Windows» позволяет моделировать сосредоточенные и распределенные (равномерно или неравномерно) силы, статические, динамические, гармонические, температурные нагрузки, сосредоточенные или распределенные моменты, нагрузки от собственного веса, а также задавать начальные смещения и повороты узлов модели. Правильное задание типа и величины нагрузки, а также закрепленных узлов во многом определяет корректность постановки задачи и соответственно получаемых результатов.

После выполнения работ первых 6 этапов конечно-элементная модель считается сформированной и может быть запущена на выполнение, т.е. можно производить расчет всех параметров НДС узлов и элементов.

На *шестом* этапе производится расчет всех перемещений (поступательных и вращательных) вдоль осей координат, изгибающих и крутящих моментов, продольных и перерезывающих сил (для стержней); нормальных и тангенциальных напряжений (для пластин и объемных КЭ). Для оценки состояния конструкции (возникают ли допустимые или критические напряжения, произойдет ли разрушение) выбирается теория прочности (максимальных нормальных напряжений, максимальных деформаций, максимальных касательных напряжений, энергетическая, Мора, Ягна-Бужинского, Баландина, Миролюбова, Друккера-Прагера, Волкова, Писаренко-Лебедева) и в соответствии с выбранной теорией производится расчет главных и эквивалентных (сжимающих и растягивающих) напряжений, по которым и делается вывод о состоянии конструкции или массива при воздействии заданных нагрузок.

*Седьмой этап* включает построение графических отображений результатов расчета. Для стержневых систем в масштабе строятся эпюры изгибающих моментов, продольных и перерезывающих сил, указываются их максимальные значения, а также мозаика перемещений по узлам. Для пластин и объемных элементов строятся изополя перемещений по осям и эквивалентных напряжений. По желанию пользователя можно построить эпюру напряжений или перемещений по любому выбранному сечению.

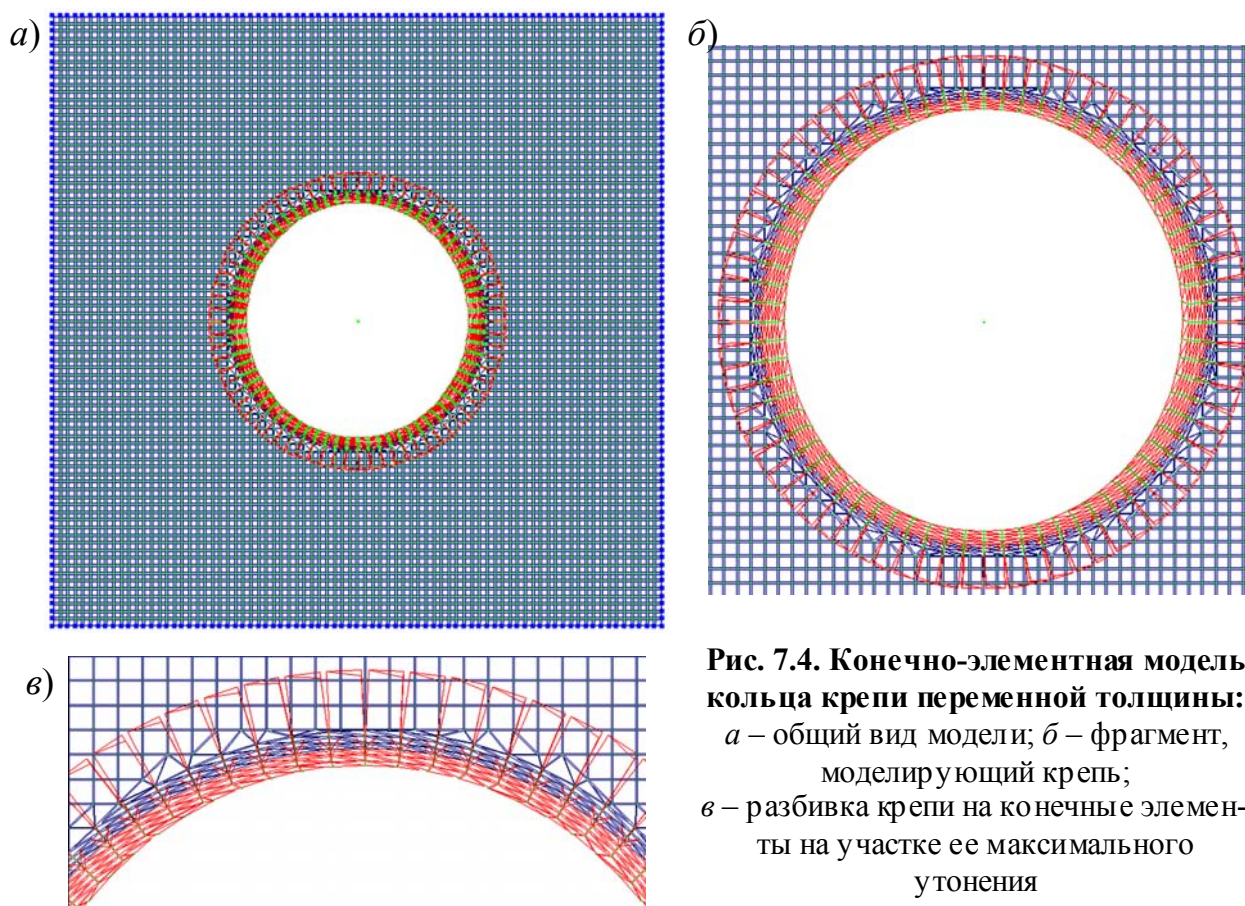
Значения параметров НДС конструкции или массива могут быть сформированы в отчет и распечатаны.

## 7.9. Примеры построения моделей в ПК ЛИРА для геомеханических исследований

Приведем примеры конечно-элементных моделей и результатов расчета НДС пород, вмещающих горные выработки.

### 7.9.1. Исследование НДС массива и бетонной крепи вертикального ствола

На рис. 7.4. показана конечно-элементная модель для исследования НДС бетонной крепи, представляющей собой кольцо переменной толщины.



**Рис. 7.4. Конечно-элементная модель кольца крепи переменной толщины:**  
*a* – общий вид модели; *б* – фрагмент, моделирующий крепь;  
*в* – разбивка крепи на конечные элементы на участке ее максимального утонения

Рассмотрим порядок построения модели (рис. 7.4).

1) *Выбор системы и начала координат.* Для построения модели используется горизонтальная координатная плоскость  $z = 0$ . Начало координат – точка  $O(0,0)$  – совпадает с точкой пересечения осей поперечного сечения ствола, т.е. с проектным центром ствола. Каждая пластина, моделирующая участок крепи или породного массива, имеет глобальную  $xu$  и собственную локальную  $x_1u_1$  систему координат, которая используется для удобства задания внешней нагрузки.

2) *Расстановка узлов модели и разбиение на конечные элементы.* В качестве узлов приняты точки, ограничивающие участки конструкции с одинаковыми механическими свойствами или определяющие пространственное расположение конструкции и граничные условия. Таким образом, кольцо крепи разбивается на близкие по размеру конечные элементы, получаемые радиальным разбиением кольца на 72 равные части, т.е. на секторы с углом  $\pi/36$  рад в направлении локальных осей  $u_1$ . Для получения более точной картины напряжений в бетоне каждый полученный конечный элемент разбит дополнительно параллельными отрезками, ориентированными вдоль локальных координатных осей  $x_1$ . Таким образом, кольцо крепи предстанет в виде 5-8 (в зависимости от толщины моделируемой крепи) тонких колец толщиной 5 см, разбитых на треугольники, конечные элементы которых жестко соединены между собой.

Для моделирования вмещающих пород задан квадратный массив размерами  $16 \times 16$  м, разбитый на квадратные конечные элементы размерами  $20 \times 20$  см, не соприкасающиеся с элементами крепи, и на подходящие по размерам треугольные конечные элементы на участках контакта массива с крепью.

3) *Определение координат узлов.* Координаты узлов крепи рассчитываются, исходя из геометрических параметров крепи и выбранного направления координатных осей. При построении моделей координаты внешнего контура определялись исходя из уравнения окружности, а внутреннего – из уравнения эллипса, смещенного относительно центра координат.

4) *Задание геометрических параметров и механических свойств элементов модели.* Для моделирования породного массива и бетонной крепи приняты пластины толщиной 10 см.

Принятые в моделях жесткостные характеристики конечных элементов приведены в табл. 7.1.

**Таблица 7.1**

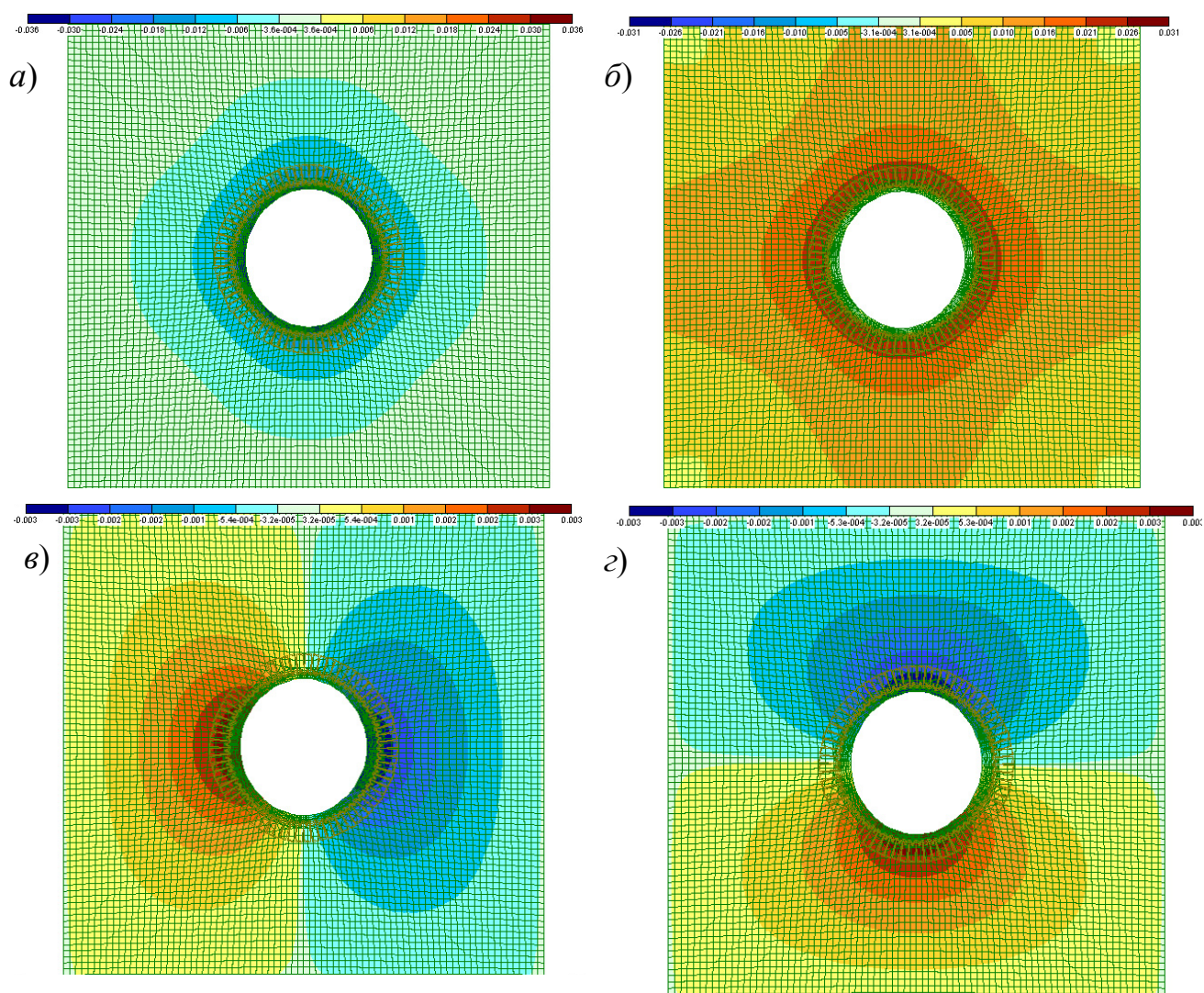
**Механические характеристики моделируемых материалов**

Моделируемый материал (крепь, порода)	Модуль Юнга $E$ , МПа· $10^{-3}$	Коэффициент Пуассона $\nu$	Удельный вес $\gamma$ , кН/м <sup>3</sup>
Бетон класса В25	30	0,2	25
Крепкий алевролит	25	0,34	27
Аргиллит	6	0,36	24
Углистый сланец	0,022	0,4	20

5) *Моделирование конечных кинематических условий.* Для закрепления моделей (создания геометрически неизменяемых систем) и моделирования связи рассматриваемого участка вмещающих пород с остальным (не попавшим под влияние ствола) массивом, все узлы, ограничивающие по контуру массив, моделируются шарнирно-неподвижными опорами.

б) *Моделирование внешней нагрузки на крепь ствола* производилась согласно расчетным формулам проф. Н.С. Булычева [1] и проф. В.Н. Каретникова [4].

Для графической интерпретации результатов расчета параметров НДС, использован графический постпроцессор программы «ЛИРА-Windows», с помощью которого для разных типов вмещающих пород построены изополя эквивалентных сжимающих и растягивающих напряжений в крепи и вмещающем массиве, изополя перемещений относительно осей  $x$  и  $y$  (рис. 7.5).



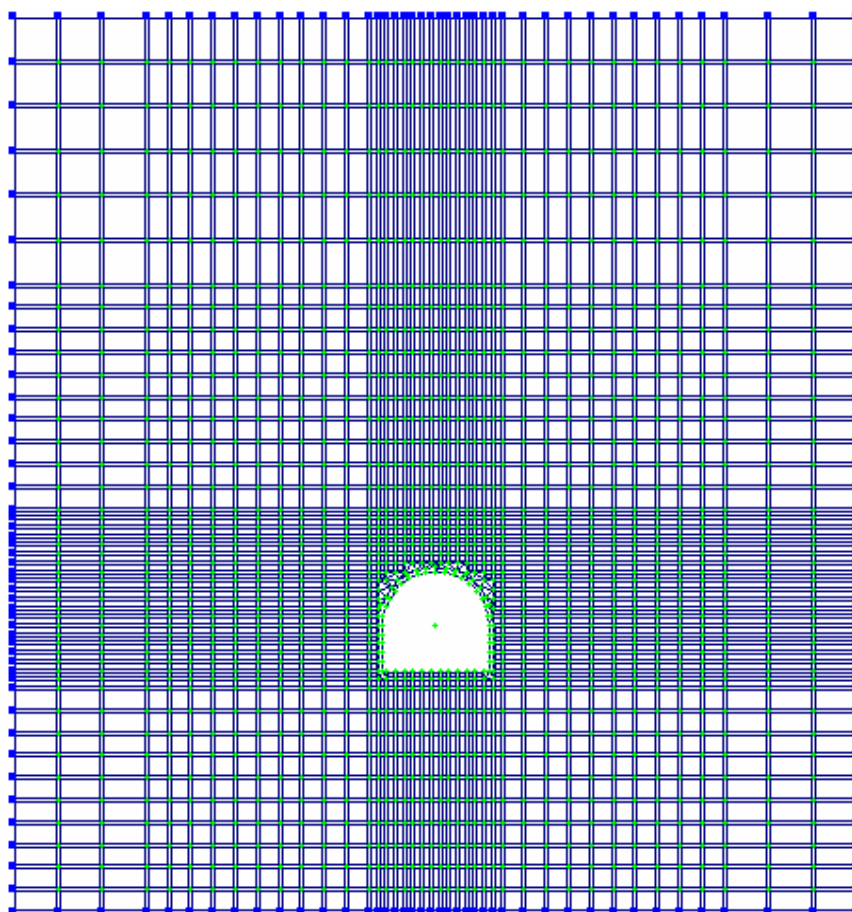
**Рис. 7.5.** Изополя, характеризующие НДС крепи и крепких пород:  
*а* – изополя сжимающих эквивалентных напряжений; *б* – то же растягивающих;  
*в, г* – соответственно изополя перемещений относительно осей  $x$  и  $y$



### ***7.9.2. Исследование НДС массива, вмещающего незакрепленную выработку арочной формы***

На рис. 7.6 приведен пример конечно-элементной модели одиночной выработки арочной формы, пройденной в однородных породах (квершлага или полевого штрека).

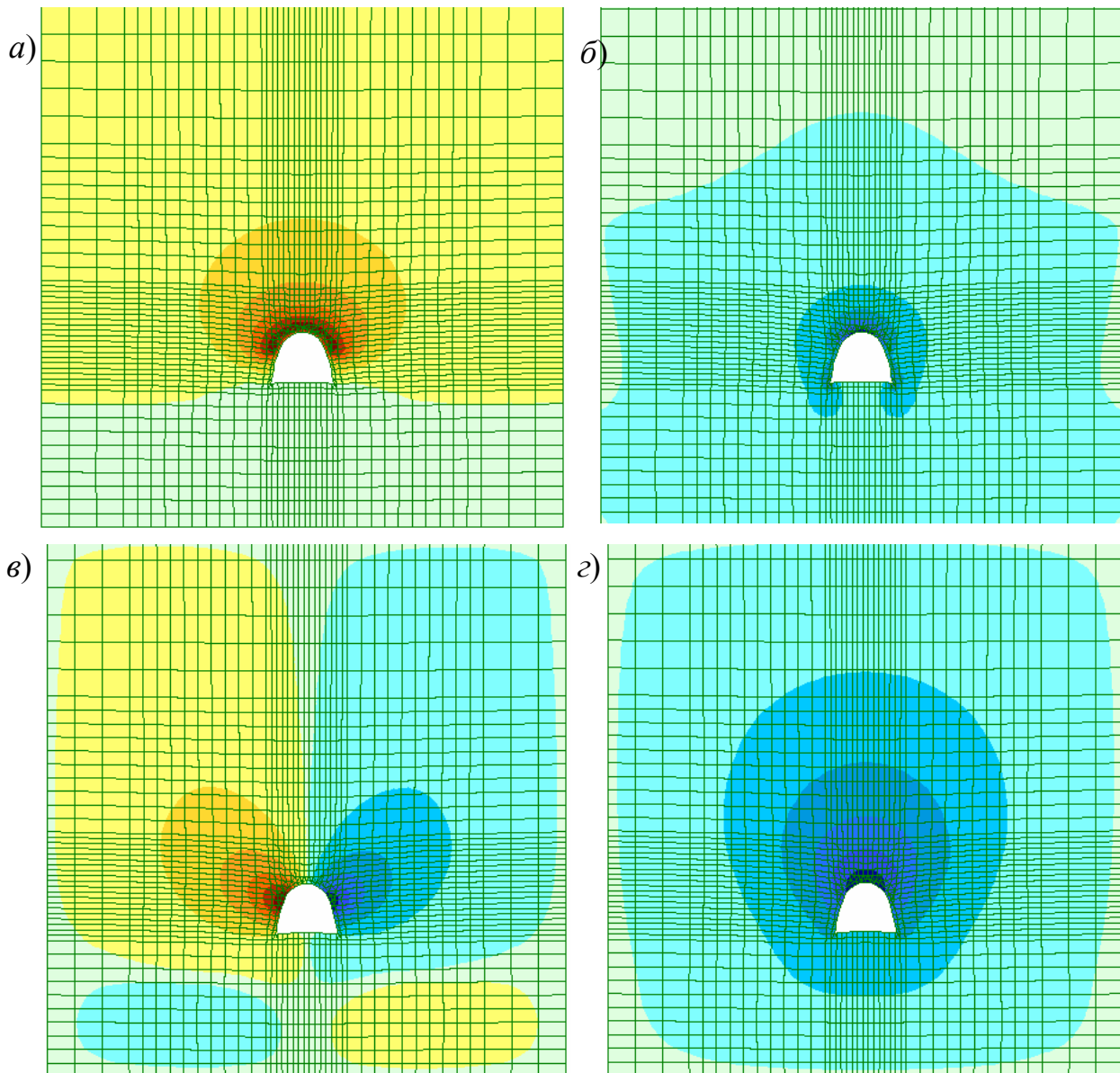
В отличие от предыдущей задачи для моделирования взята вертикальная плоскость. Разбивка модели на конечные элементы выполнена таким образом, чтобы их сетка была наиболее плотной (для большей точности картины НДС) непосредственно у контуров выработки и менее плотной (для экономии ресурсов памяти ЭВМ) – в глубине массива.



**Рис. 7.6. Пример конечно-элементной модели однородного массива, вмещающего выработку арочной формы**

Для определения параметров НДС массива при воздействии расчетных нагрузок были построены изополя перемещений узлов модели и эквивалентных напряжений в ее элементах.

Графическая интерпретация результатов расчета приведена на рис. 7.7.



**Рис. 7.7. Изополя, характеризующие НДС породного массива:**

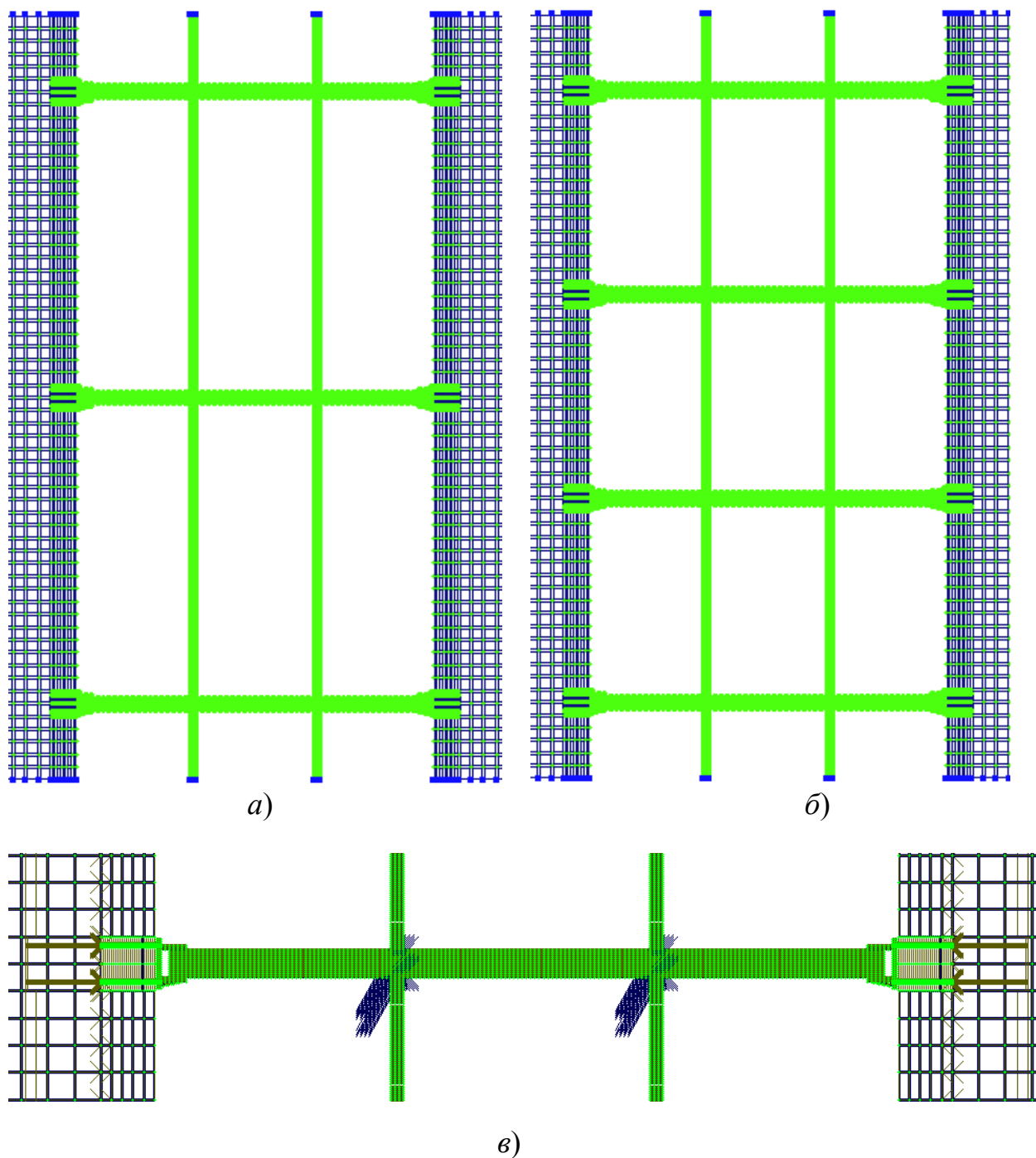
*a* – изополя сжимающих эквивалентных напряжений; *б* – то же растягивающих;  
*в, з* – соответственно изополя перемещений относительно осей *x* и *y*

Расчет методом конечных элементов позволяет получить как качественную, так и количественную картину НДС массива и прогнозировать проявления горного давления.

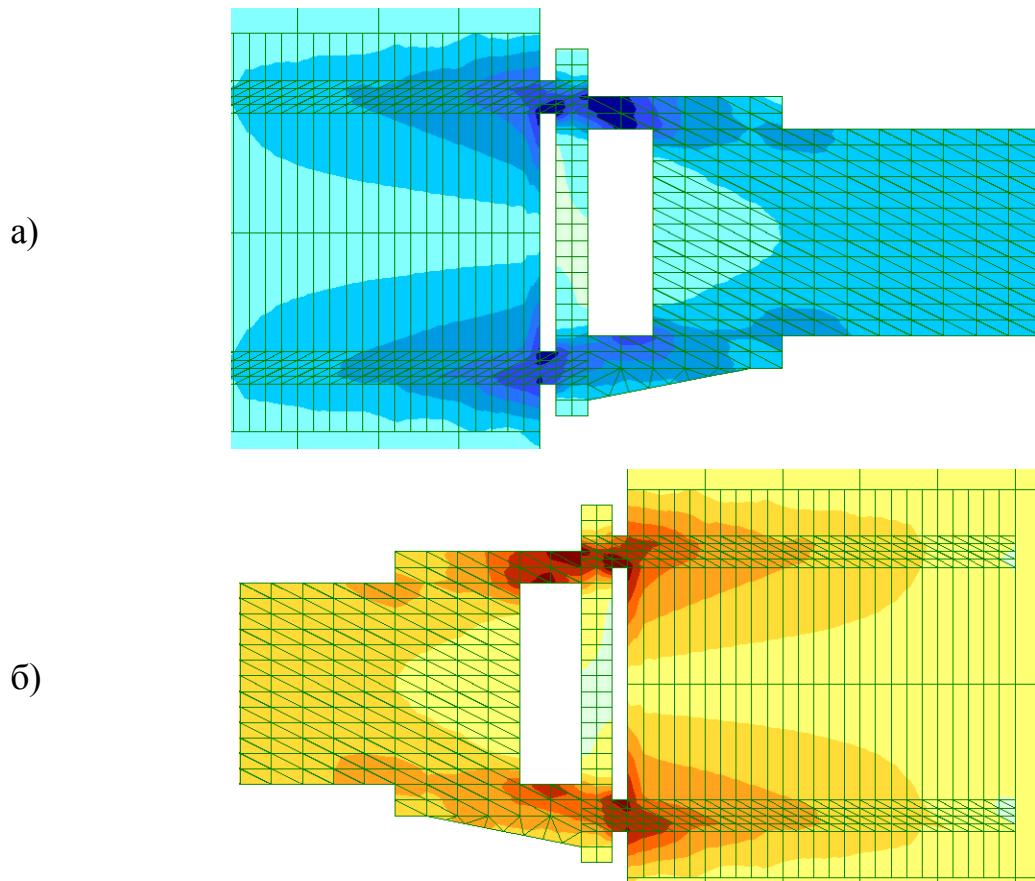
### ***7.9.3. Исследование НДС узлов крепления армировки ствола***

В ПК «Лира-Windows» можно исследовать не только однородные по свойствам объекты, но и исследовать взаимодействие элементов с совершенно различными физико-механическими характеристиками. Это позволяет моделировать податливость или, наоборот, повышенную жесткость конструкции. Рассмотрим пример моделирования жесткого и податливого узла крепления элементов армировки к крепи вертикального ствола.

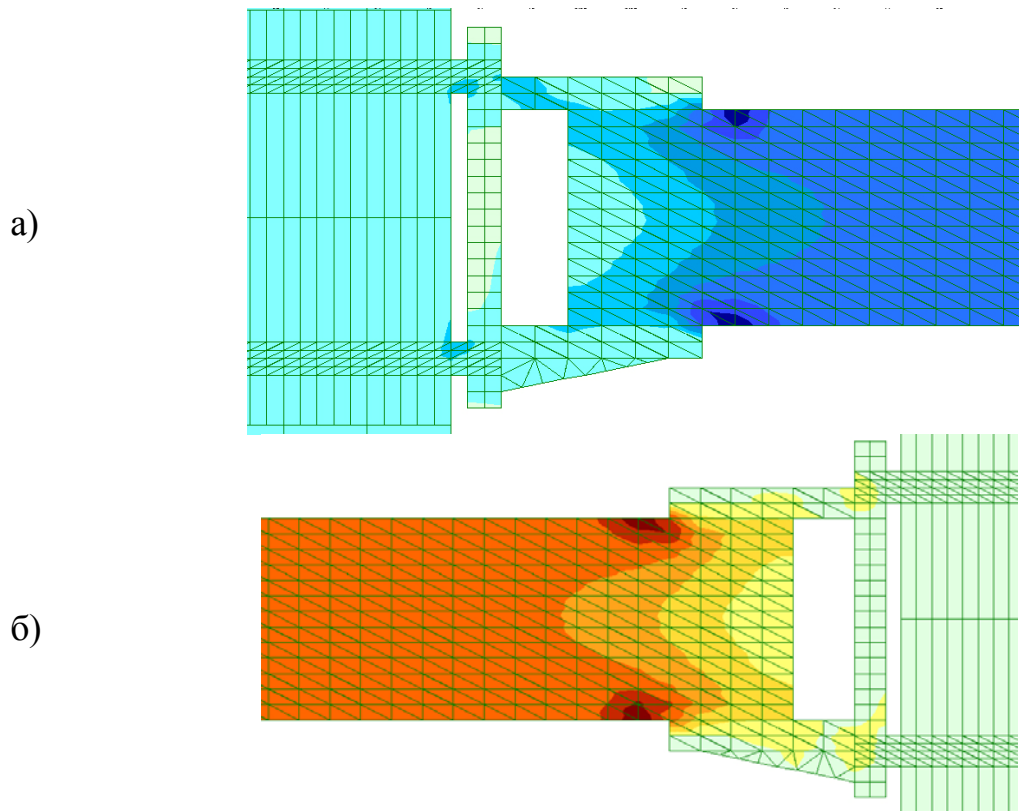
Фрагменты конечно-элементной модели приведены на рис. 7.8, а полученные в результате расчета изополя эквивалентных сжимающих и растягивающих напряжений для жесткого узла – на рис. 7.9, для податливого – на рис. 7.10.



**Рис. 7.8. Конечно-элементные модели амировки:**  
 а – с шагом 6,250 м; б – с шагом 4,168 м;  
 в – фрагмент модели с нагрузками на расстрельную балку от подъемных сосудов и со стороны породного массива



**Рис. 7.9. Изополя эквивалентных напряжений в жестком узле:**  
*a* – сжимающих; *б* – растягивающих



**Рис. 7.10. Изополя эквивалентных напряжений в податливом узле:**  
*a* – сжимающих; *б* – растягивающих

Анализ напряженно-деформированного состояния (НДС) жестких и податливых конструкций с анкерным креплением производился для различных нагрузок на армировку со стороны движущихся подъемных сосудов, которые зависят в свою очередь от массы  $m$  и скорости движения  $V$  подъемных сосудов. В результате исследования установлено, что при увеличении интенсивности подъема  $mV^2$  значительно возрастает разница между НДС жестких и податливых конструкций. Поэтому разработанные податливые конструкции имеют более широкую область применения, в которую входят не только стволы, проходимые в сложных горно-геологических условиях, но стволы с высокой интенсивностью подъема.

Таким образом, компьютерное моделирование позволило исследовать разработанные податливые конструкции, сравнить их работоспособность с аналогичными жесткими и определить область их рационального применения.

### *Контрольные вопросы*

1. Дайте понятие модели и моделирования.
2. Как классифицируются модели?
3. Какие принципы и схемы моделирования Вы знаете?
4. Какие программно-вычислительные комплексы могут использоваться для моделирования геомеханических процессов?
5. Каково назначение и состав ПВК «Лира-Windows»?
6. Из каких этапов состоит разработка конечно-элементной модели в ПВК «Лира-Windows»?

## 8. ИНТЕРНЕТ-ТЕХНОЛОГИИ

---

«...к началу следующего столетия в развитых странах основная масса информации будет храниться в памяти ЭВМ, а человек XXI века, который не будет уметь пользоваться ЭВМ, будет подобен человеку XX века, не умевшему ни читать, ни писать».

Академик В.М. Глушков

### 8.1. Краткая история создания сети Интернет

Следующим шагом после создания персональных компьютеров стала идея их объединения с целью обмена информацией между ними. Конечно, для переноса файлов можно использовать внешние носители информации, такие как дискеты, CD, USB-носители, но в тех случаях, когда требуется оперативность и надёжность передачи, эти средства не подходят. Например, дискеты, портятся, теряются, имеют небольшой объём. И вот в начале 60-х гг. XX в. американские учёные, работавшие в области компьютерных технологий, подошли к проблеме: как объединить вместе несколько компьютеров и их пользователей. Эта актуальная проблема получила всестороннюю поддержку от правительства США, которое в конце 60-х годов решило оказать финансовую помощь экспериментальной компьютерной сети. Эта сеть была учреждена Агентством Перспективных Исследований США (Advanced Research Project Agency) и получила название **ARPANET**. День рождения этой сети – 2 января 1969 года.

Пожалуй, самым важным итогом развития сети **ARPANET** стало создание сетевых протоколов, а именно семейства протоколов **TCP/IP** (Transmission Control Protocol/Internet Protocol). Протокол – это своего рода язык общения удалённых компьютеров. Таким образом, **Agranet** стала представлять собой высокоскоростную магистраль (**backbone**), обеспечивающую физическую связь между узлами (хостами). Вначале число хостов было невелико, всего 213. В середине 80-х годов Национальный научный фонд (**NSF**) создал несколько суперкомпьютерных центров, расположенных в разных частях США. Вторым шагом было объединение их в новую магистральную сеть **NSFNET** и подключение к ней региональных центров. Сеть **NSFNET** заняла место **ARPANET**, которую в итоге ликвидировали. Параллельно создавались национальные сети в других странах. К 90-м годам они объединились: таким образом возник сегодняшний Интернет.

В нашей стране Интернет появился сравнительно недавно. Пик развития сети пришёлся на 1995-96 гг. Во многом это связано с выходом в свет операционной системы **Windows 95**, значительно упрощающей подключение и настройку. На сегодняшний день Интернет доступен каждому.

## 8.2. Адресация в Интернете

Интернет представляет собой совокупность многих тысяч компьютерных сетей, объединённых в одну глобальную сеть. Рассмотрим подробнее как устроен Интернет.

Любой житель огромного мегаполиса имеет свой уникальный идентификатор – это его почтовый адрес и паспортные данные. В сети Интернет ситуация очень похожа. Каждый компьютер, подключённый к сети, имеет свой номер, так называемый IP-адрес.

IP-адреса состоят из четырёх частей, размер каждой из которых – один байт. Это означает, что каждая из четырёх частей может принимать значение от 0 до 255. Они объединены в запись, где каждая часть отделена точкой, например, 129.102.223.94. Всякий раз при передаче информации указывается IP-адрес отправителя и получателя. Означает ли это, что в голове придётся держать множество цифр? К счастью нет. Дело в том, что существует гораздо более простой способ адресации в сети – *Domain Name System* (Доменная система имён), или, сокращённо, DNS.

Доменное имя компьютера – это уникальное алфавитно-цифровое имя, как правило, несущее смысловую нагрузку и гораздо легче запоминаемое, чем IP-адрес. Оно состоит из нескольких частей, причём домен верхнего уровня находится в имени справа, а домен нижнего уровня слева. Как правило, в среднем используется 3-5 уровней. Возьмем два уровня и поясним их организацию.

microsoft.com

http://home.projects.alpha.com/

В большинстве случаев, говоря об адресации в сети Интернет, имеется в виду такая форма записи: www.microsoft.com или http://www.microsoft.com/

В этих именах представлены домены двух и четырёх доменов соответственно. Начнём с домена верхнего уровня – «com». Домены верхнего уровня можно условно поделить на две категории: стандартные и организованные по региональному признаку. В первом случае имя домена установлено организацией, контролирующей подключение к Интернет (Network Information Center). Таких доменов шесть:

- com – коммерческие компании
- edu – образовательные учреждения
- gov – правительственные учреждения
- mil – военные учреждения
- net – сетевые организации
- org – другие организации

Эти домены в основном используются в Америке. Во втором случае имя домена определяется по территориальному признаку. Такое имя состоит из двух букв:

- ru – Россия
- uk – Великобритания

- dk – Дания
- de – Германия
- fr – Франция
- ua – Украина и т.д.

Домен второго уровня, как правило, совпадает с именем фирмы или организации. У большинства крупных компаний адрес очень простой, например, pentagon.mil, pepsi.ru и т.д. Откуда берётся домен третьего уровня? Предположим, крупная компания содержит ряд подразделений, тогда каждому подразделению выделяют свой домен.

### 8.3. Функции Интернет

Глобальная мировая сеть Internet выполняет следующие основные функции

- *Интернет – кладовая информации.*

Представим себе, что лет десять назад нам срочно надо было бы узнать какую-нибудь важную информацию. Скорее всего, мы обратились бы в библиотеку или в справочную службу. Это требовало определённых усилий и временных затрат. Более того, не во всех библиотеках есть та информация, которая нужна нам. И так далее – масса проблем и неудобств. Теперь представим себе другую картину. Мы сидим за своим персональным компьютером, входим в сеть Интернет и буквально за считанные минуты получаем тексты, фотографии, музыку, видео. Можно воспользоваться Британской энциклопедией или познакомиться с последними разработками NASA в области космических исследований. Мы можем взглянуть на фотографию обратной поверхности Луны или насладиться пейзажами пустынь. Мы можем послушать последние хиты клубной музыки. А главное, мы можем сохранить все эти данные у себя в компьютере и использовать их в работе.

- *Оперативные новости со всего мира.*

Действительно, в сети можно получить наиболее свежую информацию об общественно-политических, финансово-экономических, спортивных событиях по всему миру. Информация появляется раньше, чем в газетах и аналитических журналах. Кроме того новости ежечасно (а бывает и ежеминутно) обновляются.

- *Интернет – бизнес и финансы.*

Финансисты, работники банков и биржевые игроки – вот люди, чья успешная работа зависит от свежих новостей. Котировки акций, курс валют, оптовые, розничные цены на товары, услуги и т.д. – всё это более чем доступно в сети.



- *Интернет – обучение.*

Работая за своим компьютером, можно получить сертификаты различных курсов и дипломы университетов. Можно учить иностранные языки или экономические науки.

- *Интернет – развлечение и отдых.*

Вот уж чего в Интернете хватает, так это различной забавной и полезной (или бесполезной) ерунды. Анекдоты, шутки, картинки, слухи, сплетни, смешные истории. Всё, что душе угодно на любом языке и в любом количестве.

- *Интернет – общение в реальном времени.*

Ничто не затягивает глубже в сети Интернет, чем «чат» – общение с людьми из различных уголков мира в реальном времени. То есть, мы у себя дома печатаем текст на клавиатуре, а наш партнер сразу же видит его на своем экране и уже готов печатать ответ. Такой диалог возможен, как с одним, так и с несколькими партнёрами одновременно.

- *Интернет – самая быстрая и надёжная почта.*

Электронная почта – понятие неотделимое от понятия «сеть Интернет». Исторически почта была одним из самых первых видов сервиса и использовалась для передачи личных сообщений. Передача сообщений предполагает наличие у нас и у получателя специального компьютерного почтового адреса и определённых программных средств для создания, отправки и получения письма. Сейчас помимо передачи служебной и личной информации с посланием можно отправить какой-нибудь небольшой файл.

- *Интернет – хранилище файлов.*

Именно в сети Интернет можно найти самые последние версии популярных, а также редких программ, самые новые дополнения и исправления к уже вышедшим версиям, самые новые антивирусы и многое, многое другое.

- *Интернет – новые возможности с каждым годом.*

Развитие сети Интернет открывает всё новые и новые горизонты перед пользователями.

#### **8.4. Подключение к сети Интернет**

Работать с сетью можно либо через выделенную линию, либо через телефонную сеть с использованием модема. В первом случае можно стать счастливым обладателем высокоскоростного соединения, без помех и ослабления сигнала. Но этот способ является более дорогим. Во втором случае используется телефонная линия. Это наиболее распространённый ва-

риант работы с сетью для частных лиц. Первое, что понадобится для выхода в Интернет через телефонную линию, это *модем* (**МО**дулятор-**ДЕ**Модулятор), устройство, посылающее и принимающее данные. Модемы разделяют на два типа: внутренние и внешние. Внутренний модем представляет собой плату расширения, которая вставляется в один из слотов на материнской плате. Внешние модемы – те же платы, но в отдельном пластмассовом корпусе.

Весьма важной характеристикой модема является скорость передачи данных, измеряемая в BPS (биты в секунду). Первые модемы имели скорость 300 или 1200 bps. У современных модемов скорость доходит до 200 000 bps и более.

После того, как модем куплен, провайдер выбран, и вся необходимая информация получена, нам остается только настроить операционную систему компьютера для работы в сети Интернет.

Для настройки подключения к Интернет операционной системы Windows98/ME и выше необходимо найти в главном меню программу «Мастер подключения к Интернет» и запустите её. Все настройки протоколов и драйвера мастер установит самостоятельно.

Для доступа к информации необходимо запустить браузер (сетевую программу для работы в сети Интернет на персональных ЭВМ. Наиболее популярны браузер Nestcape Navigator фирмы Nestcape и браузер Internet Explorer фирмы Microsoft для операционной системы Windows.

Рассмотрим, как устроено окно браузера. Верхняя часть окна отведена под заголовок. Здесь отображено название программы и название текущей страницы. Ниже строка меню и панель инструментов. Под панелью мы видим адресную строку. В самом низу находится строка состояния.

На панели инструментов расположено несколько кнопок, помогающих нам осуществлять переходы. Поясним их значения:

- «Назад». Открыв несколько документов, мы можем возвращаться обратно на предыдущие странички.
- «Вперёд». Если мы вернёмся на несколько шагов назад, эта кнопка станет доступной. Она выполняет обратную функцию.
- «Остановить» Если документ долго загружается или мы нечаянно переходим на не нужную нам страницу, эта кнопка отменит переход.
- «Обновить». Есть документы, обновление которых происходит каждую минуту или каждые несколько минут. Данная кнопка поможет повторно загрузить данную страницу.
- «Домой». Эта кнопка сразу же перенесёт нас на домашнюю страницу.

## **8.5. Поиск информации в Интернет**

Для нахождения необходимого документа нужно знать адрес странички. Какие же адреса могут понадобиться? Прежде всего, это –

[www.rambler.ru](http://www.rambler.ru). – наиболее удобная и популярная поисковая система в российском Интернете. Содержит миллионы документов с более чем 250000 сайтов. Имеет развитый язык запросов и гибкую форму вывода результатов. На страничке можно видеть список ссылок, сгруппированных по темам. Этот список озаглавлен Top100, то есть лучшие сайты Интернет, по рейтингу Rambler. Там можно найти почти всё, что нужно. Также можно воспользоваться «Поиском по ключевому слову» в самом начале страницы. Для этого в пустое поле необходимо ввести запрос и нажать кнопку «Поиск». Через несколько секунд откроется результат поиска.

Приведем примеры некоторых интересных адресов:

Адрес	Краткое описание
<a href="http://www.anecdotov.net">http://www.anecdotov.net</a> .	Шутки, анекдоты, карикатуры, забавные фотографии.
<a href="http://www.atlant.ru">http://www.atlant.ru</a>	Прайс-листы по товарам и услугам в Санкт-Петербурге
<a href="http://www.auto.ru">http://www.auto.ru</a>	Автомобили в России
<a href="http://www.doctor.ru">http://www.doctor.ru</a>	Лучший сайт о медицине, здоровье и нездоровье
<a href="http://www.freeware.ru">http://www.freeware.ru</a>	Сборник полезных программ, утилит и файлов бесплатно
<a href="http://www.job.ru">http://www.job.ru</a>	Поиск работы
<a href="http://www.kino.ru">http://www.kino.ru</a>	Новости кино
<a href="http://www.kiss.ru">http://www.kiss.ru</a>	Клуб знакомств
<a href="http://www.lenta.ru">http://www.lenta.ru</a>	Новости
<a href="http://www.mail.ru">http://www.mail.ru</a>	Бесплатная почта
<a href="http://www.mp3.ru">http://www.mp3.ru</a>	Музыка
<a href="http://www.rbc.ru">http://www.rbc.ru</a>	Финансовые новости и аналитическая информация
<a href="http://www.memstick.chat.ru">http://www.memstick.chat.ru</a>	Бумага для записок
<a href="http://www.video.ru">http://www.video.ru</a>	Новости видеорынка
<a href="http://www.wallpapers.ru">http://www.wallpapers.ru</a>	Обои для рабочего стола

Наиболее распространенными среди отечественных поисковых систем являются:

- yandex.ru – наиболее полная система поиска информации;
- aport.ru – наиболее удобная система поиска информации;
- rambler.ru – самая популярная система поиска.

Среди международных информационно-поисковых систем наиболее известны: Alta Vista, Yahoo, Infoseek. Эти системы позволяют выражать запросы и искать информацию не только на английском, но и на русском, а также испанском, французском, немецком и др. языках.

С начала 90-х гг. интенсивно развиваются справочные службы Интернет, помогающие пользователям найти нужную информацию, и эти службы можно разделить на две категории: универсальные и специализированные.

В универсальных службах используется обычный принцип поиска в неструктурированных документах – по ключевым словам.

Ключевым словом документа называется отдельное слово или словосочетание, которое каким-то образом отражает содержание данного документа. Во многих текстовых процессорах ключевым словом является текст, по которому осуществляется поиск нужной информации.

Универсальная служба поиска (поисковая система) – это комплекс программ и мощных компьютеров, выполняющих следующие функции.

- *Специальная программа (поисковый робот)* непрерывно просматривает страницы «Всемирной паутины», выбирает ключевые слова и адреса документов, в которых эти слова обнаружены. Web-сервер принимает от пользователя запрос на поиск, преобразует его и передает специальной программе – поисковой машине.
- *Поисковая машина* просматривает базу данных индексов, составляет список страниц, удовлетворяющим условиям запроса (точнее список ссылок на эти страницы) и возвращает его Web-серверу.
- *Web-сервер* оформляет результаты выполнения запроса в удобном для пользователя виде и передает их на машину клиента.

Специализированные справочные службы – это тематические каталоги (subject catalogs), в которых собраны более или менее структурированные сведения об адресах серверов по той или иной тематике. В отличие от универсальных баз индексов, тематические каталоги составляются специалистами и обеспечивают клиента более строгой, достоверной систематизированной информацией о Сети.

Кроме того, многие сайты Интернет располагают собственными механизмами поиска (в пределах данного сайта).

## **8.6. Навигация в World Wide Web**

*Всемирная паутина (World Wide Web или WWW)* на сегодняшний день это наиболее продвинутый и интересный ресурс – гипертекстовая система навигации в Internet. От обычного гипертекста WWW отличается главным образом тем, что позволяет устанавливать ссылки не только на соседний файл, но и на файл, находящийся на компьютере в другом полушарии Земли. От вас же не требуется никаких усилий – компьютер устанавливает связь самостоятельно. В World Wide Web к ресурсам можно обращаться, непосредственно, указывая их адрес.

Система World Wide Web является попыткой интегрировать разнообразные инструментальные системы и данные с помощью общего форма-

та данных на базе концепции гипертекста. Результат этих разработок был чрезвычайно успешным. Фактически, они сформировали к настоящему времени лицо сети Internet.

Во Всемирной паутине встречается немало интересного, но особенно привлекательно здесь выглядят **главные страницы**. Благодаря средствам просмотра WWW хаотические джунгли информации в Internet приобретают форму привычных аккуратно оформленных страниц с текстом и фотографиями, а в некоторых случаях даже с видеосюжетами и звуком. Привлекательные титульные страницы сразу же помогают понять, какая информация последует дальше. Каждое ключевое слово соединяется с соответствующими информационными файлами посредством гипертекстовых связей. Они содержат текст, графическое и звуковое оформление, видеоизображения и другие **мультимедийные** элементы, позаимствованные у различных средств информации. Кроме того, главные страницы пестрят **гиперссылками**. Путешествуя по Internet, мы переходим с сайта на сайт и с одной страницы Всемирной паутины на другую. Иными словами, World Wide Web – это все, с чем мы обычно сталкиваемся в Internet и что видим на экране нашего монитора.

Система Web базируется на методе связывания слов и фраз в документе для ссылки к соответствующей информации в этом же или другом документе. Поскольку другие документы могут быть на разных серверах, эти ссылки образуют своеобразную «паутину» взаимных связей, которая пронизывает сеть Internet.

Появление этого сервиса значительно упростило доступ к информации и стало одной из основных причин взрывообразного роста Internet с 1990 г.

Чтобы впервые попасть на желаемый *Web-сайт*, надо набрать соответствующий *URL-адрес* в адресной строке «Обозревателя» и нажать клавишу ENTER. При этом имя протокола **http://** разрешается не указывать – он подразумевается сам собой. Адреса полезных ресурсов можно узнать из книг, журналов, сообщений СМИ. Далее в пространстве WWW можно перемещаться с помощью гиперссылок.

Работая в Интернете, желательно как можно реже вводить что-либо с клавиатуры. При ручном наборе адресов неминуемо возникают ошибки. Лучший способ борьбы с ними – сокращение ручного ввода. Механизм закладок позволяет практически полностью исключить ручной ввод адресов. Закладки – это небольшие файлы, в которых запоминаются *URL-адреса* ранее посещённых Web-страниц. Разыскав какую-либо Web-страницу, содержание которой выглядит полезным и интересным, можно сохранить её адрес в виде закладки. Для этого служит команда Избранное > Добавить в Избранное.

Чтобы в будущем посетить Web-страницу, адрес которой сохранен в папке Избранное, надо дать команду Избранное и выбрать нужную страницу в раскрывшемся меню.

## 8.7. Сохранение Web-страниц

Просматриваемые Web-страницы можно сохранить на жёстком диске в виде файлов. Сохранение возможно для страницы, открытой в данный момент. Команда сохранения – Файл → Сохранить как. Она открывает окно сохранения файла, в котором следует:

- выбрать в раскрывающемся списке Папка имя папки, в которую Web-документ будет сохранен;
- в поле Имя документа проверить сформированное там имя и при необходимости изменить;
- в поле Тип файлов выбрать метод сохранения.

Web-страницы – это комплексные документы, содержащие кроме обычного текста команды форматирования и встроенные объекты. В зависимости от того, что мы хотим получить, существуют разные методы их сохранения.

Если выбран тип Текстовый файл, то сохраняется только текст. Его можно просматривать и изменять в любом текстовом редакторе. Такой метод применяют, когда в документе важно только его содержание.

Если выбран тип Веб-страница, только HTML, то кроме текста сохраняются команды форматирования. Просматривать документ, сохраненный таким способом, можно либо в «Обзревателе», либо в текстовом редакторе.

Самый мощный метод сохранения – Веб-страница полностью. В этом случае документ сохраняется вместе со всеми встроенными объектами (рисунками, аудио- и видеоклипами и прочими). При таком сохранении кроме файла в формате .HTM (собственно Web-страница) сохраняется также одноименная папка, в которую и помещаются все объекты, встроенные в текст.

## 8.8. Электронная почта

*Электронная почта* – это самостоятельная служба Интернета весьма почтенного возраста: ей более 30 лет. Поскольку это служба, в ней можно выделить две программы – *почтовый сервер* и *почтовый клиент*. Почтовые серверы работают на узловых компьютерах Интернета, а почтовый клиент должен быть у пользователя. Простейший почтовый клиент – программа *Microsoft Outlook Express*. Он входит в стандартный пакет *Windows*. Все сервис-провайдеры, обеспечивающие общее подключение к Интернету, предоставляют и возможность подключения к почтовому серверу.

Электронная почта основана на принципе эстафеты. Сначала сообщение готовится в почтовом клиенте. Затем оно передается на почтовый сервер сервис-провайдера и далее пересылается по цепочке почтовых серверов, пока не достигнет сервера адресата. Как только адресат подклю-

чится к своему почтовому серверу, он автоматически получит всё, что накопилось в его «почтовом ящике».

Электронная почта работает очень надёжно и оперативно. Срок доставки сообщения в любую точку земного шара составляет от нескольких секунд до нескольких часов.

#### *Адрес электронной почты.*

При создании учётной записи на почтовом сервере каждый клиент получает уникальный почтовый адрес. Он состоит из двух частей – регистрационного имени клиента и доменного имени сервера. Разделяются две части адреса символом «@», который играет роль предлога «в».

Например, адрес petrov@yandex.ru – это почтовый адрес лица, имеющего регистрационное имя petrov в почтовой базе сервера с доменным именем yandex.ru. Уникальность адресов электронной почты обеспечивается тем, что в момент создания учётной записи сервер не позволит клиенту избрать регистрационное имя, совпадающее с именем ранее зарегистрированного клиента.

#### *Почтовый клиент Outlook Express.*

Программа Microsoft Outlook Express – это стандартный почтовый клиент для работы с электронной почтой E-Mail. В мире существует немало и других почтовых клиентов. Они различаются своими функциями и уровнем автоматизации операций.

На рис. 8.1 показано рабочее окно программы Outlook Express. В нём четыре важных панели: панель папок, панель заголовков, панель сообщений и панель контактов.

Левая верхняя панель – это **панель папок**. Основных папок – пять. Все они предназначены для упорядочения работы с сообщениями. Их названия говорят за себя сами. В папке Входящие «хранятся» поступившие сообщения, а в папке Исходящие – сообщения, подготовленные к отправке, но ещё не отправленные. После отправки они перемещаются в папку Отправленные, а если их удалить, то в папку Удалённые.

Нетрудно догадаться, что в папке Удалённые показываются сообщения, для которых поставлена пометка «Удалено», хотя на самом деле никуда они не удалялись и продолжают храниться там же, где хранились.

Правая верхняя панель окна программы – это **панель заголовков сообщений**. Когда на панели папок выделена одна из папок, здесь отображаются заголовки сообщений, которые к ней относятся. Панель служит для просмотра служебных полей, входящих в заголовок. Наиболее важны поля, в которых указано, от кого и когда поступило сообщение, а также тема сообщения.

**Панель сообщений** находится под панелью заголовков. Когда на панели заголовков выделен заголовок какого-либо сообщения, на панели сообщений отображается его содержание. Выделите другой заголовок – увидите другой текст.

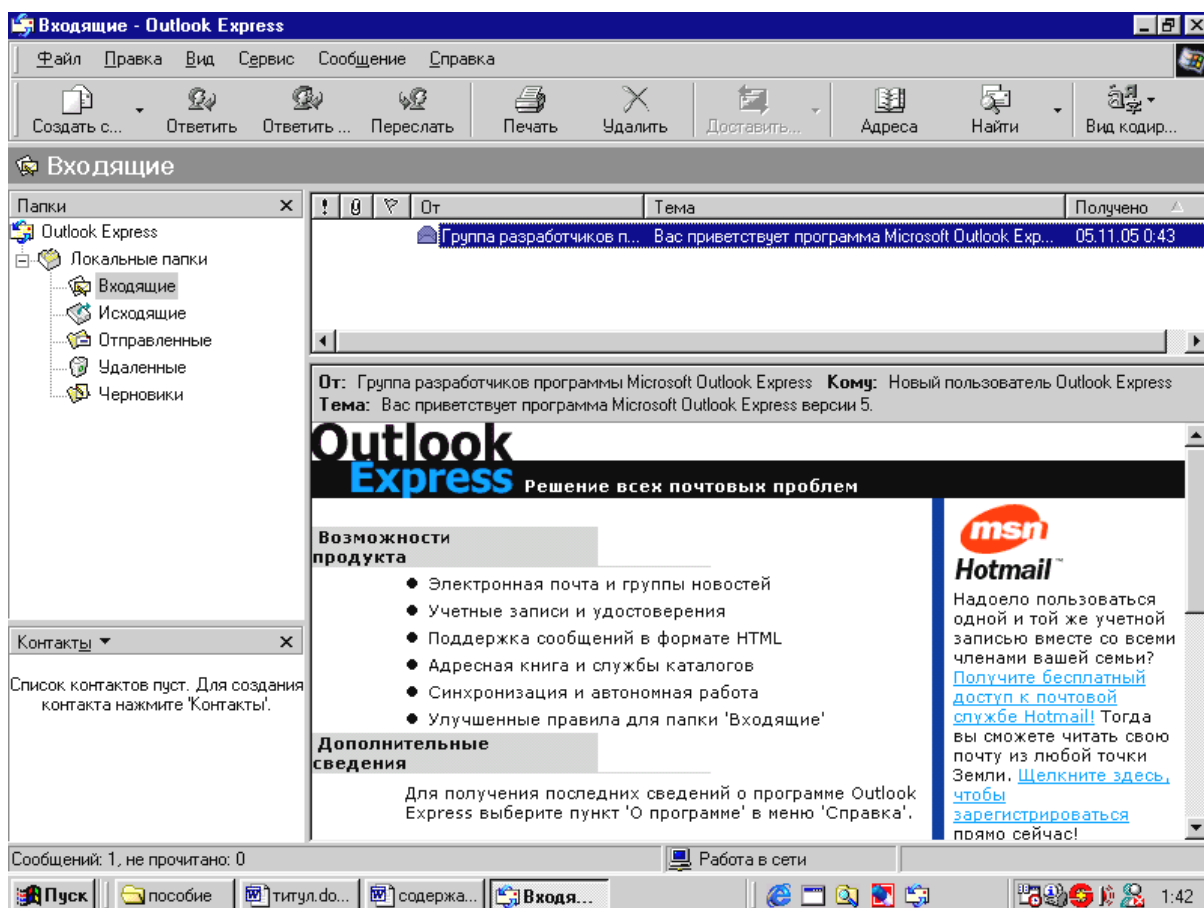


Рис. 8.1. Интерфейс программы Microsoft Outlook Express

В левом нижнем углу окна программы находится небольшая, но очень важная **панель контактов**. Она удобна для быстрой подготовки сообщений постоянным корреспондентам. Для приёма сообщений электронной почты она не нужна.

Обычно **приём сообщений** происходит автоматически во время соединения с сервером. Порядок действия при этом примерно такой:

1. Устанавливаем соединение с сервис-провайдером с помощью значка настроенного соединения удалённого доступа.

2. Когда соединение установится, запускаем программу Outlook Express. Обнаружив в момент запуска наличие работающего соединения, она сама соединится с почтовым сервером и примет накопившуюся там почту.

3. При приёме корреспонденции рядом с папкой Входящие отмечается число поступивших сообщений. Если новых сообщений нет, то в строке состояния программы появляется запись Нет новых сообщений.

Поступившую почту можно просмотреть, сохранить или удалить. На неё можно также ответить. Если предполагается в будущем вступить в переписку с отправителем письма, имеет смысл внести его адрес в свой список контактов.

Щёлкните правой кнопкой мыши на заголовке сообщения и в открывшемся контекстном меню выберите пункт Добавить отправителя в



адресную книгу. В списке контактов образуется запись, соответствующая данному лицу, и в будущем не придётся вручную вводить его адрес.

## 8.9. Некоторые ресурсы Интернета по горному делу

В этом разделе приведем некоторые ссылки на Интернет-ресурсы, которые могут оказаться полезными для студентов вузов горных специальностей, горных инженеров, аспирантов и всех интересующихся вопросами горного производства.

В табл. 8.1. приведены ссылки на порталы и сайты, посвященные горному делу.

Таблица 8.1.

Сайты, содержащие информацию по горному делу

Ссылка в Интернет	Наименование сайта	Краткая характеристика
<a href="http://www.gornoe-delo.ru/">http://www.gornoe-delo.ru/</a>	«Горное дело»	Информационно-справочный сайт о горной промышленности
<a href="http://www.minprom.gov.ru/">http://www.minprom.gov.ru/</a>	Сайт Министерства промышленности и энергетики РФ	Новости и нормативная база промышленности и энергетики
<a href="http://www.gosnadzor.ru/">http://www.gosnadzor.ru/</a>	Сайт Ростехнадзора РФ	Материалы по безопасности в горной промышленности
<a href="http://www.rosugol.ru/">http://www.rosugol.ru/</a>	Информационный портал Российский уголь	Многосторонний информационный ресурс о горной промышленности РФ
<a href="http://www.mine.info.donntu.edu.ua">http://www.mine.info.donntu.edu.ua</a>	Информационный портал по горному делу ДонНТУ	<b>Полнотекстовые учебные, учебно-методические и научные материалы по горному делу, оперативная информация о конференциях и симпозиумах</b>
<a href="http://miningpage.narod.ru/index.html">http://miningpage.narod.ru/index.html</a>	Страничка горняков Украины	Авторефераты диссертаций, журналы, законы, научные исследования в области горного дела
<a href="http://mpe.energy.gov.ua">http://mpe.energy.gov.ua</a>	Сайт Министерства топлива и энергетики Украины	Официальная информация министерства
<a href="http://www.ukrcoal.com/">http://www.ukrcoal.com/</a>	Сайт «Уголь Украины»	Полезная информация и большое количество ссылок на горнодобывающие, шахтостроительные и научно-исследовательские организации Украины
<a href="http://www.mining.kz">http://www.mining.kz</a>	Казахстанский горно-промышленный портал	Ссылки на Интернет-ресурсы по горной тематике

Большой интерес для студентов-шахтостроителей представляет информационный портал «Горное дело» (рис. 8.2), открытый в феврале 2006 г. в Донецком национальном техническом университете на базе материалов, разработанных и собранных сотрудниками кафедр «Строительство шахт и подземных сооружений» горно-геологического факультета ДонНТУ и «Подземное, промышленное, гражданское строительство и строительные материалы» Шахтинского института (филиала) ЮРГТУ(НПИ). Его адрес в Internet:

[www.mine.info.donntu.edu.ua](http://www.mine.info.donntu.edu.ua)

Рис. 8.2. Главная страница информационного портала «Горное дело»

В настоящее время объем материалов портала, количество ВУЗов и организаций, обеспечивающий портал информацией, а также число пользователей (посетителей портала) существенно возросло. К их числу относятся Донецкий национальный технический университет (г. Донецк, Украина), Национальный горный университет (г. Днепропетровск, Украина), Донбасский государственный технический университет (г. Алчевск, Луганская обл., Украина), Шахтинский институт (филиал) ЮРГТУ(НПИ) (г. Шахты, РФ), НТЦ «Наука и практика» (г. Ростов-на-Дону, РФ), МГГУ (г. Москва, РФ), Криворожский технический университет (г. Кривой Рог, Украина), ТулГУ (г. Тула, РФ).

В результате, в одном месте собрана научно-техническая, учебно-методическая и научная информация по горному делу по специальностям высшего профессионального образования «Шахтное и подземное строительство», «Подземная разработка месторождений полезных ископаемых», «Безопасность технологических процессов и производств (в горной промышленности)», а также научным специальностям горного профиля и др.

Портал состоит из нескольких разделов, а именно:

– **Актуальные новости.** Здесь регулярно публикуется оперативная информация о последних пополнениях портала – размещении новых учебных, учебно-методических, научных, нормативных и других материалов;

– **Конференции.** В данном разделе размещается официальная информация организаторов о конференциях, симпозиумах, форумах, семинарах, проводимых по горной тематике в России, Украине и других странах ближнего и дальнего зарубежья. По каждой конференции дается подробная программа, условия участия и опубликования материалов. Также здесь имеется оперативная информация о постоянно действующих конференциях и форумах, проводимых на базе вышеперечисленных вузов и организаций. В настоящее время на портале доступно более 60 ссылок на конференции по горной тематике, проводившиеся в 2003-06 гг., информация о новых конференциях регулярно пополняется;

– **Учебники и монографии.** В данном разделе находятся полнотекстовые электронные версии учебников, учебных, учебно-методических пособий и монографий по горной тематике, и, прежде всего, по шахтному и подземному строительству. Общее число таких электронных документов – более 50, в том числе ставшие редкими, издания 50-80-х гг. XX в. (авторы – известные ученые-шахтостроители Н.М. Покровский, А.П. Максимов, С.А. Федоров, В.В. Белый и др.). Чтобы сохранить и сделать общедоступными эти книги, была выполнена их оцифровка и размещение на информационном портале. Здесь же размещены и самые новые учебники и монографии, опубликованные в 2000-07 гг. известными учеными-горняками России (профессора Булычев Н.С., Сыркин П.С., Мартыненко И.А., Страданченко С.Г. и др.) и Украины (профессора Шашенко А.Н., Шевцов Н.Р., Левит В.В., Сдвижкова Е.А., Лысиков Б.А., Усаченко Б.М., Булат А.Ф. и др.). Работа по размещению на портале вновь изданных и оцифровке старых (но, зачастую, не менее полезных для студентов и ученых) учебников и монографий продолжается;

– **Статьи и сборники.** В данном разделе размещены полнотекстовые документы, включающие отдельные научные статьи и сборники научных трудов ДонНТУ, ШИ(ф) ЮРГТУ(НПИ)(ф), ДонГТУ, НГУ и других горных ВУЗов. В настоящее время раздел представлен более 30 сборниками, вышедшими за последние 5 лет, и продолжает пополняться;

– **Авторефераты.** С целью обеспечения быстрого и удобного поиска материалов по защищенным докторским и кандидатским диссертациям

на портале создан данный раздел, включающий более 60 полнотекстовых авторефератов диссертаций по научным специальностям «Маркшейдерия», «Механика грунтов и горных пород», «Шахтное и подземное строительство», «Физические процессы горного производства», «Охрана труда» (по рубрикатору научных специальностей Украины), а также «Геотехнология (открытая, подземная, строительная)», «Геомеханика, разрушение горных пород, рудничная аэрогазодинамика и горная теплофизика», «Горнопромышленная и нефтегазопромысловая геология, геофизика, маркшейдерское дело и геометрия недр» и др. (по рубрикатору научных специальностей России). Раздел постоянно пополняется по мере новых защит диссертаций, выполненных или защищенных в вышеперечисленных ВУЗах, подписавших договора о сотрудничестве;

– **Методические указания.** Раздел предназначен в основном для студентов горных специальностей, но может быть полезен и преподавателям для учебно-методической работы. Раздел содержит более 100 методических разработок к выполнению дипломных и курсовых проектов, практических и лабораторных работ, к самостоятельному изучению отдельных дисциплин специальности «Шахтное и подземное строительство» и других специальностей горного профиля.

Таким образом, размещение на едином портале статей, материалов конференций, монографий, учебников и учебных пособий, методического обеспечения, авторефератов диссертаций по тематике портала значительно облегчило доступ к научно-технической информации.

В перспективе не исключается возможность создания и проведения на базе информационного портала международных научно-технических конференций в режиме ONLINE.

В табл. 8.2 приведены ссылки на официальные сайты основных журналов по горной тематике.

**Таблица 8.2**

**Сайты журналов по горной тематике**

Ссылка в Интернет	Наименование журнала
<a href="http://www.rosugol.ru/jur_u/ugol.html">http://www.rosugol.ru/jur_u/ugol.html</a>	Уголь
<a href="http://www.rudmet.ru/gurnal.php?idname=1">http://www.rudmet.ru/gurnal.php?idname=1</a>	Горный журнал
<a href="http://www.gornoe-delo.ru/magazine/gp.php?v=list&amp;gp=52005">http://www.gornoe-delo.ru/magazine/gp.php?v=list&amp;gp=52005</a>	Горная промышленность
<a href="http://novtex.ru/gormash/">http://novtex.ru/gormash/</a>	Горное оборудование и электромеханика
<a href="http://www.russian-mining.com/">http://www.russian-mining.com/</a>	Russian-mining
<a href="http://glueckauf-rus.rosugol.ru/index.php?option=com_frontpage&amp;Itemid=1">http://glueckauf-rus.rosugol.ru/index.php?option=com_frontpage&amp;Itemid=1</a>	Глюкауф
<a href="http://www.gornoe-delo.ru/magazine/mgp.php">http://www.gornoe-delo.ru/magazine/mgp.php</a>	Мировая горная промышленность
<a href="http://www.miningpage.narod.ru/lib/jurnal/ugol_ua/ugol">http://www.miningpage.narod.ru/lib/jurnal/ugol_ua/ugol</a>	Уголь Украины

В табл. 8.3 приведены ссылки на сайты горных институтов и университетов России и Украины, имеющих горные факультеты.

Таблица 8.3

## Сайты горных институтов и университетов

Ссылка в Интернет	Институт (университет)
<a href="http://www.msmu.ru/">http://www.msmu.ru/</a>	Московский государственный горный университет
<a href="http://www.spmi.ru/index.php">http://www.spmi.ru/index.php</a>	Санкт-Петербургский государственный горный институт имени Г.В. Плеханова
<a href="http://www.gorny-ins.ru/">http://www.gorny-ins.ru/</a>	Воркутинский горный институт
<a href="http://tsu.tula.ru/">http://tsu.tula.ru/</a>	Тульский государственный университет
<a href="http://www.sibsiu.ru/">http://www.sibsiu.ru/</a>	Сибирский государственный индустриальный университет
<a href="http://www.siurgtu.ru/">http://www.siurgtu.ru/</a>	Шахтинский институт ЮРГТУ(НПИ)
<a href="http://library.kuzstu.ru">http://library.kuzstu.ru</a>	Кузбасский государственный технический университет
<a href="http://istu.irk.ru/">http://istu.irk.ru/</a>	Иркутский ГТУ
<a href="http://pstu.ac.ru/">http://pstu.ac.ru/</a>	Пермский ГТУ
<a href="http://www.magtu.ru/">http://www.magtu.ru/</a>	Магнитогорский ГТУ
<a href="http://www.usmga.ru">http://www.usmga.ru</a>	Уральская горно-геологическая академия
<a href="http://www.fentu.ru/">http://www.fentu.ru/</a>	Дальневосточный государственный технический университет
<a href="http://www.rudn.ru/">http://www.rudn.ru/</a>	РУДН
<a href="http://www.ysu.ru/">http://www.ysu.ru/</a>	Якутский гос. университет
<a href="http://www.nmu.org.ua">http://www.nmu.org.ua</a>	Национальный горный университет
<a href="http://donntu.edu.ua">http://donntu.edu.ua</a>	Донецкий национальный технический университет
<a href="http://dgmі.edu.ua/">http://dgmі.edu.ua/</a>	Донбасский государственный технический университет
<a href="http://www.snu.org.ua/index.php">http://www.snu.org.ua/index.php</a>	Восточно-Украинский национальный университет им. Даля
<a href="http://www.ktu.edu.ua">http://www.ktu.edu.ua</a>	Криворожский технический университет

В табл. 8.4 приведены ссылки на сайты научных, проектных и общественных организаций, работающих в области горного дела

Таблица 8.4

## Сайты научных, проектных и общественных организаций

Ссылка в Интернет	Организация
<a href="http://www.ras.ru/">http://www.ras.ru/</a>	Российская академия наук (РАН)
<a href="http://www.ipkonran.ru/">http://www.ipkonran.ru/</a>	ИПКОН РАН
<a href="http://igd.uran.ru/">http://igd.uran.ru/</a>	ИГД Уральское отделение РАН
<a href="http://www.misd.nsc.ru/">http://www.misd.nsc.ru/</a>	ИГД Сибирское отделение РАН
<a href="http://www.igd.khv.ru/">http://www.igd.khv.ru/</a>	ИГД Дальневосточное отделение РАН
<a href="http://www.igdkochinsky.ru/">http://www.igdkochinsky.ru/</a>	ННЦ ГП ИГД им. Скочинского
<a href="http://www.wnimi.ru/index.html">http://www.wnimi.ru/index.html</a>	ВНИМИ
<a href="http://www.viogem-sp.ru/">http://www.viogem-sp.ru/</a>	ФГУП ВИОГЕМ
<a href="http://vniagriugol.webrostop.ru/">http://vniagriugol.webrostop.ru/</a>	ФГУП ВНИГРИУголь
<a href="http://www.agnrf.ru/">http://www.agnrf.ru/</a>	Академия горных наук

<a href="http://www.raen.ru">http://www.raen.ru</a>	Российская академия естественных наук
<a href="http://www.humanistica.ru/maneb/">http://www.humanistica.ru/maneb/</a>	МАНЭБ
<a href="http://www.giproshakt.ru/">http://www.giproshakt.ru/</a>	СпбГипрошахт
<a href="http://www.rostovgiproshaht.com">http://www.rostovgiproshaht.com</a>	Ростовгипрошахт
<a href="http://www.igph.kiev.ua/">http://www.igph.kiev.ua/</a>	Институт геофизики НАН Украины
<a href="http://igtm.narod.ru/">http://igtm.narod.ru/</a>	Институт геотехнической механики НАН Украины
<a href="http://www.niigm.dn.ua/">http://www.niigm.dn.ua/</a>	НИИГМ им. Федорова
<a href="http://ukrnimi.donetsk.ua/">http://ukrnimi.donetsk.ua/</a>	УкрНИМИ
<a href="http://www.lgh.com.ua/rus/index.htm">http://www.lgh.com.ua/rus/index.htm</a>	ГОО «Луганскгипрошахт»
<a href="http://gorprom.riu.ru/">http://gorprom.riu.ru/</a>	Некоммерческая организация «Горнопромышленник и России»
<a href="http://www.ntga.ru/napravlenia.html">http://www.ntga.ru/napravlenia.html</a>	Научно-техническая горная ассоциация
<a href="http://www.dsp.dn.ua/rus.htm">http://www.dsp.dn.ua/rus.htm</a>	ГОО «Грест Донецкшахтопроходка»

В табл. 8.5 приведены ссылки на иноязычные Интернет-ресурсы по горному делу.

Таблица 8.5

**Иноязычные сайты о горной промышленности**

Ссылка в Интернет	Краткая характеристика
<a href="http://www.armorocks.org">http://www.armorocks.org</a>	<b>American Rock Mechanics Association (ARMA)</b> – Страница американской ассоциации механики горных пород.
<a href="http://www.auca.org">http://www.auca.org</a>	<b>American Underground Construction Association (AUA)</b> – Страница американской ассоциации шахтного строительства
<a href="http://www.utrc.org">http://www.utrc.org</a>	<b>Underground Technology Research Council (UTRC)</b> – Сайт исследовательского совета подземной технологии
<a href="http://www.coloradomining.org">http://www.coloradomining.org</a>	<b>Colorado Mining Association (CMA)</b> Сайт ассоциации горного дела Колорадо
<a href="http://www.mining-technology.com">http://www.mining-technology.com</a>	<b>Mining Technology</b> – сайт, посвященный горно-рудной промышленности. Здесь представлены: оборудование, фирмы, компании, новости
<a href="http://www.deutsche-steinkohle.de">http://www.deutsche-steinkohle.de</a>	<b>Deutsche Steinkohle</b> – Сайт дочернего предприятия DSK - Deutsche Steinkohle немецкого концерна RAG - Ruhrkohle AG. На сайте представлены материалы по каменноугольной промышленности Германии
<a href="http://www.untertage.com">http://www.untertage.com</a>	<b>Materialien zur Geologie und zum Bergbau des Schaumburger Landes</b> Страница посвящена геологии и отработке полезных ископаемых в Шаумбурге
<a href="http://www.thyssen-schachtbau.de">http://www.thyssen-schachtbau.de</a>	<b>Thyssen Schachtbau Group</b> Банк данных диссертаций в области вскрышных и подготовительных работ для каменноугольной промышленности Германии.
<a href="http://www.arcs.ac.at">http://www.arcs.ac.at</a>	<b>Aesterreichische Dissertationsdatenbank</b> Банк данных диссертаций. Австрия.

### *Контрольные вопросы и задания*

1. Назовите основные функции Internet.
2. Как в Internet найти необходимую информацию.
3. Какие поисковые системы Вы знаете?
4. Создайте электронный почтовый ящик и напишите электронное письмо.
5. Подготовьте реферат по горной тематике, используя ресурсы Internet, указанные в табл. 8.1 – 8.5.

## **ЗАКЛЮЧЕНИЕ**

---

Круг вопросов, освещенных в настоящей книге, далеко не полностью охватывает возможности информационных технологий, применяемых в горном деле. Основываясь на собственном опыте учебной, научно-исследовательской и производственной деятельности, авторы попытались изложить основы применения электронно-вычислительных средств и современного программного обеспечения на примерах прикладных задач шахтного строительства.

Надеемся, что настоящее пособие поможет студентам самостоятельно изучить основные принципы и приемы разработки баз данных, получить навыки работы с компьютерной графикой, освоить азы объектно-ориентированного программирования, научиться создавать и использовать в инженерной деятельности и научных исследованиях математические модели. Изложенные основы работы в Интернет и подбор полезных ссылок на отечественные и зарубежные сайты, содержащие информацию по горному делу, позволят будущим инженерам широко использовать «всемирную паутину» для пополнения собственных знаний, быстрого и эффективного обмена информацией.

Освоение современных информационных технологий позволит будущим горным инженерам на высоком уровне решать многие производственно-технические, организационные, управленческие и другие задачи, более эффективно использовать рабочее время, постоянно повышать свой профессиональный уровень.

Надеемся, что данное пособие окажется полезным не только студентам горных специальностей вузов, но и инженерно-техническим работникам, начинающим овладение информационными технологиями в своей профессиональной деятельности.



## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

---

1. Булычев Н.С. Механика подземных сооружений. – М.: Недра, 1994. – 382 с.
2. Баклашов И.В., Картозия Б.А. Механика подземных сооружений и конструкции крепей. – М.: Недра, 1992. – 543 с.
3. Программный комплекс «ЛИРА-Windows». Руководство пользователя. – Киев: НИИАСС, 2001.
4. Каретников В.Н., Клейменов В.Б., Нуждихин А.Г. Крепление капитальных и подготовительных горных выработок: Справочник. – М.: Недра, 1989. – 571 с.
5. Коннолли, Томас, Бегг, Каролин. Базы данных. Проектирование, реализация и сопровождение. Теория и практика. 3-е изд. Пер. с англ. – М.: Издательский дом «Вильямс», 2003. – 1440 с.
6. Кузьменко В.Г. Visual Basic 6. Самоучитель. – М.: Бином-пресс, 2002. – 400 с.
7. Шахтное и подземное строительство. Решение практических задач на ЭВМ/ А.Ю. Прокопов, И.А. Мартыненко, С.Г. Страданченко и др. – Новочеркасск: ЮРГТУ, 2000. – 172 с.
8. Зельднер Г.А. QuickBASIC для носорога. – М.: АБФ, 1994. – 480 с.
9. Семенов Ю.А. Протоколы Интернет. Энциклопедия / Ю.А. Семенов. – М.: Горячая линия-Телеком, 2001. – 1100 с.
10. Коэн К., Дэниэлс Э. Сети под управлением Windows XP/ Каки Коэн, Эндру Дэниэлс; пер. с англ. – М.: ИТ Пресс, 2005. – 504 с.
11. Семенов Ю.А. Сети Интернет. Архитектура и протоколы/ Ю.А.Семенов. – М.: СИРИНЬ, 1998. – 543 с.
12. Райордан Р. Основы реляционных баз данных. Пер. с англ. – М.: Издательско-торговый дом «Русская Редакция», 2001. – 384 с.
13. Гарсиа-Молина, Гектор, Ульман, Джеффри, Д., Уидом, Дженнифер. Системы баз данных. Полный курс. Пер. с англ. – М.: Издательский дом «Вильямс», 2003. – 1088 с.
14. Грабер М. Введение в SQL. – М.: Лори, 1997. – 548 с.
15. Базы данных: модели, разработка, реализация / Т.С. Карпова. – СПб.: Питер, 2001. – 304 с.
16. Порев В.Н. Компьютерная графика. – СПб.: БХВ-Петербург, 2002. – 432 с.
17. Сдвижков О.А. Математика в Excel 2002. – М.: СОЛОН-Пресс, 2004. – 192 с.
18. Ткачев Д.А. AutoCAD 2005. Самоучитель. – СПб.: Питер; Киев: Издательская группа BHV, 2005. – 462 с.
19. Соколова Т. AutoCAD 2005 для студента. Популярный самоучитель. – СПб.: Питер, 2005. – 320 с.

Юрий Николаевич Попков  
Альберт Юрьевич Прокопов  
Марина Валентиновна Прокопова

Учебное издание

**Информационные технологии  
в горном деле**

---

Редактор Н.А. Юшко

Темплан 2007 г. Подписано в печать 1.03.07. Формат 60x84<sup>1</sup>/<sub>16</sub>. Бумага офсетная. Печать оперативная.  
Печ.л. . Уч.-изд. л. . Тираж 200.

---

Южно-Российский государственный технический университет

Шахтинский институт (филиал)

Учебно-методический отдел

Адрес института: 346500, г. Шахты, пл. Ленина