

С. Т. Мухамбетжанова, А. С. Тен, Д. Н. Исабаева, В. В. Сербин

Информатика

Учебник для 7 классов общеобразовательных школ

7

Рекомендовано Министерством образования и науки Республики Казахстан



Алматы «Атамұра» 2017

УДК 373.167.1
ББК 32.81я72
М 92

Учебник подготовлен в соответствии с Типовой учебной программой по предмету «Информатика» для 5–9 классов уровня основного среднего образования по обновленному содержанию, утвержденной Министерством образования и науки РК

Условные обозначения:



– Важная информация



– Вопросы



– Практическое задание



– Творческое задание



– Дополнительный материал



– Индивидуальная работа



– Работа в паре



– Работа в группе



– Работа в классе

Мухамбетжанова С. Т.

М92 Информатика: Учебник для 7 кл. общеобразоват. шк./ С. Т. Мухамбетжанова, А. С. Тен, Д. Н. Исабаева, В. В. Сербин. – Алматы: Атамұра, 2017. – 176 с.

ISBN 978-601-306-745-2

УДК 373.167.1
ББК 32.81я72

ISBN 978-601-306-745-2

© Мухамбетжанова С. Т., Тен А. С.,
Исабаева Д. Н., Сербин В. В., 2017
© «Атамұра», 2017

СОДЕРЖАНИЕ

От авторов	4
Раздел 1. ИЗМЕРЕНИЕ ИНФОРМАЦИИ И КОМПЬЮТЕРНАЯ ПАМЯТЬ	
1.1. Единицы измерения информации.....	5
1.2. Перевод из одних единиц измерения информации в другие.....	8
1.3. Компьютерная память.....	10
1.4. Размеры файлов	18
1.5. Создание и распаковка архивов различных форматов.....	24
Раздел 2. СЕТИ И ИХ БЕЗОПАСНОСТЬ	
2.1. Компьютерные сети и их классификация.....	30
2.2. Антивирусная защита	39
Раздел 3. РЕШЕНИЕ ЗАДАЧ С ПОМОЩЬЮ ЭЛЕКТРОННЫХ ТАБЛИЦ	
3.1. Таблицы в текстовом процессоре.....	49
3.2. Форматирование элементов электронных таблиц	56
3.3. Форматы типов данных.....	66
3.4. Условное форматирование в электронной таблице.....	74
3.5. Графическое представление табличных данных	80
3.6. Моделирование процессов в электронных таблицах.....	87
Раздел 4. ПРОГРАММИРОВАНИЕ РЕШЕНИЙ	
4.1. Языки программирования	98
4.2. Системы программирования	103
4.3. Типы данных	106
4.4. Интерфейс проекта	110
4.5. Программирование линейных алгоритмов	115
4.6. Программирование алгоритмов ветвления	123
4.7. Программирование вложенных условий.....	127
4.8. Программирование составных условий	134
Раздел 5. МОДЕЛИРОВАНИЕ ОБЪЕКТОВ И СОБЫТИЙ	
5.1. Трехмерные модели	142
5.2. Объекты, встроенные в редактор	147
5.3. Трехмерные модели объектов.....	155
5.4. Трехмерные модели событий.....	163
Глоссарий	171

Дорогие семиклассники!

Сейчас, когда компьютеры, мобильные технологии, Интернет стали неотъемлемой частью современной жизни, школьный предмет «Информатика» становится проводником в мир информационных технологий, информационной культуры.

Материал, изучаемый на уроках информатики, поможет вам расширить кругозор и лучше понимать те процессы, которые связаны с передачей, приемом, преобразованием и хранением информации.

Учебник состоит из **пяти разделов**. Изучив **первый раздел**, вы будете знать, в каких единицах измеряется информация, как определять размер файла и какие виды компьютерной памяти существуют для компьютеров.

Второй раздел научит вас классифицировать компьютерные сети и защищать компьютер от вредоносных программ.

Мощные возможности электронных таблиц, изученных в **третьем разделе**, помогут вам при обработке результатов экспериментов, самостоятельных исследований, при моделировании процессов и при математическом обосновании разрабатываемых проектов.

Четвертый раздел – программирование, самый сложный и самый интересный. Программирование – это часть цифровой грамотности (digital literacy), которой можно обучить любого человека, так же как читать и писать. Программирование будет формировать особый структурный стиль мышления, помогать во всех областях жизни.

Пятый раздел – моделирование объектов и событий. Этот раздел познакомит вас с возможностями 3D-графики. Сведения и навыки, полученные в ходе изучения трехмерной графики, могут быть востребованы в вашей будущей практической деятельности.

В конце тем предложены вопросы, задания, практические и творческие задания, исследовательские работы, составленные по уровням учебных целей таксономии Б. Блума.

Все разделы учебника позволят обеспечить вашу общую уверенность в востребованности ваших знаний в будущем и подготовят вас для продолжения образования в современном высокотехнологичном мире.

Желаем успехов!

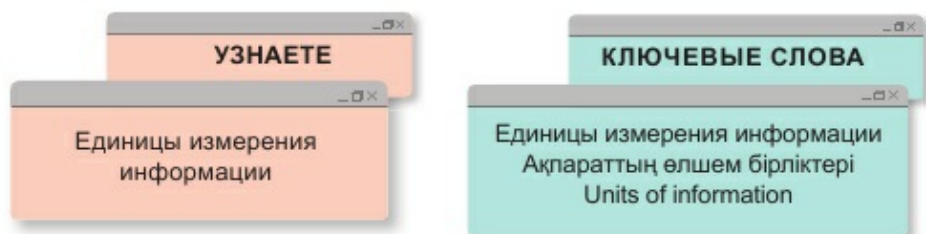
Ваши авторы



Раздел 1

ИЗМЕРЕНИЕ ИНФОРМАЦИИ
И КОМПЬЮТЕРНАЯ ПАМЯТЬ

1.1. Единицы измерения информации



С понятием «информация» вы познакомились в пятом классе. Знания, которые человек получает из различных источников, являются для него информацией. Информация может быть **информативной** и **неинформативной**. Информативная информация содержит ненулевую информацию и дополняет знания человека. Например, прогноз погоды на завтра будет информативным сообщением, а вчерашний прогноз погоды – неинформативным, так как он уже известен. В данном примере две ситуации: «есть информация» или «нет информации». Иными словами, количество информации может быть **равно нулю** или **не равно нулю**.

Для количественного определения любой величины необходимо определить единицу измерения. Например, для измерения длины в качестве единицы измерения выбран метр, массы – килограмм и т. д.

ЗАПОМНИТЕ

За единицу количества информации принят **1 бит** – количество информации, содержащееся в сообщении, уменьшающем неопределенность знаний в два раза.

Неопределенность знаний некоторого события – это количество возможных результатов события.

Например, при бросании монеты реализуется из двух возможных событий одно. Следовательно, полученное количество информации равно 1 бит.



ПОДУМАЙТЕ И ОТВЕЬТЕ

Сколько событий выпадает при бросании кубика из шести граней?
Приведите свои примеры.

На практике применяется единица измерения информации – **байт**.

1 байт информации может содержать 256 различных кодов, что вполне достаточно для кодирования одного знака текста. При кодировании числовой, звуковой или графической информации может использоваться большее количество байт. У единиц измерения информации, как и единиц измерения веса и длины, существуют более крупные единицы (рис. 1.1).

Принята следующая система единиц измерения количества информации:

$$\begin{aligned} 1 \text{ Гигабайт (Гбайт)} &= 1024 \text{ Мбайт} = 2^{10} \text{ Мбайт} = 2^{20} \text{ Кбайт} = 2^{30} \text{ байт} \\ 1 \text{ Терабайт (Тбайт)} &= 1024 \text{ Гбайт} = 2^{10} \text{ Гбайт} = 2^{20} \text{ Мбайт} = \\ &= 2^{30} \text{ Кбайт} = 2^{40} \text{ байт} \end{aligned}$$

Каждая следующая больше предыдущей в 1024 (2^{10}) раза.

ВЫПОЛНИТЕ

Переведите 1 петабайт, 1 эксабайт, 1 зеттабайт, 1 йоттабайт в байты.

$$\begin{aligned} 1 \text{ Петабайт (Пбайт)} &= 1024 \text{ Тбайт} \\ 1 \text{ Эксабайт (Эбайт)} &= 1024 \text{ Пбайт} \\ 1 \text{ Зеттабайт (Збайт)} &= 1024 \text{ Эбайт} \\ 1 \text{ Йоттабайт (Йбайт)} &= 1024 \text{ Збайт} \end{aligned}$$

Рис. 1.1. Единицы измерения информации

С помощью одного байта можно записать двоичные коды 256 (2^8) чисел от 0 до 255.

Измерим количество информации в следующем предложении: «*Сабит Муканов – классик казахской литературы, поэт, общественный деятель, академик, председа-*



тель Союза писателей Казахстана». Любой символ текста занимает 1 байт. Текст содержит 105 букв, 6 знаков препинания, 13 пробелов. Значит, для обработки этого предложения нужно 124 байт.

ВЫПОЛНИТЕ

Книга содержит 100 страниц; на каждой странице 35 строк, в каждой строке 50 символов. Объем информации, содержащийся в книге, рассчитывается следующим образом: страница содержит $35 \times 50 = 1750$ байт информации. Какой объем всей информации в книге (в разных единицах измерения информации: Кбайт, Мбайт)?

Уровень – Знание и понимание



1. Для чего нужны единицы измерения информации?
2. Что принято за единицу измерения количества информации?
3. Что такое *байт*, *килобайт*, *мегабайт* и *гигабайт*?
4. Как они связаны между собой?
5. Расположите единицы измерения информации в порядке возрастания: Тбайт; Кбайт; байт; Мбайт; Гбайт.

Уровень – Применение



Какое количество вопросов достаточно задать вашему собеседнику, чтобы наверняка определить месяц, в котором он родился?

Уровень – Анализ

Проанализируйте информацию, поставив вместо вопросительных знаков числа.

- а) 1 байт = ? бит = 2^3 бит;
- б) 213 бит = ? Кбайт;
- в) 1 Мбайт = ? Кбайт = ? байт = ? бит;
- г) ? Кбайт = ? байт = 12288 бит.

Уровень – Синтез

Замените вопросительные знаки на пропущенные числа.

- а) 1 Мбайт = ? Кбайт = ? байт;
- б) 1 Гбайт = ? Мбайт = ? Кбайт;
- в) 2 Тбайт = ? Гбайт = ? Мбайт;
- г) ? Гбайт = 1536 Мбайт = ? Кбайт.

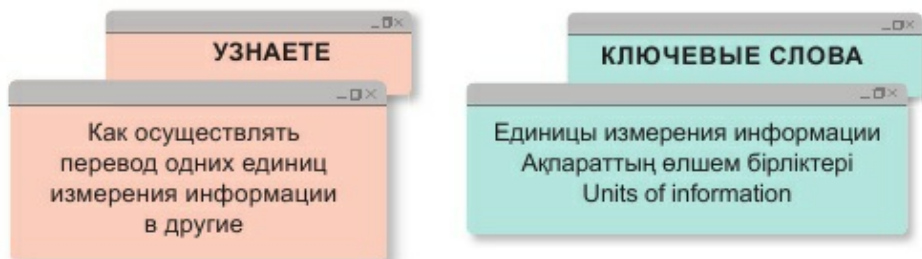
Уровень – Оценивание



Подготовьте сообщение «Большие единицы измерения информации». Дайте оценку применения каждой единице измерения информации по степени возрастания.



1.2. Перевод из одних единиц измерения информации в другие



Рассмотрим несколько примеров для перевода из одних единиц измерения информации в другие.

ОПРЕДЕЛИТЕ

Подсчитайте информационный объем сообщения «ИНФОРМАЦИЯ» в битах. Если 1 символ равен одному байту, то чему равно количество байтов в сообщении ИНФОРМАЦИЯ?

Для перевода из одних единиц измерения информации в другие используйте следующую схему (рис. 1.2):



Рис. 1.2. Схема перевода из одних единиц измерения в другие

СФОРМУЛИРУЙТЕ

Сформулируйте правило перехода от крупных единиц к более мелким (от байтов к битам) и наоборот.

ВЫПОЛНИМ ВМЕСТЕ

1. Рассчитайте объем информации, требуемый для хранения рисунка, который состоит из 1200 цветных точек по горизонтали и 600 точек по вертикали, а каждая цветная точка кодируется одним байтом, т. е. может иметь один из 256 различных цветов.



Сначала выясним, сколько всего точек содержит рисунок. Для этого выполним:

$$1200 \times 600 = 720\,000 \text{ точек} = 720\,000 \text{ байт.}$$

Переведем байты в килобайты:

$$720\,000 : 1024 = 703,125 \text{ Кбайт.}$$

Ответ: рисунок требует для хранения 703,125 Кбайт.

2. Выясните, сколько килобайт информации займет книга, которая имеет 200 страниц, по 60 строк на странице и 80 символов в строке.

Определим, сколько символов размещается на одной странице:

$$60 \times 80 = 4800 \text{ символов.}$$

Определим, сколько символов (байт) в книге:

$$4800 \times 200 = 960\,000 \text{ символов} = 960\,000 \text{ байт.}$$

Переведем байты в килобайты:

$$960\,000 : 1024 = 973,5 \text{ Кбайт.}$$

Ответ: книга займет 973,5 Кбайт информации.

Уровень – Знание и понимание



1. Как происходит перевод крупных единиц измерения информации в мелкие? Приведите пример.
2. Как происходит перевод мелких единиц в крупные? Приведите пример.
3. Определите информационный объем сообщений «Компьютер» в битах.
4. Сколько мегабайт в одном гигабайте?
5. Объясните схему перевода одних единиц измерения в другие (рис. 1.2).

Уровень – Применение



1. Переведите одну единицу измерения в другие и заполните ячейки табл. 1.1:

Таблица 1.1

1 байт	=		бит
128 бит	=		байт
1 Кбайт	=		бит
1,5 Кбайт	=		байт
2048 байт	=		Кбайт
2 Мбайт	=		Кбайт
81920 бит	=		Кбайт
1,5 Мбайт	=		байт



2. Каждая страница учебника содержит 40 строк, в каждой строке 60 символов. Учебник состоит из 176 страниц. Определите информационный объем учебника.



Уровень – Анализ



Определите объем информации для хранения имен и телефонов всех своих друзей в смартфоне.



Уровень – Синтез

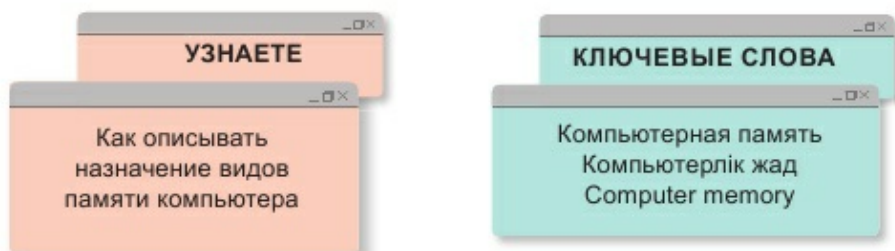
Сколько останется места на флеш-накопителе с емкостью 2 Гбайт, если на нее записать книгу, в которой 256 страниц по 95 строк и 80 символов в строке?



Уровень – Оценивание

Размер фотографии 4000×1500 точек. Каждая точка имеет столько цветов, что для ее кодирования требуется 2 байта. Сколько места займет данная фотография?

1.3. Компьютерная память



Современные компьютеры имеют много разнообразных запоминающих устройств. Эти устройства отличаются между собой по назначению, временным характеристикам, объему хранимой информации и др.



ЗАПОМНИТЕ

Компьютерная память – это устройство хранения информации, или запоминающее устройство. Память является обязательной частью компьютера.

Компьютерную память можно разделить на *внутреннюю* и *внешнюю*.

Внешняя память предназначена для долговременного хранения информации пользователя, а внутренняя используется компьютерной системой.

ЗАПОМНИТЕ

Оперативная память (*Random Access Memory – RAM*) – это быстрое запоминающее устройство не очень большого объема, связанное с процессором и предназначенное для записи, считывания и хранения выполняемых программ и данных, обрабатываемых этими программами (рис. 1.3, 1.4).

После загрузки компьютера в оперативную память загружается информация. Информация хранится до тех пор, пока работает компьютер. После выключения компьютера вся информация из памяти стирается.



Рис. 1.3. Оперативная память для настольного компьютера



Рис. 1.4. Оперативная память для ноутбука



ЗАПОМНИТЕ

Кэш-память, или сверхоперативная память, – очень быстрое запоминающее устройство небольшого объема. Она используется при обмене данными между микропроцессором и оперативной памятью.

Кэш-память бывает двух видов: *первого уровня*, которая располагается на процессоре, и *второго уровня*, которая является отдельной высокоскоростной микросхемой статической памяти (SRAM) (рис. 1.5, 1.6).



Рис. 1.5. Процессор и кэш-память



Рис. 1.6. Кэш-память в виде отдельной высокоскоростной микросхемы

Обмен между данными происходит по следующей схеме: когда процессору требуются данные, он сначала ищет их в кэш-памяти первого уровня. Если там нет данных, обращение идет к кэш-памяти второго уровня. Если процессор не находит данных в кэш-памяти, он обращается к оперативной памяти, поиск в которой занимает намного больше времени, чем в кэш-памяти.

Кэш-память используется для увеличения быстродействия компьютера в целом.



CMOS-память (рис. 1.7) предназначена для длительного хранения данных о конфигурации и настройке компьютера (дата, время, пароль). Для этого используют специальные электронные схемы со средним быстродействием, установленные на материнской плате.



Рис. 1.7. CMOS-память

ЗАПОМНИТЕ



Постоянная память (ПЗУ, англ. ROM, *Read Only Memory* – память только для чтения) – энергонезависимая память, используется для хранения данных, которые никогда не требуют изменения.

Прежде всего в постоянную память (рис. 1.8) записывают программу управления работой самого процессора, программы управления дисплеем, клавиатурой, принтером, внешней памятью, программы запуска и остановки компьютера, тестирования устройств.



Рис. 1.8. Постоянная память





Видеопамять (VRAM) – это разновидность оперативного запоминающего устройства, в котором хранятся закодированные изображения.



Рис. 1.9. Видеопамять

Для хранения графической информации используется **видеопамять** (рис. 1.9).

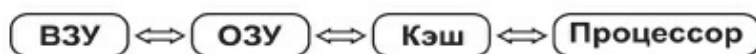
Это запоминающее устройство организовано так, что его содержимое доступно сразу двум устройствам – процессору и дисплею. Поэтому изображение на экране меняется одновременно с обновлением видеоданных в памяти.

ЗАПОМНИТЕ



Внешняя память (ВЗУ) предназначена для длительного хранения программ и данных. Целостность ее содержимого не зависит от того, включен или выключен компьютер, и не имеет прямой связи с процессором.

Информация от ВЗУ к процессору, и наоборот, циркулирует примерно по следующей цепочке:



К устройствам внешней памяти относятся: накопитель на жестких магнитных дисках (НЖМД или HDD), гибких магнитных дисках (НГМД), магнитных лентах (стримеры), оптических дисках (CD-ROM), магнитооптических дисках, USB-флеш (рис. 1.10).





Рис. 1.10. Накопители

Устройства, относящиеся к внешней памяти компьютера, постоянно модифицируются. Их производительность, объем, размер и внешний вид улучшаются в соответствии с современными требованиями.

Уровень – Знание и понимание



1. Объясните, как вы понимаете, что такое «компьютерная память».
2. Для чего необходима внутренняя, внешняя память?
3. Сравните внутреннюю и внешнюю память.
4. Что такое *кэш-память*?
5. Где хранятся закодированные изображения?
6. Для чего предназначена постоянная память?

Уровень – Применение

1. Установите соответствие между назначением и видами памяти в табл. 1.2.



Таблица 1.2

№	Назначение памяти	№	Вид памяти
I	Хранит информацию о начальной загрузке и настройках компьютера	A	Оперативное запоминающее устройство
II	Промежуточное запоминающее устройство для обмена информацией между ОЗУ и процессором	B	Внешний жесткий диск
III	Видеопамять	C	Постоянное запоминающее устройство
IV	Хранит информацию в текущий момент времени, пока включен компьютер	D	Внешний лазерный (оптический) диск
V	Хранит большие объемы информации после выключения компьютера	E	Внутренняя КЭШ-память
VI	Хранит большие объемы информации, служит для переноса ее на другие компьютеры	F	Оперативное запоминающее устройство, в котором хранятся закодированные изображения

I	II	III	IV	V	VI

2. Выполните на компьютере:

- Откройте «Диспетчер устройств».
- Отметьте устройства, подключенные к компьютеру.
- Вставьте флеш-накопитель.
- Что изменилось в «Диспетчере устройств»?

3. Заполните табл. 1.3 недостающей информацией.

Таблица 1.3

Функция	Человек	Компьютер
Хранение информации		
Обработка информации		
Прием информации		
Передача информации	Речь, двигательная система	



Уровень – Анализ

Составьте кластер по видам компьютерной памяти.
 Пояснение приема критического мышления «кластер».
 Кластер — это графическая форма организации информации, когда выделяются основные смысловые единицы, которые фиксируются в виде схемы с обозначением всех связей между ними. Он представляет собой изображение, способствующее систематизации и обобщению учебного материала (рис. 1.11).



Рис. 1.11. Кластер

Уровень – Синтез и оценивание

Разгадайте кроссворд (рис. 1.12).

						10						
1												
					2							
	3											
		4										
					5							
				6								
		7										
				8								
			9									

Рис. 1.12. Кроссворд «Память компьютера»

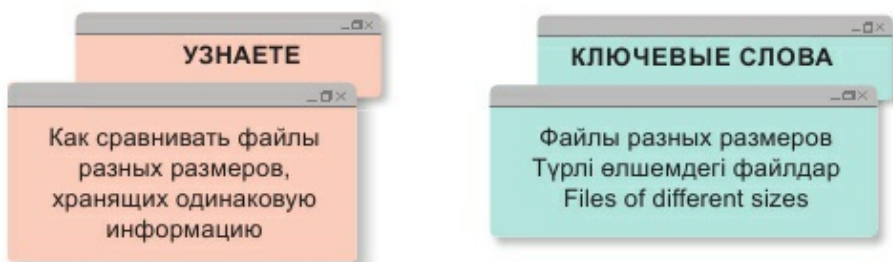


По горизонтали:

1. Внутренняя память, которая хранит информацию временно, пока компьютер включен.
2. Гибкий магнитный диск.
3. Внутренняя память, которая хранит информацию для начальной загрузки компьютера.
4. Вид памяти, которая может хранить большие объемы информации. Принцип чтения/записи информации – луч лазера.
5. Один из магнитных дисков, хранит все программы и данные на вашем компьютере.
6. Устройство для чтения и записи информации на диск.
7. Устройство для хранения информации на компьютере.
8. Один из видов внешней памяти, предназначенный для длительного хранения и переноса на другой компьютер больших объемов информации.
9. Другое название компакт-диска (оптического диска).

По вертикали:

10. Другое название жесткого магнитного диска.

1.4. Размеры файлов

Часто при работе с разными файлами с одинаковым содержанием необходимо сравнить их и выбрать тот файл, который будет удовлетворять вашим требованиям. Например, документ, создан-



ный в текстовом процессоре Word 97-2003, имеет расширение **.doc**, а документ в текстовом процессоре Word 2007 и выше версий имеет расширение **.docx** (рис. 1.13).

Сегодня актуальны файлы с именем и расширением **.docx**. Количество файлов с расширением имени **.doc**, ввиду постепенного обновления программного обеспечения первых, становится все меньше.



Рис. 1.13. Расширение **.doc** и **.docx**

ИССЛЕДУЙТЕ

Чтобы узнать разницу между двумя текстовыми файлами с расширением **.docx** и **.doc**, проведем мини-исследование. Для этого сравним их, применяя команду **Свойства** из контекстного меню, нажав правую кнопку мыши по выбранному файлу (рис. 1.14).

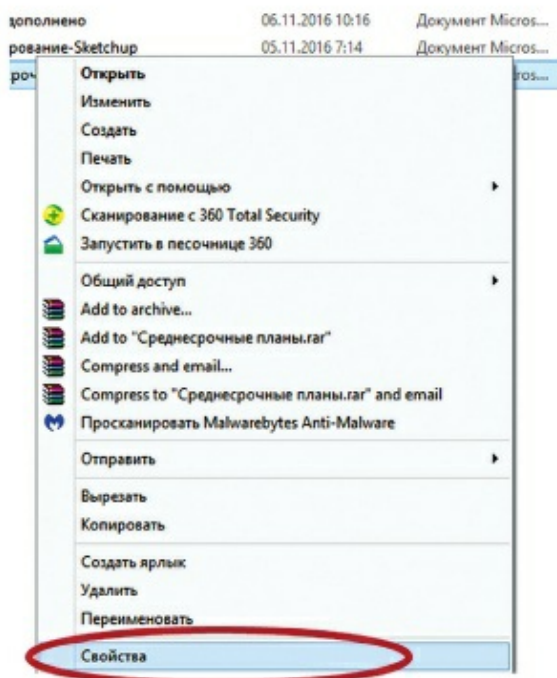


Рис. 1.14. Свойства файла



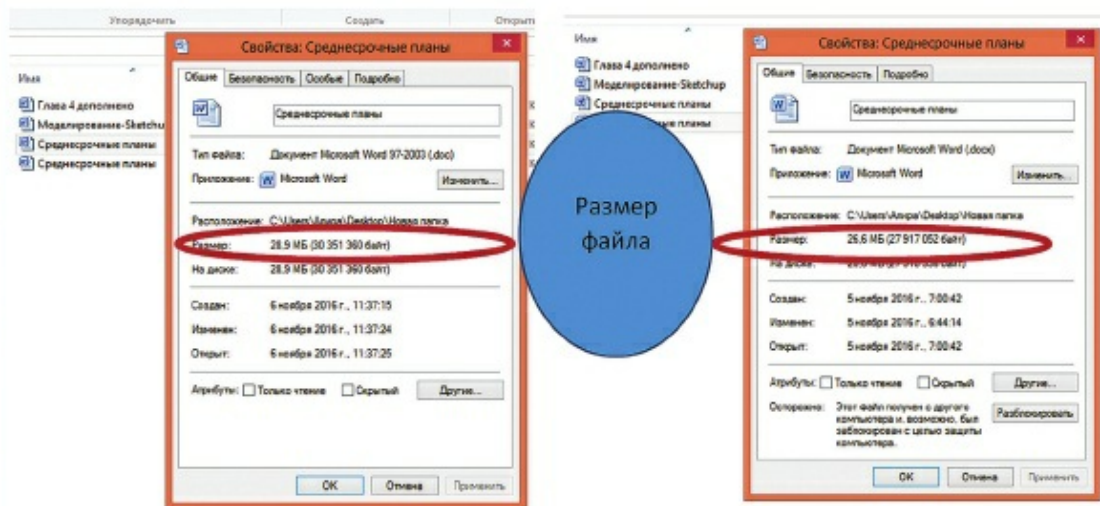


Рис. 1.15. Сравнение текстовых файлов .doc и .docx

Из рис. 1.15 мы видим, что размеры файлов немного отличаются. Файл, созданный с расширением .doc, имеет размер 28,9 Мбайт, а документ с расширением .docx занимает 26,6 Мбайт. Мы видим, что документы, имеющие одинаковое содержание, отличаются размером файла. И разница между ними составляет 2,6 Мбайт.

Данное сравнение хорошо иллюстрирует, сколько места на диске занимает одна и та же информация в зависимости от того, в каком формате она записана.

Рассмотрим следующий пример, сравнив два графических файла с одинаковым изображением, но с разным расширением, одинакового содержания, но разных форматов: TIFF и JPEG (рис. 1.16).

Разберем первый файл в формате TIFF. Он очень популярен для хранения изображений. Данный формат позволяет сохранять фотографии в различных цветовых пространствах и с большой глубиной цвета (8, 16, 32 и 64 бит). TIFF широко поддерживается графическими приложениями и используется в полиграфии. Изображение в TIFF не будет терять качество после каждого сохранения файла. Но, к сожалению, именно из-за этого TIFF-файлы имеют большой размер. В данном примере он составляет 12,5 Мбайт.

Второй файл с расширением JPG – это самый распространенный и популярный формат графических файлов. На практике сохранение



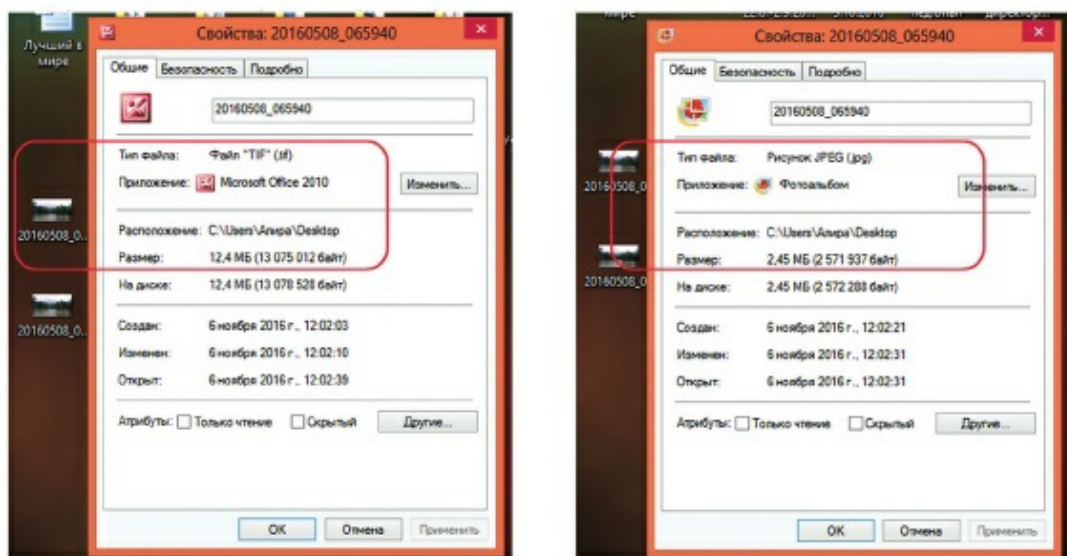


Рис. 1.16. Сравнение графических файлов

фотографии с минимальной степенью сжатия не дает видимого ухудшения качества изображения. На рис. 1.15 это же изображение имеет размер файла 2,45 Мбайт. Разница при сравнении размеров двух файлов составляет 10,5 Мбайт.

С самыми разными **видеоформатами** мы сталкиваемся постоянно. Сегодня существует множество форматов **видеофайлов**. Существующие видеоформаты обладают уникальными особенностями. Возникает вопрос: какой формат видео лучше? Какому видеоформату отдать предпочтение? Рассмотрим классификацию **некоторых форматов видеофайлов** на рис. 1.17.

Существует множество различных **аудиоформатов** (звуковой файл). Наиболее часто используются такие форматы, как **MP3** и **WAV**. Тип формата обычно определяется расширением файла **.mp3**, **.wav**, **.ogg**, **.wma**.

WAV – один из первых аудиоформатов. Обычно используется для хранения несжатых аудиозаписей. В среднем одна минута звука в формате **WAV** занимает около 10 Мбайт.

MP3 (**MPEG Layer-3**) – наиболее распространенный в мире звуковой формат. На текущий момент **MP3** не является лучшим форматом по соотношению размера файла с качеством звучания, но поддерживается большинством устройств.



Формат 3GP, расширение файла .3gp – формат видео из мобильного контента, которое понимается не всеми плеерами и предназначено для мобильных экранов. Качество очень низкое, а размер очень маленький.



Формат MOV, расширение файла .mov – видео, предназначенное для плеера QuickTime от Apple. При выборе файла надо обращать внимание на другие его характеристики: размер, разрешение и т. д.



Формат AVI, расширение файла .avi – самый популярный формат. Он может содержать как видео, так и аудиотекст. Изображение и звук воспроизводятся синхронно.



Формат MPEG-4, расширение файла .mp4 – самый распространенный формат для большинства современных видеоустройств. В этот контейнер входят все популярные видеокодеки. Главное – качество изображений теряется минимально в процессе сжатия.

Рис. 1.17. Классификация некоторых форматов видеофайлов

Для сжатия данных в аудиоформат применяется определенный алгоритм кодирования – **кодек**. Например, в формате **MP3** всегда используется кодек MPEG Layer-3, а в формате **MP4** могут быть использованы разные кодеки.

При сжатии **MP3** звук теряет качество, тем самым уменьшается размер файла.

При хранении и отправке файлов разных форматов, но с одинаковым содержанием предпочтение чаще отдается файлу с меньшим размером.



Уровень – Знание и понимание



1. Каким образом можно сравнить файлы с одинаковым содержанием, но разных форматов?
2. В чем преимущество расширений текстовых документов **.doc** от **.docx**, **.txt** от **.rtf**?
3. Чем отличаются графические файлы с расширениями **.tiff** от **.jpg**, **.bmp** от **.jpg**?
4. Как можно поменять расширение файла для уменьшения его размера?
5. Какие форматы видеофайлов наиболее распространены? Приведите примеры.
6. В чем разница между аудиоформатами **MP3** и **WAV**?



Уровень – Применение

Создайте два текстовых документа: в Word с расширением .doc и Блокноте с расширением .txt.

- Файлы сохраните с одинаковым именем «Алгоритм сравнения».
- Сравните два файла, открыв документ, созданный в текстовом процессоре Word и в Блокноте.

Уровень – Анализ

В пустые ячейки табл. 1.4 напротив форматов (расширений) информации впишите их форму представления: текстовый, звуковой, графический, видеoinформация или аудиoinформация.

Таблица 1.4

										
1	2	3	4	5	6	7	8	9	10	11

Уровень – Синтез

Проведите исследование: какие текстовые, графические, аудио- и видеоформаты часто используются в вашей практике?

Какие звуковые и видеоформаты поддерживает программное обеспечение вашего телефона?

Уровень – Оценивание

1. Если хранить одну тысячу картин с расширением .jpg (приблизительный размер одной картины – 5 Мбайт), то сколько места на жестком диске они займут?
2. Почему компьютерную графику используют почти всегда в сжатом виде?



1.5. Создание и распаковка архивов различных форматов



Проблема сжатия данных существует давно, с момента появления компьютеров. **Сжатием** (уменьшением размера) данных называется преобразование данных с целью уменьшения занимаемого ими объема. Сжатие данных используется при работе с файлами больших форматов. Степень сжатия в основном зависит от исходных размеров файлов. Хорошо сжимаются почти все предварительно не сжатые данные, например, исполняемые файлы, тексты, простые несжатые изображения. В настоящее время сжимать данные необходимо в двух случаях:

- 1) для длительного хранения данных на различных носителях информации;
- 2) для передачи данных по каналам связи.

При сжатии данных получаются файлы особого типа, называемые **архивами**.

ЗАПОМНИТЕ



Архив – это файл, содержащий в себе один или несколько файлов в сжатом виде.

Расширения файлов, являющихся архивами, могут быть zip, arj, rar и др. Тип расширения зависит от программы архиватора, в которой они были созданы.

Программы-архиваторы – это специальные программы, с помощью которых можно сжимать отдельные файлы или группы файлов.

ЗАПОМНИТЕ



Процесс сжатия файлов называется **созданием архива**. Процесс восстановления сжатых файлов – **распаковкой архива**.



С помощью программ-архиваторов WinRar, WinZip, WinArj можно не только сжимать файлы, но и просматривать содержание архивов, распаковывать, устанавливать защиту.

Наиболее популярным архиватором является программа WinRar, которая имеет три главных преимущества:

- поддержка разных платформ операционной системы;
- работа с разными типами архивов, в том числе запакованными другими архиваторами;
- высокая степень сжатия архива.

ВЫПОЛНИМ ВМЕСТЕ

Существуют два способа создания архива (распаковки) данных в программе архиватора WinRar: использование окна программы или контекстного меню. Используем простейший способ создания (распаковки) файлов через контекстное меню (рис. 1.18).



Рис. 1.18. Добавить в архив

Создадим архив по алгоритму:

1. Выберите файл (для выделения группы файлов нажмите клавишу Ctrl и щелкните левой кнопкой мыши – ЛКМ) или папку.
2. Нажмите правой кнопкой мыши (ПКМ) по выделенным файлам, появится контекстное меню (или нажмите на клавиши Shift + F10).
3. Выберите в контекстном меню команду **Add to archive...** (Добавить в архив...).



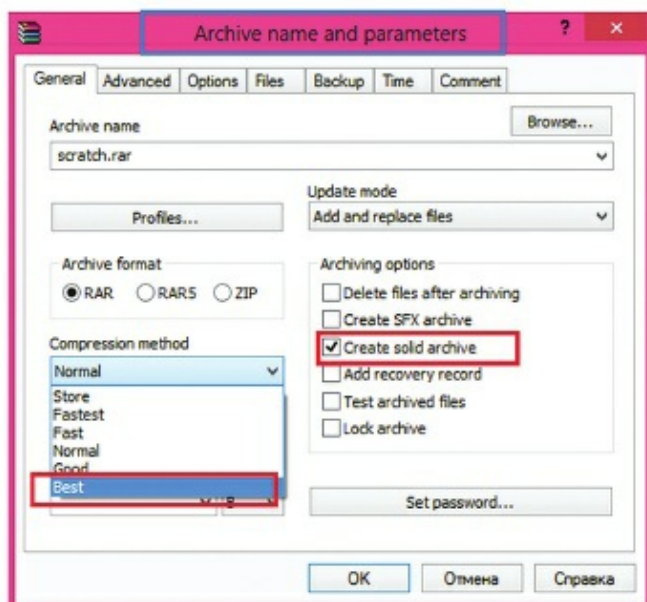


Рис. 1.19. Имя и параметры архива

4. В диалоговом окне **Archive name and parameters (Имя и параметры архива)** нажмите на кнопку **Browse...** (Обзор...) (рис. 1.19).

5. В поле **Archive name (Имя архива)** задайте имя архива или оставьте то имя, которое будет отображено в поле по умолчанию (по названию файла или папки).

6. Выберите параметр архивации **Create solid archive (Создать непрерывный архив)** из группы **Archiving options (Параметры архивации)** на рис. 1.19.

Выберите из списка **Compression method (Метод сжатия)** **Best – Лучший**.

7. Нажмите кнопку **Ок**.

8. Будет создан один непрерывный файл с расширением **.rar**.

ПОЭКСПЕРИМЕНТИРУЙТЕ

1. Попробуйте самостоятельно распаковать на любом компьютере файлы из архива (рис. 1.20).

2. Обсудите алгоритм распаковки.

3. Создайте самораспаковывающийся архив.

4. В чем преимущество такого архива?



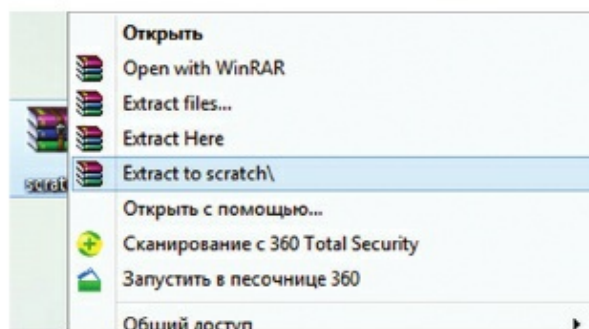


Рис. 1.20. Распаковка архива

Уровень – Знание и понимание



1. Объясните своими словами значение терминов «создание архива» и «распаковка».
2. Какие программы-архиваторы вы еще знаете?
3. Как вы думаете, всегда ли необходимо создавать архив?
4. В чем преимущество архива?
5. Какие файлы будут максимально сжаты: текстовые или графические?

Уровень – Применение



1. Создайте папку **Заготовки** с текстовыми, графическими, звуковыми и видеофайлами.
2. Создайте рабочую папку под именем **Архивы**.
3. В папке **Архивы** создайте четыре подпапки с именами: **Документы**, **Графика**, **Музыка** и **Видео**.
4. В текстовом процессоре Word создайте таблицу.
5. Впишите в нее названия файлов и их размер из папки **Заготовки**.

Таблица 1.5

Текстовые файлы		Графические файлы		Звуковые файлы		Видеофайлы	
Название	Размер	Название	Размер	Название	Размер	Название	Размер
Итого		Итого		Итого		Итого	
Документы.rar		Графика.rar		Музыка.rar		Видео.rar	
Размер		Размер		Размер		Размер	



6. С помощью архиватора WinRar создайте архивы соответствующих файлов из папки «Заготовки» в архивы с именами: «Документы.rar», «Графика.rar», «Музыка.rar», «Видео.rar». Создаваемые архивы поместите соответственно в папки: «Документы», «Графика», «Музыка» и «Видео».

Уровень – Анализ

1. Выпишите размер созданных архивов в таблицу.
2. Сделайте вывод: какие файлы лучше всего сжимаются?
3. Распакуйте архивы «Документы.rar», «Графика.rar», «Музыка.rar», «Видео.rar» в папки «Документы», «Графика», «Музыка», «Видео».
4. Создайте в рабочей папке самораспаковывающийся архив (SFX-архив) с именем **Самораспаковка.exe**, в который запакуйте все файлы из папки «Заготовки».
5. Извлеките архив **Самораспаковка.exe**, дважды нажав левой кнопкой мыши.
6. Проверка архивов на вирусы:
 - Откройте программу WinRar.
 - С помощью комбинации клавиш **Alt+D** выполните проверку архивных файлов на вирусы.
7. Создайте многотомный архив, выбрав размер тома из выпадающего списка, связанного с этим полем ввода. Создайте архив файлов, разделенных на тома, емкостью в 700 Кбайт.

Уровень – Синтез и оценивание



1. Создайте папки «Большие файлы» и «Маленькие файлы», соответственно записав туда файлы большого и маленького размеров.
2. Выполните архивацию файлов из папки, используя разные способы архивации:
 - с помощью контекстного меню;
 - с помощью кнопки **Добавить (Add...)**;
 - создайте архив с паролем (**Ctrl+P**).
3. Убедитесь, что уже сжатые файлы в формате .mp3 и .jpg совсем или почти не сжимаются.



4. Заполните таблицу «Архивация файлов» в текстовом процессоре Word, определите, во сколько раз уменьшился размер файла.

5. Убедитесь, что размер файла после создания архива почти равен исходному файлу.

Таблица 1.6

Имя файла	Тип файла	Истинный размер файла	Размер файла после создания архива	Во сколько раз уменьшился размер файла



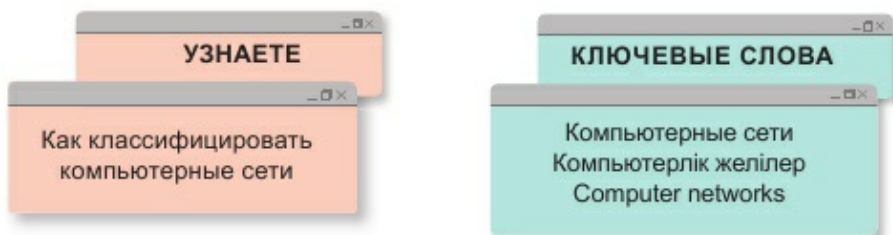
Архив данных – это копия данных компьютера, хранящаяся, как правило, на внешних носителях информации. С помощью внешнего носителя с архивом данных можно впоследствии восстановить данные в том виде, в котором они были на момент архивации.



Раздел 2

СЕТИ И ИХ БЕЗОПАСНОСТЬ

2.1. Компьютерные сети и их классификация



Компьютеры могут быть связаны в сети для коллективного использования данных и ресурсов. Сеть может быть очень простой и представлять собой два компьютера, соединенных одним кабелем (рис. 2.1). В состав сложной сети входят сотни компьютеров.



Рис. 2.1. Схема сети из двух компьютеров

ЗАПОМНИТЕ

Компьютерные сети – это объединение компьютеров для обмена информацией и совместного использования ресурсов.



Компьютерные сети разделены по:

- масштабу;
- типу организации работы компьютеров в сети;
- топологии;
- типу среды передачи.

ПОДУМАЙТЕ И ОТВЕЬТЕ

По масштабу компьютерные сети бывают: локальные, региональные, глобальные (рис. 2.2). Какая территория будет охвачена в составе перечисленных сетей?



Рис. 2.2. Масштаб компьютерных сетей

Существуют различные способы объединения компьютеров в сеть. Основными типами компьютерных сетей являются одноранговая сеть и сеть на основе сервера (рис. 2.3).

Сервер – абонент сети, отдающий в сеть свой ресурс и имеющий или не имеющий доступ к ресурсам сети. Он используется для хранения файлов, веб-страниц, для управления почтой, для хранения копий файлов.

Виды серверов: *файл-серверы* и *принт-серверы*, *серверы приложений*, *почтовые серверы*, *факс-серверы*, *коммуникационные серверы*, *сервер служб каталогов*.



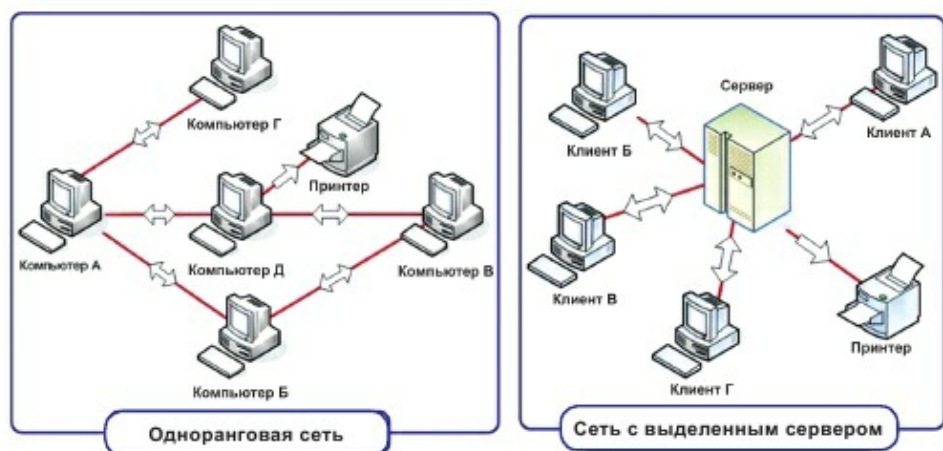


Рис. 2.3. Типы организации работы компьютеров в сети

В одноранговой сети нет выделенных серверов, функции управления сетью передаются по очереди от одного компьютера к другому.

В одноранговой сети все компьютеры равноправны. Каждый из них имеет уникальное имя компьютера и обычно пароль для входа в него во время загрузки операционной системы.

Компьютер-сервер – это центральный компьютер, который распределяет общие ресурсы между многими пользователями. Компьютер, который постоянно не включен в сеть, а подключается периодически, называется **клиентом**, или **абонентом**.

Для обеспечения возможности подключения к сети можно использовать различные устройства (рис. 2.4): компьютеры, сотовые телефоны, концентраторы, коммутаторы, маршрутизаторы, точки беспроводного доступа.



Рис. 2.4. Концентратор, коммутатор, маршрутизатор



Топология характеризует физическое расположение компьютеров, кабелей и других компонентов сети. **Физическая топология** определяет способ подключения компьютеров, принтеров и других устройств к сети.

Наиболее распространенные физические топологии локальных вычислительных сетей (рис. 2.5): шинная, кольцевая, звездообразная, ячеистая, иерархическая звезда.

ПРОАНАЛИЗИРУЙТЕ И ОЦЕНИТЕ

Проанализируйте информацию по образцу: *В звездообразной топологии имеется центральная точка подключения, в качестве которой обычно выступает такое устройство, как концентратор, коммутатор или маршрутизатор.*

По какому принципу построены топологии остальных компьютерных сетей, представленных на рис. 2.5? Оцените, какая из топологий сетей является наиболее распространенной.

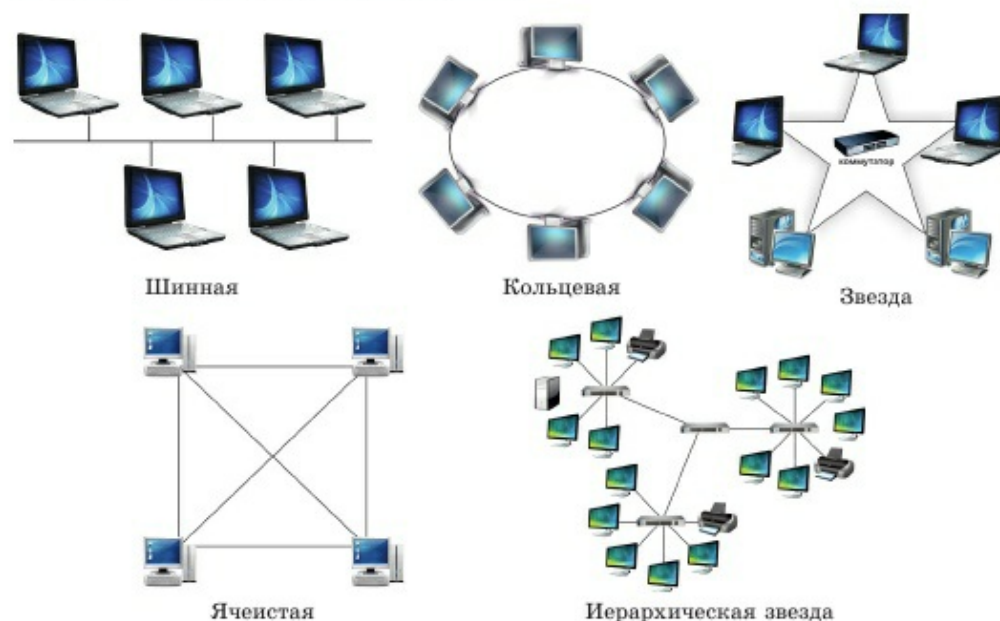


Рис. 2.5. Топология сетей

По типу среды передачи:

- **проводные** (витая пара, коаксиальный кабель, оптоволокно);





Рис. 2.6. Спутниковое подключение

– **беспроводные** (радиосвязь (WiFi, WiMAX), инфракрасная связь, СВЧ-связь (Bluetooth), Li-Fi – передача с помощью света).

Беспроводными локальными сетями называются сети для передачи и приема данных с помощью радиоволн. Область действия сети может ограничиваться лишь одной комнатой, но может охватывать и большее пространство.

Беспроводное подключение очень редко используется в стационарных компьютерах. Это идеальное решение для ноутбуков, умных телефонов и других подобных устройств.

Спутниковое подключение имеет самый широкий радиус подключения, но используется только в стационарных компьютерах (рис. 2.6).

В настоящее время мобильные телефоны превратились практически в полноценный компьютер, позволяющий выходить в Интернет, отправлять почту, смотреть онлайн передачи и многое другое. Но некоторые функции не могут работать без качественной связи. Именно по этой причине сейчас мы можем наблюдать быстрый рост и прогресс в области **беспроводных технологий связи**.

Тенденция увеличения передаваемого объема информации в системах сотовой связи создала предпосылки постоянной эволюции мобильных телесистем. В процессе решения проблем пропускной способности, развитие сотовых систем разделилось на четыре основных поколения: 1-е поколение – 1G (от англ. *Generation* – «поколение»), 1-е поколение – 1G, 2-е поколение – 2G, 3-е поколение – 3G, 4-е поколение – 4G (рис. 2.7).

На данный момент бурно развивается четвертое поколение мобильной связи – поколение с повышенными требованиями. К четвертому поколению принято относить перспективные технологии, позволяющие осуществлять передачу данных со скоростью, превышающей 100 Мбит/с.

Разрабатываемое пятое поколение мобильной связи – 5G. Скорость передачи данных в сетях доходит до 7 Гбит/с.

Канал связи – технические средства, позволяющие осуществлять передачу данных на расстоянии. Компьютеры могут обмениваться информацией с использованием каналов связи различной физической природы: кабельных, оптоволоконных, радиоканалов и др.





Рис. 2.7. 1G, 2G, 3G, 4G, 5G мобильной связи

Скорость передачи информации – количество информации, передаваемое за единицу времени. Общая схема передачи информации включает в себя **отправителя информации, канал передачи информации и получателя информации.**

Основной характеристикой каналов передачи информации является их *пропускная способность*. Пропускная способность канала – максимальная скорость передачи информации по каналу связи в единицу времени.

Объем переданной информации V вычисляется по формуле:

$V = q \cdot t$, где q – пропускная способность канала (в битах в секунду или другая), t – время передачи.

Обычно пропускная способность измеряется в битах в секунду (**бит/с**) и кратных единицах **Кбит/с** и **Мбит/с**.

Для обозначения больших скоростей передачи применяют более крупные единицы:

- килобит в секунду – Кбит/с (kbps);
- мегабит в секунду – Мбит/с (Mbps);
- гигабит в секунду – Гбит/с (Gbps).



РЕШИМ ВМЕСТЕ

1. Какое количество информации в мегабайтах можно передать при пропускной способности канала в 1 Мбит/с?

Решение.

1 Мегабит = 1 000 000 бит.

1 Килобит = 1000 бит.

1 000 000 бит/8 = 125 000 байт.

125 000 байт/1024 = 122,07 Кбайт.

122,07 кбайт/1024 = 0,119 Мбайт.

Ответ: 1 Мбит = 0,12 Мбайт.

При скорости 1 Мбит/с за один час из сети можно скачать:

0,12 Мбайт × 3600 с = 432 Мбайт.

2. Сколько секунд потребуется модему, передающему сообщения со скоростью 28 800 бит/с, чтобы передать 500 страниц текста в 30 строк по 60 символов каждая, при условии, что каждый символ кодируется 1 байтом?

Решение.

$V = 50 \cdot 50 \cdot 60 \cdot 8$ бит = 1 200 000 бит – объем файла в битах

$q = 28800$ бит/с – скорость передачи сообщения

$t = V/q = 1\,200\,000/28\,800 = 42$ с – потребуется модему.

**Уровень – Знание и понимание**

1. Как классифицируются сети по своим масштабам?
2. В чем отличие одноранговой сети от сети с выделенным сервером?
3. Какие топологии сетей вы знаете?
4. Где применяется беспроводное подключение, спутниковое подключение?
5. Какое подключение имеют сотовые телефоны?
6. Что такое *скорость передачи информации*?
7. Что такое *канал связи*?
8. В каких единицах измеряется пропускная способность канала?





Уровень – Применение



1. Необходимо объединить компьютеры в многоквартирном доме с несколькими подъездами. Какую топологию сети выберете? Какой тип кабеля? С какими средствами передачи? Обоснуйте свой выбор.

2. Какой тип сети вы бы выбрали для объединения трех смартфонов среди друзей? Обоснуйте свой выбор.

3. Скорость передачи данных через ADSL-соединение равна 256 000 бит/с. Передача файла через данное соединение заняла 3 мин. Определите размер файла в килобайтах.

(Ответ: 5625 Кбайт.)

4. Скорость передачи данных через ADSL-соединение равна 128 000 бит/с. Через данное соединение передают файл размером 625 Кбайт. Определите время передачи файла в секундах.

(Ответ: 40 с.)

5. Скорость передачи данных через ADSL-соединение равна 1 024 000 бит/с. Передача файла через данное соединение заняла 5 с. Определите размер файла в килобайтах.

(Ответ: 625 Кбайт.)



Уровень – Анализ

Перечислите наиболее распространенные физические топологии локальных вычислительных сетей. Придумайте различные способы соединения 10 компьютеров в компьютерном классе в сеть.

– Нарисуйте схемы топологии сетей.

– Найдите способ, обеспечивающий самый короткий маршрут передачи информации между любыми двумя абонентами.

– Рассмотрите достоинства и недостатки таких объединений. Заполните табл. 2.1.



Таблица 2.1

Критерии оценки сети	Шинная	Кольцевая	Звезда	Иерархи- ческая	Ячейчатая
<i>Экономические затраты на кабель</i>					
<i>Возможность нелегального подключения</i>					
<i>Возможность подключения абонента без остановки работы сети</i>					
<i>Возможность обмена информацией без сервера</i>					
<i>Влияет ли поломка компьютера абонента на работу сети?</i>					



Уровень – Синтез



Определите топологию сетей, изображенных на рис. 2.8.

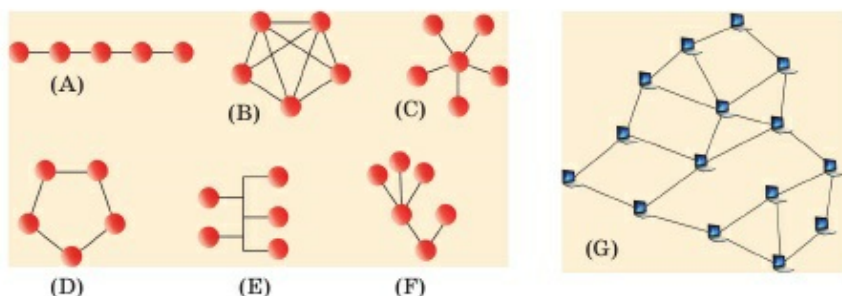


Рис. 2.8. Топология разнообразных сетей

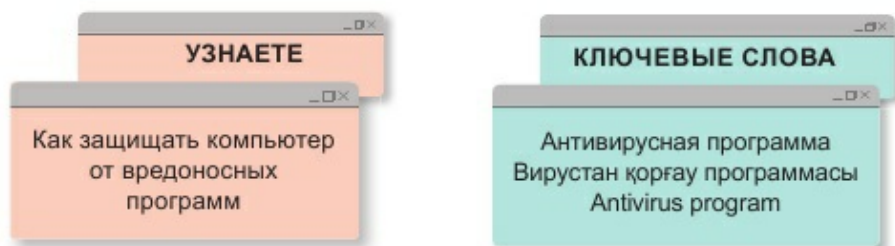


Уровень – Оценивание

Проведите мини-исследование. Оцените, какая топология сетей в классе, дома, в крупных супермаркетах, аэропорту.



2.2. Антивирусная защита



Человек как живой организм подвергается воздействию различных вирусов. А может ли «заболеть», «заразиться» вирусом компьютер? Как это происходит, как этого избежать и как «вылечить» компьютер? Причиной заражения компьютера является вирус, только компьютерный.

Вирус – это программа, которая активизируется, копируя саму себя в исполняемые объекты. Вирусы могут проникнуть в ваш компьютер из других зараженных компьютеров, через носители данных или через сеть (локальную или Интернет).

ЗАПОМНИТЕ



Вредоносная программа – (Malware – сокращение от «malicious software») – любое программное обеспечение, специально созданное для того, чтобы причинять ущерб компьютеру, серверу или компьютерной сети и хранящимся на них данным.

К вредоносным программам относятся:

- ✓ Черви – это программы, которые распространяют свои копии по сети.
- ✓ Программы-шпионы – устанавливаются скрытым путем на компьютеры для сбора информации о пользователе без его ведома.
- ✓ Троянские программы – это вредоносные программы, выполняющие несанкционированные пользователем действия. Такие действия могут включать:
 - удаление данных;



- блокирование данных;
- изменение данных;
- копирование данных;
- замедление работы компьютеров и компьютерных сетей.

Ниже перечислены некоторые типы вирусов и их описания.

1. Загрузочные вирусы – атакуют загрузочные секторы (гибкого или жесткого диска) и устанавливают свои подпрограммы, загружаемые при запуске компьютера.

2. Файловые вирусы – атакуют исполняемые программы, в частности, все файлы с расширениями **exe** и **com**.

3. Вирусы скриптов – являются разновидностью файловых вирусов. Вирусы скриптов способны заразить файлы других форматов, позволяющих выполнение скриптов, например **html**.

4. Макровирусы – атакуют документы, в которые могут вставляться макрокоманды (макросы). Эти вирусы часто внедряются в приложения обработки текстов или электронных таблиц, так как в эти типы файлов легко вставляются макросы.

5. Сетевые вирусы – распространяются по компьютерной сети. Именно они представляют главную угрозу для пользователей глобальной сети. К ним относятся: интернет-черви, троянские программы (**Trojan** и **Backdoor**).

ЗАПОМНИТЕ



Признаки, которые показывают, что компьютер может быть заражен вирусом:

- прекращение работы или неправильная работа ранее успешно функционировавших программ;
- медленная работа компьютера;
- невозможность загрузки операционной системы;
- исчезновение файлов и каталогов или искажение их содержимого;
- изменение даты и времени модификации файлов;
- изменение размеров файлов;
- неожиданное значительное увеличение количества файлов на диске;



- существенное уменьшение размера свободной оперативной памяти;
- вывод на экран непредусмотренных сообщений или изображений;
- подача непредусмотренных звуковых сигналов;
- частые зависания и сбои в работе компьютера.

ПРОАНАЛИЗИРУЙТЕ

На рис. 2.9. приведен алгоритм работы вирусов на компьютере. Проанализировав схему, объясните, каким образом будет осуществляться проникновение вируса на компьютер.

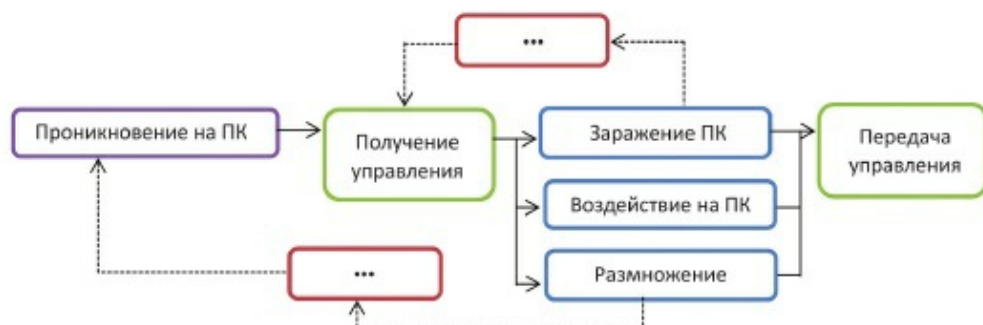


Рис. 2.9. Алгоритм работы вирусов на компьютере

Различают следующие виды антивирусных программ (рис. 2.10):

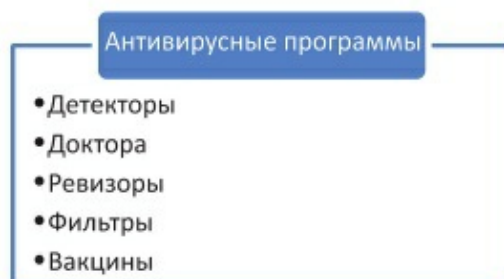


Рис. 2.10. Виды антивирусных программ



Программы-детекторы осуществляют поиск конкретного вируса в оперативной памяти и в файлах. Недостатком программ-детекторов является то, что они могут находить только те вирусы, которые известны разработчикам таких программ.

Программы-доктора и программы-вакцины находят зараженные вирусами файлы и «лечат» их. При «лечении» они удаляют из файла тело программы-вируса, возвращая файлы в исходное состояние.

Программы-ревизоры относятся к самым надежным средствам защиты от вирусов. Ревизоры запоминают исходное состояние программ, каталогов и системных областей диска компьютера, не зараженного вирусом, а затем по желанию пользователя сравнивают текущее состояние с исходным.

Программы-фильтры представляют собой небольшие резидентные программы, предназначенные для обнаружения подозрительных действий при работе компьютера, характерных для вирусов: попытки коррекции файлов с расширениями **com** и **exe**, загрузка резидентной программы и др. Программы-фильтры способны только обнаруживать вирус на самой ранней стадии его существования до размножения. Для уничтожения вирусов требуется применение других программ.

К профилактическим мерам защиты компьютера от вируса относятся:

- регулярная проверка компьютера с помощью антивирусных программ;
- контроль информации, поступающей из Интернета, через электронную почту, съемные носители;
- общие средства защиты информации – своевременное создание резервных копий данных, использование лицензионного программного обеспечения.

Применение совместно всех видов защиты информации позволит достигнуть высокой безопасности.

На сегодняшний день существуют различные антивирусные программы. Наиболее известные из них **Aidstest, McAfee VirusScan, Norton Anti-Virus, Antiviral Toolkit Pro, Doctor Web, 360 Total Security** и др.

Антивирусные программы быстро устаревают в связи с появлением новых вирусов, поэтому необходимо регулярное обновление их версий.

Рассмотрим подробнее работу антивирусной программы **360 Total Security**. Программа основана на пяти модулях: эффективных антивирусных ядрах **Avira** и **Bitdefender**, «облачной» системе анализа **360 Cloud**, проактивной технологии выявления угроз **QVMII** и восстанавливающем наборе инструментов **System Repair**.



Антивирусная программа 360 Total Security проверяет все подозрительные файлы в реальном времени, может анализировать объекты в защищенной среде и содержит ежедневно пополняемую библиотеку антивирусных сигнатур.

ПОЭКСПЕРИМЕНТИРУЙТЕ

Проанализируйте известные антивирусные программы. Найдите в Интернете бесплатную версию антивирусной программы 360 Total Security. Установите ее на ваш компьютер (рис. 2.11). Проведите с ее помощью комплексную проверку вашего компьютера на наличие вируса.

Ответьте на вопросы:

С помощью какого модуля можно очистить систему от ненужных объектов? Каким образом можно улучшить быстродействие устройств?

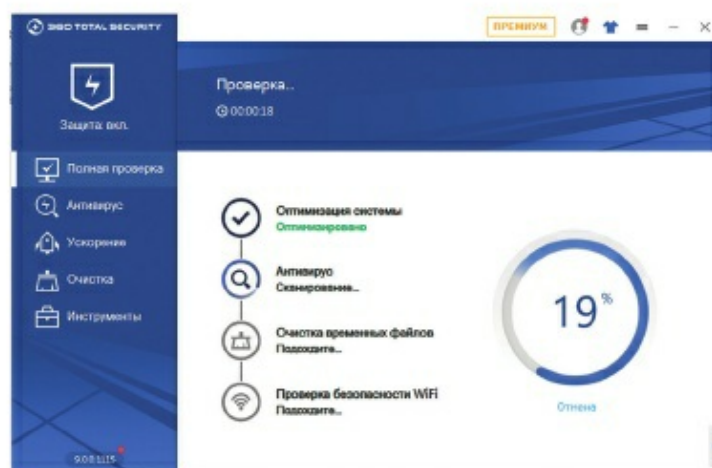


Рис. 2.11. Антивирусная программа 360 Total Security

Уровень – Знание и понимание



1. Какие виды вирусов вам известны?
2. Кто и для чего создает вредоносные программы?
3. Какие признаки проявления вирусов на компьютере вы знаете?
4. Какие антивирусные программы наиболее часто используются?
5. Какие антивирусные программы вы применяете на своем компьютере?
6. Существует ли такая вероятность, что та или иная антивирусная программа не может обезвредить какие-либо вирусы? Каковы будут ваши действия в таких случаях?





Уровень – Применение



Что вы будете делать, если на ваш компьютер проникнет вирус? Как избежать заражения компьютерным вирусом?



Уровень – Анализ

Классифицируйте вредоносные программы по принципу действия и их «вредности». Используйте различные источники информации.



Уровень – Синтез



1. Почему даже чистая отформатированная «флешка» может стать источником заражения вирусом?
2. Изучите вредоносную троянскую программу, используя различные источники информации.



Уровень – Оценивание

Заполните диаграмму Венна (рис. 2.12). Запишите в нее то, что знаете об отличиях между компьютерными вирусами и антивирусными программами. Что объединяет их? Объясните.



Рис. 2.12. Диаграмма Венна





Антивирусные программы основаны на следующих принципах: проверка файлов, проверка загрузочных секторов дисков и оперативной памяти, поиск в них известных и новых вирусов.

Большинство антивирусных программ сочетают в себе, во-первых, функцию постоянной защиты – **антивирусный монитор**. Антивирусный монитор запускается автоматически при старте операционной системы. Он работает в качестве фонового системного процесса, проверяя на вредоносность совершаемые другими программами действия.

Основная задача антивирусного монитора – обеспечение максимальной защиты от вредоносных программ.

Во-вторых, антивирусные программы содержат функцию защиты по требованию пользователя – **антивирусный сканер**. Такой сканер запускается по заранее выбранному расписанию или в произвольный момент пользователем. Он производит поиск вредоносных программ в оперативной памяти, а также на жестких и сетевых дисках компьютера.



ТЕСТОВЫЕ ЗАДАНИЯ К РАЗДЕЛАМ 1, 2

Из предложенных альтернативных ответов выберите только один верный. Каждый правильный ответ оценивается в 1 балл. Всего 15 баллов.

1. Перевод текста с английского языка на русский является процессом:

- а) хранения информации;
- б) передачи информации;
- в) поиска информации;
- г) обработки информации;
- д) ни одним из перечисленных выше процессов.

2. За единицу измерения информации принимается:

- а) 1 бод;
- б) 1 бар;
- в) 1 бит;
- г) 1 кг;
- д) 1 фут.

3. Как изменяет неопределенность 1 бит информации:

- а) не изменяет;
- б) уменьшает в два раза;
- в) увеличивает в два раза;
- г) уменьшает в восемь раз?

4. Что такое 1 байт:

- а) 1024 Кбайт;
- б) 10 Мбайт;
- в) 8 бит;
- г) 4 бит?

5. Чему равен 1 Гбайт:

- а) 1024 Кбайт;
- б) 1000 Кбайт;
- в) 1024 Мбайт;
- г) 1000 Мбайт?

6. Чему равны 4 килобайта:

- а) 5120 байт;
- б) 4000 байт;
- в) 4096 байт;
- г) 4000 бит?



7. Программой-архиватором называют:

- а) программу для уменьшения информационного объема (сжатия) файлов;
- б) программу резервного копирования файлов;
- в) интерпретатор;
- г) транслятор;
- д) систему управления базами данных.

8. Архивный файл представляет собой:

- а) файл, которым долго не пользовались;
- б) файл, защищенный от копирования;
- в) файл, сжатый с помощью архиватора;
- г) файл, защищенный от несанкционированного доступа;
- д) файл, зараженный компьютерным вирусом.

9. Какое из названных действий можно произвести с архивным файлом:

- а) переформатировать;
- б) распаковать;
- в) просмотреть;
- г) запустить на выполнение;
- д) отредактировать?

10. Степень сжатия файла зависит:

- а) только от типа файла;
- б) только от программы-архиватора;
- в) от типа файла и программы-архиватора;
- г) от производительности компьютера;
- д) от объема оперативной памяти персонального компьютера, на котором производится архивация файла.

11. Отличительными особенностями компьютерного вируса являются:

- а) значительный объем программного кода;
- б) необходимость запуска со стороны пользователя;
- в) способность к повышению помехоустойчивости операционной системы;
- г) маленький объем; способность к самостоятельному запуску и многократному копированию кода, к созданию помех корректной работе компьютера;
- д) легкость распознавания.



12. Назначение антивирусных программ под названием «детекторы»:

- а) обнаружение и уничтожение вирусов;
- б) контроль возможных путей распространения компьютерных вирусов;
- в) обнаружение компьютерных вирусов;
- г) «излечение» зараженных файлов;
- д) уничтожение зараженных файлов.

13. К антивирусным программам не относятся:

- а) сторожа;
- б) фаги;
- в) ревизоры;
- г) интерпретаторы;
- д) вакцины.

14. Локальные компьютерные сети в качестве средства общения используются:

- а) для общения людей непосредственно;
- б) только для осуществления обмена данными между несколькими пользователями;
- в) для организации доступа к общим для всех пользователей устройствам вывода – принтерам, графопостроителям и общим информационным ресурсам местного значения;
- г) для осуществления обмена данными между несколькими пользователями, для организации доступа к общим для всех пользователей устройствам вывода (принтерам), а также к общим информационным ресурсам местного значения;
- д) только для организации доступа к общим для всех пользователей информационным ресурсам.

15. Глобальные компьютерные сети как средство коммуникации возникли:

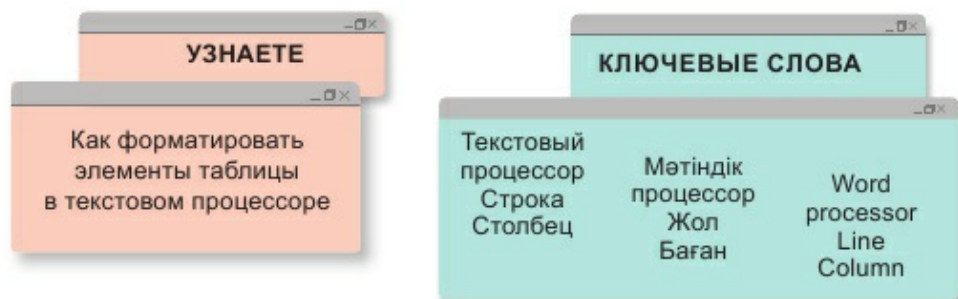
- а) когда появились компьютеры;
- б) когда созрела общественная потребность общения между людьми, проживающими в разных точках планеты и появились соответствующие технические возможности (системы и сети компьютерной коммуникации);
- в) когда свершилась научно-техническая революция;
- г) когда созрела общественная потребность общения между людьми, проживающими в разных точках планеты;
- д) когда созрела общественная потребность общения между людьми.



Раздел 3

РЕШЕНИЕ ЗАДАЧ С ПОМОЩЬЮ
ЭЛЕКТРОННЫХ ТАБЛИЦ

3.1. Таблицы в текстовом процессоре



Таблицы используются для упорядочения и представления данных, рисунков, объектов и других элементов. Они позволяют выстроить числа в столбцы, а затем отсортировать их, а также выполнить различные вычисления.

ЗАПОМНИТЕ



Таблица – это объект, который добавляется в документ для структурирования данных.

Таблица состоит из **столбцов** и **строк**, а их пересечения называются **ячейками**. Таблица имеет границу в виде тонкой сплошной линии черного цвета. Граница сохраняется при печати, а в случае удаления границы линии сетки отображаются на экране. На рис. 3.1 представлены компоненты таблицы: *маркер перемещения, интервалы между ячейками, символы конца строки, символы конца ячейки, маркер изменения размера таблицы*. Символы конца строки и конца ячейки видны в таблице, если включена команда ¶ (скрытые символы форматирования) из группы **Абзац** вкладки **Главная**.



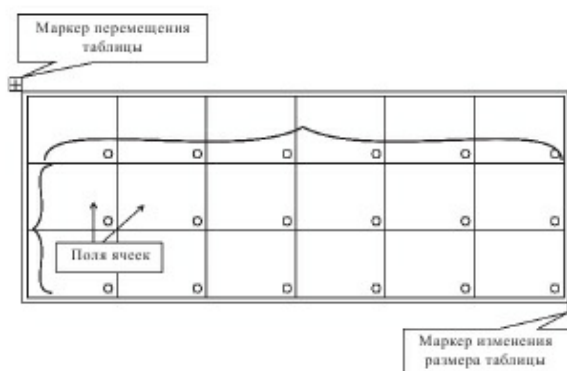


Рис. 3.1. Компоненты таблицы

СОЗДАДИМ ВМЕСТЕ

Создадим таблицу разными способами. Чтобы вставить таблицу в документ, нужно установить курсор туда, куда должна быть добавлена таблица.

Первый способ.

1. Выберите на Ленте вкладку **Вставка**, группу **Таблицы**, откройте список **Таблица**.
2. В списке **Вставка таблицы** (рис. 3.2) выделите нужное число строк и столбцов (максимальный размер таблицы 10×8).

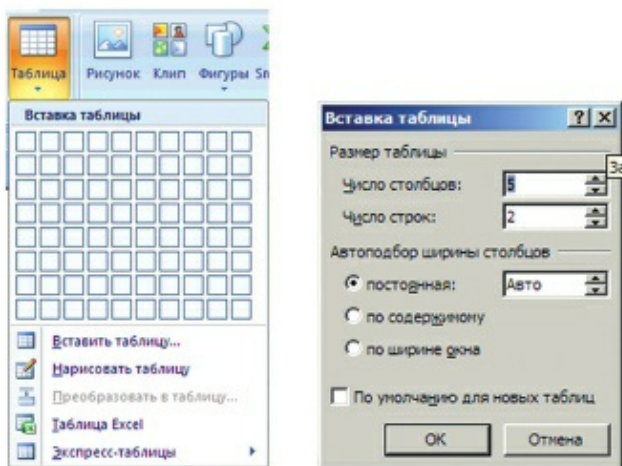


Рис. 3.2. Создание таблицы



Второй способ.

1. Выберите на Ленте вкладку **Вставка**, группу **Таблицы** (рис. 3.2), откройте список **Таблица**.
2. В списке **Вставка таблицы** выберите пункт **Вставить таблицу**.
3. В окне **Вставка таблицы** укажите число строк и столбцов.
4. Нажмите на кнопку **Ок**.

Третий способ.

1. Выберите на Ленте вкладку **Вставка**, группу **Таблицы**, откройте список **Таблица** (рис. 3.3).
2. Выберите пункт **Нарисовать таблицу**. Карандашом нарисуйте таблицу: рисуйте сначала внешний контур, затем внутренние разделители. Таблица сначала обозначается пунктиром, показывающим, как будет располагаться линия. Если отпустить кнопку мыши, линия станет разделительной.

Все таблицы, создаваемые в текстовом процессоре, выглядят одинаково: набор белых ячеек с черными границами. С помощью команд текстового процессора Word таблицу можно сделать яркой и эффектной за несколько секунд. Для этого используйте дополнительные вкладки (рис. 3.3) **Конструктор** и **Макет** (Работа с таблицами), которые появляются после создания таблицы или с помощью контекстного меню выделенной части таблицы.

С помощью дополнительной вкладки **Конструктор** можно:

- добавить (удалить) разделительные линии – карандашом **Нарисовать таблицу**;

можно также удалить их ластиком **Ластик** с вкладки **Конструктор** (Работа с таблицами);



Рис. 3.3. Дополнительные вкладки **Конструктор** и **Макет**



– к созданной таблице добавить строки (столбцы, ячейки) – поставить курсор в ячейку таблицы и вызвать контекстное меню **Вставить, Вставить столбцы** (рис. 3.4) или на вкладке **Макет** (Работа с таблицами), группа **Строки и столбцы**, команды **Вставить сверху/снизу/слева/справа**;

– добавить следующую строку в конец таблицы – поставить курсор в последнюю ячейку последней строки таблицы и нажать клавишу **Tab**.

– изменить размер строк/столбцов – контекстное меню выделенной таблицы, окно **Свойства таблицы**, вкладка **Столбец/Строка** – указать ширину/высоту или на вкладке **Макет**, группа **Размер ячейки**, команды **Высота строки** таблицы/**Ширина столбца** таблицы;

– **столбцы/строки** копировать, удалять, вставлять или перемещать – **Выделить столбец/строку**, вызвать контекстное меню (правая кнопка мыши), выбрать нужный пункт.

Для ввода данных в таблицу нужно придерживаться следующих правил:

– щелкнуть на ячейке и ввести текст с клавиатуры или вставить из буфера обмена при копировании текста. Если текст не помещается в строке, то он переносится на другую строку и увеличивает высоту строки;

– для изменения ориентации текста в ячейке необходимо установить курсор в ячейку и в меню **Формат** выбрать команду **Направление текста**;

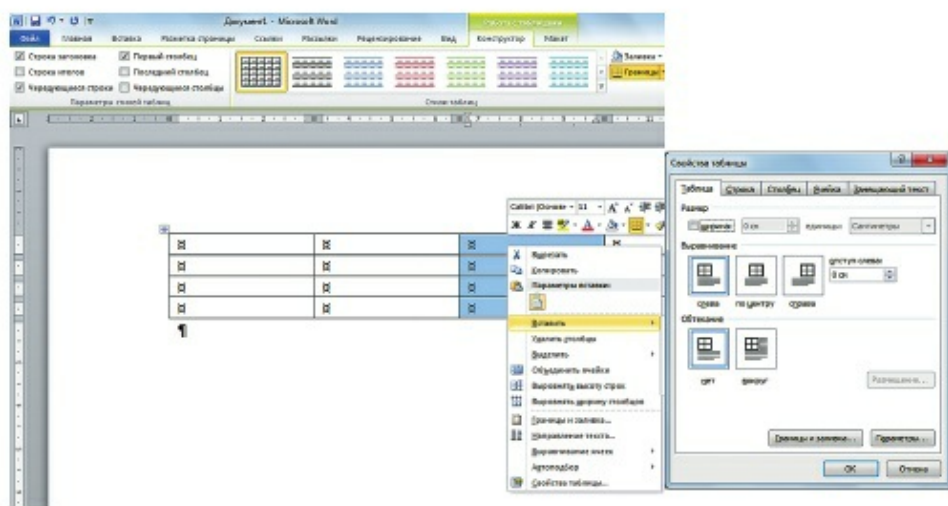


Рис. 3.4. Работа с таблицей



- для изменения выравнивания текста в ячейке на панели инструментов **Таблица и границы** выберите параметр выравнивания по вертикали и горизонтали;
- для перемещения, копирования и удаления текста в ячейках необходимо выделить этот текст. Выделенный текст можно удалять клавишей **Delete** или **Backspace**.

Для редактирования элементов (ячеек, строк, столбцов) необходимо выделить эти элементы, а затем использовать меню **Таблица** или **контекстное меню**. К операциям редактирования таблиц относятся:

- вставить и удалить строки и столбцы;
- объединить и разбить ячейки;
- разбить таблицу.

После того как установили формат таблицы и ширину полей, можно приступить к **форматированию** элементов таблицы.

Текстовый процессор Word содержит несколько десятков команд для профессионального форматирования данных.

ЗАПОМНИТЕ



Форматирование – это изменение свойств данных, определяющих его внешний вид, без изменения содержимого.

Форматирование данных означает: изменение полей, абзацев и отступов, выравнивание текста (по левому, по правому краям, по центру, по ширине), установка межстрочного интервала, вставка графических объектов, осуществление проверки орфографии и др.

Для форматирования таблицы выделите таблицу, выберите вкладку **Конструктор** группы **Стили таблиц**. Кнопка **Дополнительные параметры** раскрывает окно **Встроенные стили**, откуда можно выбрать любой стиль для быстрого оформления таблицы.

Уровень – Знание и понимание



1. Для чего применяется таблица?
2. Перечислите основные компоненты таблицы. Для чего они нужны?
3. Какие способы создания таблиц вы знаете?
4. Что такое *форматирование данных*?
5. Как форматировать таблицу с применением стилей?
6. Какие дополнительные вкладки появляются при работе с таблицами?



Уровень – Применение



Создайте в текстовом процессоре Word таблицу из четырех строк и четырех столбцов.

- Выделите всю таблицу и установите шрифт **Arial** в 12 пунктов.
- Задайте в таблице заголовок, объединив ячейки первой строки.
- Заполните табл. 3.1 данными о свойствах веществ.
- Если вам при заполнении таблицы понадобится изменить ширину столбца, то установите курсор на границу столбца так, чтобы он приобрел вид двусторонней стрелочки. Нажав и удерживая левую кнопку мыши, расширьте или уменьшите ширину столбца.

Таблица 3.1

Свойства некоторых веществ			
Свойства	Кислород	Железо	Вода
Агрегатное состояние	Газ	Твердое	Жидкость
Цвет	–	Серое	–

- Добавьте еще одну строку в таблице и запишите в ней данные о **запахе** вещества.

Таблица 3.2

Свойства некоторых веществ			
Свойства	Кислород	Железо	Вода
Агрегатное состояние	Газ	Твердое	Жидкость
Запах	–	–	–
Цвет	–	Серое	–

- Отформатируйте созданную вами таблицу (табл. 3.3).
 - Сначала отцентрируйте заголовок таблицы. Для этого выделите первую строку, нажмите кнопку **По центру**, затем – кнопку **Полужирный**.
 - Сделайте обрамление таблицы, как на образце.
- Используйте вкладку **Главная** группы **Абзац**, пункт меню **Границы и заливка**, установите ширину линии в 1,5 пт (пунктов), цвет – синий. Затем выполните обрамление таблицы, используя **Образец** и **Тип обрамления**, тип – **Все**.

Таблица 3.3

Свойства некоторых веществ			
Свойства	Кислород	Железо	Вода
Агрегатное состояние	Газ	Твердое	Жидкость
Запах	–	–	–
Цвет	–	Серое	–





Уровень – Анализ

Создайте кроссворд в текстовом процессоре Word (рис. 3.5). После ввода ответов вы узнаете, какое слово зашифровано в закрашенном столбце.

Для вставки символов используйте вкладку **Вставка** на **Ленте**, из группы **Символы** выберите **Другие символы**, **Шрифт: Wingdings, Wingdings1, Wingdings2**.

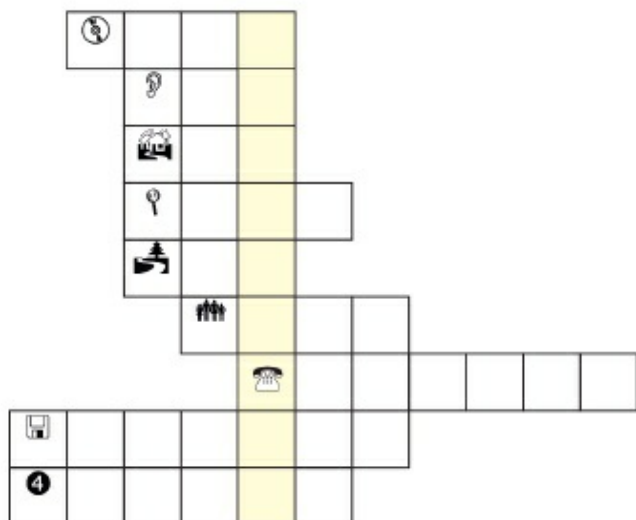


Рис. 3.5. Кроссворд «Устройство, выполняющее заданную последовательность операций»



Уровень – Синтез



Создайте и напечатайте таблицу, в которой можно увидеть сведения о 5–7 книгах, хранящихся в домашней библиотеке. Подумайте, какие столбцы должны быть в таблице, как они должны называться. Добавьте в таблицу еще 3 строки и заполните их.



Уровень – Оценивание

Проведите исследование по всем способам создания таблицы и установите, в каком случае удобнее использовать тот или иной способ, недостатки каждого из способов (если они есть). Какой из этих способов является универсальным?



3.2. Форматирование элементов электронных таблиц



При работе с информацией часто требуется создание таблиц, проведение различных вычислений с использованием формул, выполнение различных экономических, бухгалтерских и инженерных расчетов, построение графиков и диаграмм, создание математических моделей, проведение анализа данных.

Для проведения расчета данных, представленных в виде таблицы, были разработаны пакеты прикладных программ – табличные процессоры. Это специальный комплекс программ для управления электронной таблицей. В настоящее время наиболее популярным табличным процессором является электронная таблица Excel.

ЗАПОМНИТЕ



Электронная таблица Excel – это приложение, позволяющее автоматизировать обработку данных, представленных в виде таблиц.

Основной документ, используемый электронной таблицей Excel для хранения и обработки данных, называется **книгой (рабочей книгой)** и имеет расширение **.xlsx, xls**.

Рабочая книга состоит из отдельных **рабочих листов**, каждый из которых может содержать данные. По умолчанию каждая вновь созданная



рабочая книга содержит 3 рабочих листа, количество их можно изменять. Каждый рабочий лист можно переименовывать.

Рабочий лист разделен на **строки и столбцы**, на пересечении которых образуются ячейки. **Ячейка** – основная единица хранения данных.

Каждая ячейка имеет свой адрес, который составляется из заголовка столбца (A, B, C, D...), расположенного в верхней части таблицы, и заголовка строки, находящегося в левой части таблицы (1, 2, 3, 4...).

Активизировать электронную таблицу Excel можно разными способами, например командами **Пуск – Все программы – Microsoft Office – Microsoft Excel** или двойным щелчком по значку Excel на рабочем столе.

После запуска приложения открывается окно программы, состоящее из:

- 1) **строки заголовка** с названием рабочей книги в центре окна;
- 2) **панели быстрого доступа** – для быстрого доступа часто используемых команд;
- 3) **ленты с вкладками** для быстрого нахождения команд;
- 4) **строки формул (командная строка)** с отображением адреса активной ячейки и ее содержимым;
- 5) **рабочей зоны** текущего рабочего листа, состоящей из **Заголовка столбцов** и **Номеров строк**;
- 6) **строки состояния**, выводящей сообщение о текущей работе (**Готово, Ввод, Укажите** и т. п.).

Одновременно можно работать на нескольких листах, вводить данные, править, форматировать, производить с ними вычисления.

Форматирование в электронной таблице Excel применяется для облегчения восприятия данных. Форматировать можно выделенные (активные) ячейки или области. Отформатированные ячейки сохраняют свой формат, пока к ним не будет применен новый формат или не удален старый. При вводе значения в ячейку к нему применяется уже используемый в ячейке формат.

Для быстрого копирования форматов из выделенных ячеек в другие ячейки можно использовать команду **Формат по образцу** (рис. 3.6) вкладки **Главная**, группа **Буфер обмена**.

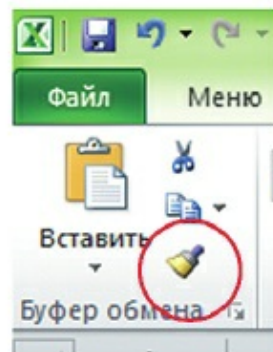


Рис. 3.6. Формат по образцу



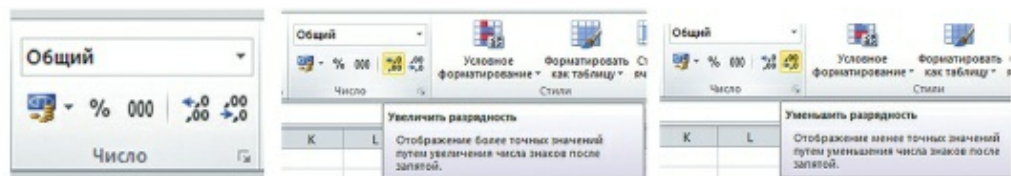


Рис. 3.7. Разрядность числа

Форматирование подразумевает изменение:

- цвета шрифта,
- цвета заливки,
- границ,
- центрирование и т. д.



Рис. 3.8. Команды группы Шрифт

Изменение разрядности чисел производится командами группы **Число**: **Увеличить разрядность** или **Уменьшить разрядность** (рис. 3.7):

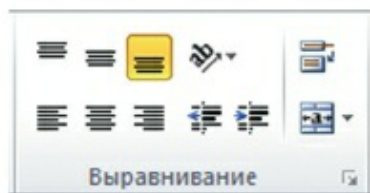




Рис. 3.9. Выравнивание

Цвета, размер и начертание шрифта, цвета заливки и границы в выделенной области меняются командами из группы **Шрифт** (рис. 3.8) вкладки **Главная** аналогично форматированию информации в других офисных приложениях.


Несколько ячеек таблицы можно объединить командой **Объединить** и поместить в центре (Вкладка **Главная**, группа **Выравнивание**) .

Отцентрировать информацию в ячейке можно командами из группы **Выравнивание** на вкладке **Главная** (рис. 3.9). Выравнивать можно по горизонтали и вертикали.

Ориентация текста внутри ячейки выполняется командой **Ориентация**  из группы **Выравнивание** на вкладке **Главная**.

ВЫПОЛНИТЕ

Изменение ширины столбца или высоты строки.

- Выделите столбец или несколько столбцов.
- Поместите курсор в заголовке таблицы между столбцами так, чтобы белый крест изменился на черный со стрелками .
- Чтобы увеличить (уменьшить) ширину столбцов, передвиньте левую кнопку мыши вправо (влево) и отпустите кнопку мыши.



- Если выделите несколько столбцов, то при изменении ширины столбцов изменяется ширина всех столбцов.

Ширину столбца можно быстро задавать в соответствии с набранным текстом.

Например:

- наведите курсор мыши на правую границу столбца возле его имени и дважды щелкните левой кнопкой мыши;
- граница столбца автоматически выравнивается по самому длинному слову, набранному в его ячейках (рис. 3.10).

	A	B
1	красный	
2	оранжевый	
3	желтый	
4	зеленый	
5	голубой	
6	синий	
7	фиолетовый	

Рис. 3.10. Автоматическое выравнивание

В электронной таблице Excel предлагаются **готовые стили таблицы** (рис. 3.11), которые можно использовать для быстрого форматирования

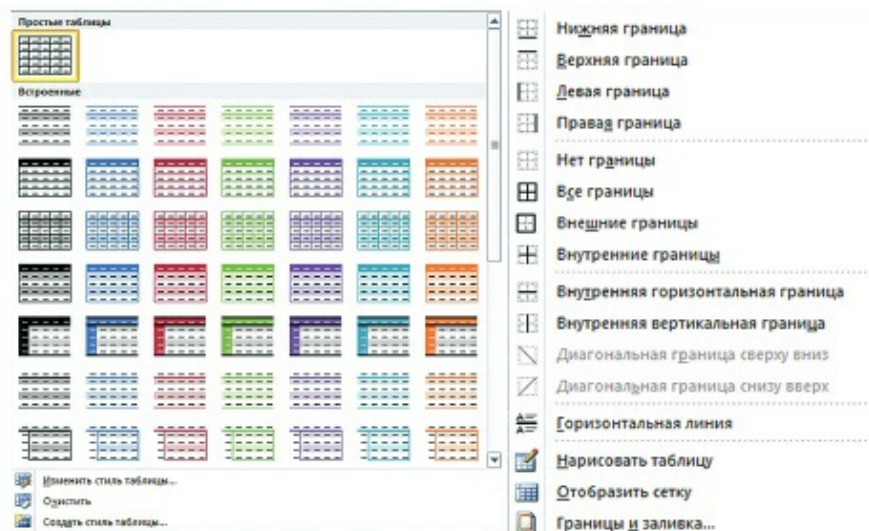


Рис. 3.11. Стили форматирования и Границы



таблицы. Если такие стили не соответствуют потребностям пользователя, можно создать и применить настраиваемый стиль таблицы.

ВЫПОЛНИТЕ

Для добавления границы:

- Выберите ячейки.
- Щелкните на стрелку раскрывающегося списка, выбрав команду **Границы и Заливка** на вкладке **Главная** (рис. 3.11).
- Выберите стиль границы, который вы хотите использовать.
- Вы можете нарисовать границы и менять **Тип линии** и **Цвет** границ с помощью **Нарисовать границы** инструментами в нижней части границы выпадающего меню.

Чтобы изменить цвет шрифта:

- Выберите ячейки, в которых нужно изменить цвет шрифта.
- Щелкните стрелку раскрывающегося списка рядом с цветом шрифта командой меню **Цвет текста** на вкладке **Главная**.
- Выберите цвет шрифта, который вы хотите использовать.

Одним из преимуществ электронных таблиц является то, что не все данные при создании таблиц нужно вводить вручную. Ввод данных, которые повторяются, имеет определенную закономерность и его можно автоматизировать.

Для этого используется **Маркер заполнения** (рис. 3.12).

ЗАПОМНИТЕ



Маркер заполнения – небольшой черный квадрат в правом нижнем углу выделенной ячейки или выделенного диапазона.

При наведении на **Маркер заполнения** указатель мыши принимает вид тонкого черного крестика.

Маркер заполнения удобен для заполнения соседних ячеек содержимым выделенной ячейки.

ВЫПОЛНИТЕ

- Наведите на **Маркер заполнения** указатель мыши, чтобы он принял вид черного креста.
- Нажмите левую клавишу мыши и, не отпуская ее, протяните вниз (или вправо, влево, вверх) на нужное количество ячеек.
- Отпустите кнопку. Значения будут скопированы.



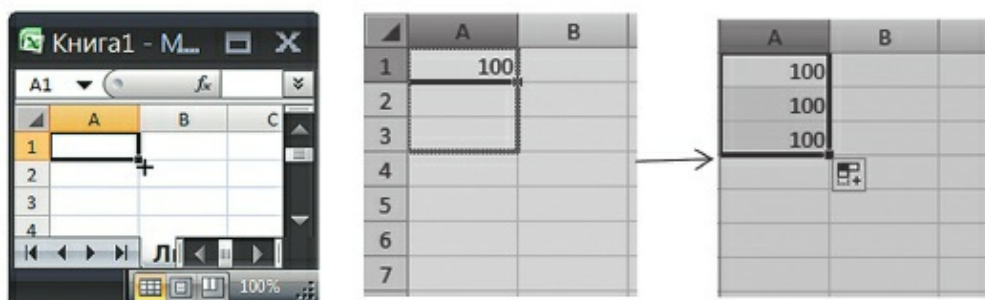


Рис. 3.12. Маркер заполнения

С помощью Маркера заполнения можно быстро формировать числовые последовательности, последовательности дат и рабочих дней, работать со списками автозаполнения.

ЗАПОМНИТЕ

Автозаполнение – это операция ускоренного автоматического заполнения ячеек данными, текстом, числами или формулами без использования обычных способов ввода.

Автозаполнение позволяет (рис. 3.13):

- быстро вводить некоторые стандартные или созданные пользователем наборы данных (названия месяцев, названия видов товаров и т. п.);
- быстро заполнять ячейки числовыми последовательностями (нумерацией, датами, часами и т. д.);
- быстро вводить в ячейки одинаковые данные;
- быстро вводить новые формулы, используя уже введенные формулы.

Последовательность данных в Excel может быть стандартной, т. е. изначально заложенной, или же она может задаваться пользователем.

ЗАПОМНИТЕ

Последовательности – это введенные в соседние ячейки данные, представляющие собой некоторую закономерную последовательность.



ВЫПОЛНИМ ВМЕСТЕ

Введите названия месяцев с января по июнь (слева направо) с использованием стандартной последовательности. Для этого:

- в начальную ячейку введите значение «Январь»;
- нажмите на кнопку **Ввод** на панели формул (или нажать Enter, а затем вернуться в ячейку, где введено значение «Январь»);
- установите курсор мыши на Маркер заполнения;
- растяните выделение автозаполнения вправо на 5 ячеек и отпустите кнопку мыши.

Если растяните выделение влево или вверх, то тоже произойдет заполнение месяцами, но в обратном порядке, т. е. Январь, Декабрь, Ноябрь и т. д. По диагонали автозаполнение не выполняется (рис. 3.13).

	Январь	Февраль	Март	Апрель
2001	1	3	5	7
2002	1	3	5	7
2003	1	3	5	7
2004	1	3	5	7
2005	1	3	5	7

Рис. 3.13. Автозаполнение

Чтобы создать последовательность по другому закону, например, с шагом два, необходимо:

- в соседние ячейки ввести два значения-образца с требуемым шагом;
- обычным способом выделить эти ячейки;
- растянуть выделение автозаполнения до требуемого размера.

В результате (если, например, ввести значения 00:00 и 02:00) продолжением последовательности будут значения 04:00, 06:00, 08:00 и т. д.

0:00	2:00	4:00	6:00	8:00
------	------	------	------	------



Используя подобную технологию, можно получить следующие последовательности (рис. 3.14):

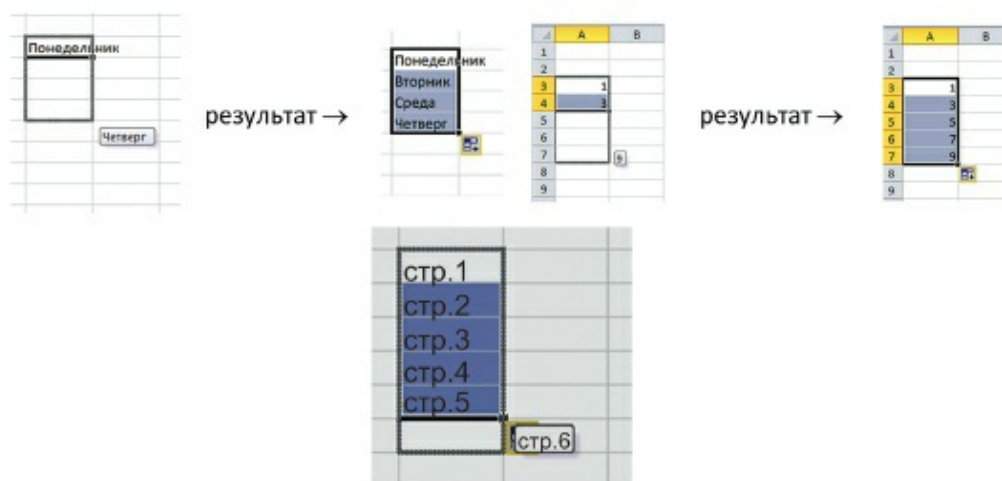


Рис. 3.14. Последовательности

Уровень – Знание и понимание



1. Что такое *форматирование в электронной таблице*?
2. Какой командой объединяются ячейки?
3. Можно ли в электронной таблице Excel написать текст по диагонали?
4. Как можно изменить ширину столбца или высоту ячейки?
5. Для чего используются границы?
6. Как можно изменить цвет шрифта?
7. Чем отличается таблица в текстовом процессоре от таблицы в электронной таблице?
8. Для чего применяется Маркер заполнения?

Уровень – Применение



1. Установите соответствие между определением и названием в табл. 3.4.

Таблица 3.4

Определение	Название
1. Это приложение, обрабатывающее информацию, которую можно представить в виде таблиц и автоматизировать обработку таких данных.	а. Электронные таблицы или табличные процессоры



Продолжение

1	2
2. Это специальные пакеты программ, обрабатывающие информацию, которую можно представить в виде таблиц.	b. Excel
3. Это файл, в котором на отдельных листах хранится информация (данные).	c. .xlsx
4. Это составная часть книги Excel, состоящая из строк и столбцов, на пересечении которых находятся ячейки.	d. Рабочая книга Excel
5. Это расширение файлов, созданных в электронной таблице Excel.	e. Ячейка
6. Это основная единица хранения данных, образуемая на пересечении столбца и строки рабочего листа.	f. Рабочий лист

2. Создайте и отформатируйте данные в таблице по образцу (табл. 3.5).

Таблица 3.5

						Курс доллара:
РАСЧЕТ ЗАРАБОТНОЙ ПЛАТЫ						
№ п/п	ФИО	Начислено	Налоги		К выдаче в тенге	Эквивалент, \$
			подох. 5%	пенс. 10%		
1	2	3	4	5	6	7
1		10 800				
1	2	3	4	5	6	7
2		105 000				
3		135 000				
Итого						
Максимальная						
Минимальная						
Средняя						



Уровень – Анализ

Отметьте, что относится к форматированию (табл. 3.6):

Таблица 3.6

<input type="checkbox"/>	Изменение ориентации текста в ячейке
<input type="checkbox"/>	Изменение цвета шрифта
<input type="checkbox"/>	Изменение содержимого ячейки
<input type="checkbox"/>	Выравнивание содержимого по правому краю ячейки
<input type="checkbox"/>	Изменение размера ячейки
<input type="checkbox"/>	Выделение ячеек
<input type="checkbox"/>	Установка и изменение границ
<input type="checkbox"/>	Изменение заливки ячейки
<input type="checkbox"/>	Изменение типа данных (число, дата, текст и т. п.)
<input type="checkbox"/>	Удаление каких-либо символов
<input type="checkbox"/>	Удаление информации
<input type="checkbox"/>	Изменение разрядности чисел

Уровень – Оценивание

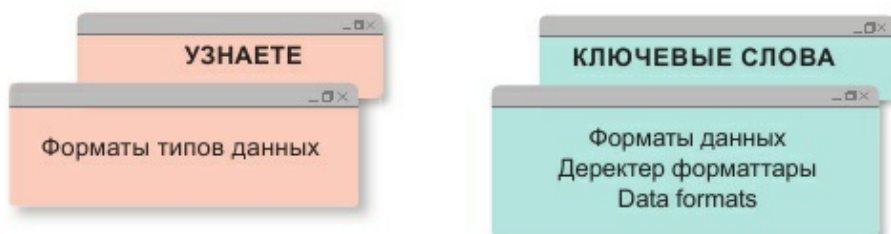
Проведите исследование по рис. 3.15. Для чего предназначена данная таблица? Какие расчеты будут проводиться в ней? Будет ли данная таблица автоматизировать расчет данных в таблице? Дайте оценку стилю форматирования таблицы. Используется ли в данной таблице Маркер заполнения?

		Средний балл	0,00	0,47	1,00	1,33	1,67	2,00	2,33	2,67	3,00	3,33	3,67	4,00	4,00
Гимназия №1		Название задания или контрольной работы													
		Максимальное число баллов													
Имя учителя		Введите в эту таблицу все задания и контрольные работы, а также присуждаемые за них баллы. После завершения работы с инструкциями удалите это текстовое поле, щелкнув его границу и нажав клавишу DELETE.													
Класс/проект		Общее число заданий и контрольных работ	0												
Год/семестр/квартал		Общее максимальное число баллов	0												
№И учащегося	Номер учащегося	Среднее значение	Буквенная оценка	Средний балл (GPA)											
Иванова Катя	1	1,00%	F	0,00											
Амишев Данир	2														
Успеваемость класса		Среднее значение	Буквенная оценка	Средний балл											
Среднее значение		1,00%	F												
Высший балл		3,00%	F												
Нижний балл		3,00%	F												

Рис. 3.15. Расчеты данных



3.3. Форматы типов данных



Какие существуют **форматы типов данных** в электронной таблице Excel, а точнее – форматы ячеек, которые отображают данные в нужном нам виде? Давайте рассмотрим их.

ВЫПОЛНИМ ВМЕСТЕ

Для того чтобы указать формат отображения данных в ячейке, выделите ячейку или диапазон ячеек.

Щелкните по выделенной области правой кнопкой мыши и выберите из контекстного меню пункт **Формат ячеек**. В открывшемся окне перейдите на вкладку **Число** (рис. 3.16). Выберите интересующий формат и нажмите ОК.

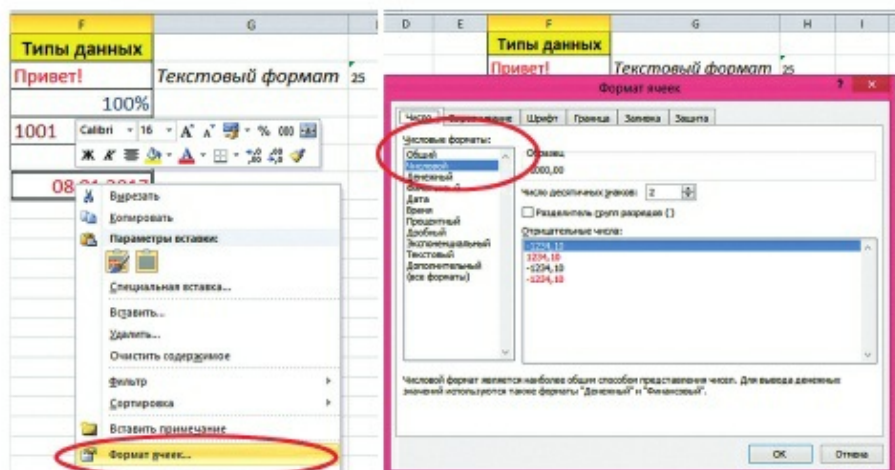



Рис. 3.16. Формат ячеек



Следующий способ – это вызов из пункта выпадающего меню **Другие числовые форматы...** вкладки **Главная** в диалоговом окне **Все форматы** (рис. 3.17). Или произведите с помощью окна **Формат ячеек**, вкладки **Число**, которое вызывается командой значка  группы **Число** (рис. 3.17).

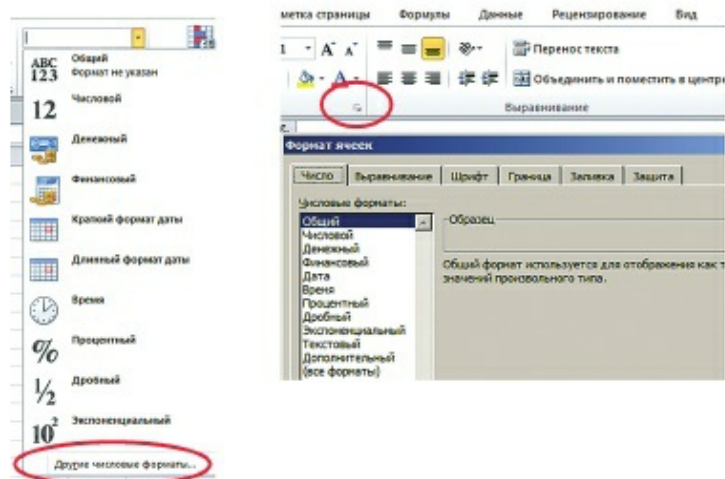


Рис. 3.17. Форматы типов данных

Классифицируем основные форматы данных (рис. 3.18):

- **Общий** – формат задается по умолчанию. Изначально в новой книге все ячейки общего формата не определены.
- **Числовой** – формат представления целых или вещественных чисел. У данного формата есть возможность добавлять/убирать число десятичных знаков, ставить разделители групп разрядов.
- **Денежный** – формат представления чисел в денежных единицах.
- **Дата** – отображает число в формате дат. В Excel все даты представлены целыми числами. Например, дата 08.01.2017 в виде числа выглядит как 42743, что означает количество дней отсчета дат в Excel от 01.01.1900. Даты раньше 01.01.1900 в Excel отражаются как текст.

	A	B	C	D	E	F
1	Числовой	Денежный	Дата	Процентный	Дробный	По маске
2	7712305,17	₸ 712 305,17	01.01.2017	100%	2/7	8 янв
3			02.01.17			

Рис. 3.18. Форматы разных данных



- **Процентный** – формат представления вещественного числа в виде процентов. Число 0 будет отображено как 0%, 1 – как 100%.
- **Дробный** – формат отображает вещественное число в виде дроби.
- **(все форматы)** – с помощью этого варианта представления можно записать свой тип отображения числа. Например, представление даты 13.01.17 выглядит как ДД.ММ.ГГ, если изменить на МММ.ГГ, то дата будет представлена как «янв.13».

ПОДУМАЙТЕ И ОТВЕЬТЕ

Зависит ли форматирование в ячейке от типа данных, которые в ней находятся? Как отформатированы данные на рис. 3.19? Если в ячейке есть текст, то по какому краю он будет отформатирован? По какому краю будут выровнены числовые данные, другие форматы?

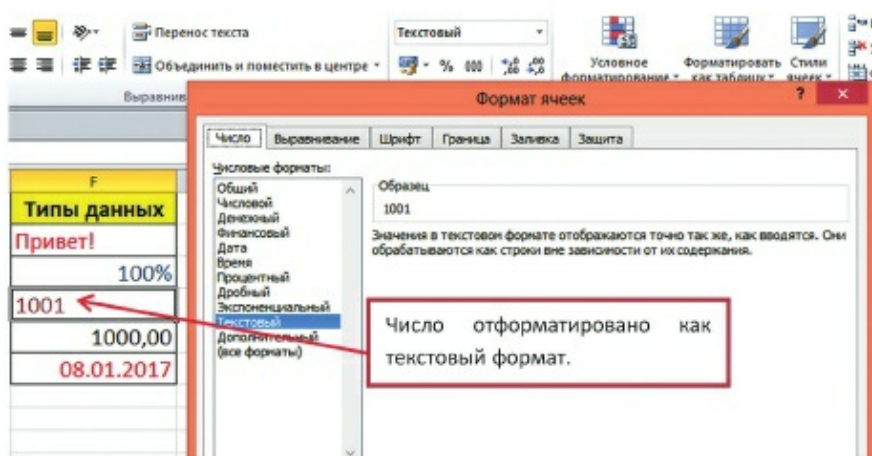


Рис. 3.19. Форматирование разных типов данных

При вводе данных в поле можно установить **автоматическую проверку** типа полей данных. Кроме того, можно установить дополнительные правила проверки вводимых данных, т. е. условие на значение, позволяющее обеспечить ввод корректных данных в поле записи. Проверка значения осуществляется при переходе к другому полю или другой записи.

Если введенное значение не соответствует заданному условию, то на экран выводится сообщение об ошибке (рис. 3.20).



F	G	H	I	J	K	L	M	N
Типы данных								
Привет!	Текстовый форм	25						
100%								
1001								
1000,00								
08.01.2017								

Рис. 3.20. Введенное значение не соответствует заданному условию

При расчетах в электронной таблице часто используют формулы.

Формулы представляют собой выражения, на основании которых выполняются вычисления в таблицах. После знака равенства «=» или «-», «+» следует **арифметическое выражение**, состоящее из цифр, адресов ячеек (ссылок) и знаков математических операций: «+» – сложение, «-» – вычитание, «/» – деление, «*» – умножение.

Вычисления сводятся к использованию формул и стандартных функций.

ЗАПОМНИТЕ



Формула — специальный инструмент в электронной таблице Excel, предназначенный для расчетов, вычислений и анализа данных.

Ввод формулы в ячейку начинается со знака «=». Это надо для того, чтобы Excel понял, что в ячейку вводится именно формула, а не данные. Формула заканчивается нажатием клавиши «Enter». Адреса ячеек в формулах используются тогда, когда нужно ввести числа, размещающиеся в этих ячейках.

ВЫПОЛНИМ ВМЕСТЕ

Рассмотрим простой пример: деление двух чисел.

Выделите произвольную ячейку, например A1. В строке формул введите знак «=», затем введите выражение 100/5. Нажмите **Enter**. В ячейке появится результат 20 (рис. 3.21). А в строке формул отобразится сама формула =100/5.



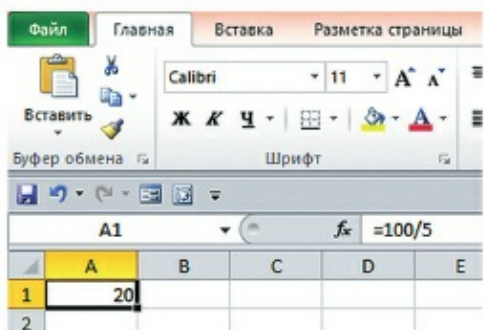


Рис. 3.21. Ввод формулы с указанными данными

ПОЭКСПЕРИМЕНТИРУЙТЕ

Поработайте с различными арифметическими операторами: сложения (+), вычитания (-), умножения (*), деления (/). Чтобы их правильно использовать, надо четко представлять их приоритет.

Пусть в ячейках A1, A2 находятся данные.

1. Выделите ячейку A3 и введите в строке формул знак «=».
2. Щелкните на ячейке A1, чтобы в строке формул отобразился адрес ячейки A1, и введите знак сложения (+).
3. Щелкните на ячейке A2, поставьте скобки для указания приоритетности, умножьте на 4 и нажмите **Enter**. Получите результат 1200 (рис. 3.22).

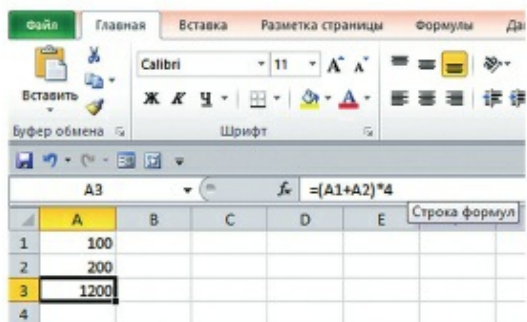


Рис. 3.22. Ввод формулы с ссылками на адреса ячеек



Ссылки в формуле выполняют роль адресов ячеек, содержимое которых используется в вычислениях. Ссылки бывают с **относительной** (A1, B2, C5) и **абсолютной адресацией** (\$A1, \$B\$2, C\$5). Диапазон обозначается A1:C8, G1:H7 и т. д.

От метода адресации ссылок зависит, что будет с ними происходить при копировании формулы из одной ячейки в другую.

По умолчанию ссылки на ячейки в формулах рассматриваются как **относительные**. Это означает, что адреса в ссылках **изменяются**.

При **абсолютной адресации** адреса ссылок при копировании формулы **не изменяются**, так что ячейка, на которую указывает ссылка, рассматривается как **постоянная**. Для указания абсолютной адресации нажимаем на клавишу **F4**, вводится символ **\$**.

Например, формулу с помощью ссылок можно записать следующим образом: $=(A1+B3)*100\%/C4$.

Уровень – Знание и понимание




1. В каких случаях применяются те или иные типы данных?
2. Когда применяется общий формат?
3. Как можно поменять формат данных?
4. Чем отличается форматирование числовых данных от текстовых? Как вы понимаете слово «формула»?
5. С какого знака начинаются формулы? Какие арифметические знаки используются в формулах?
6. Что такое *относительная адресация*?
7. Чем отличается относительная адресация от абсолютной?

Уровень – Применение



1. Используя числовой формат для десятичных чисел, разделите число на группы разрядов, поставив галочку в области **Разделитель групп разрядов (,)**, например, число 125456,456 можно представить как 125 456,456.
2. Установите для денежного формата число десятичных знаков и условное обозначение валют. Например, р. заменить на Т казахстанский (тенге). Тогда число 56,89 будет представляться как Т56,89.




Уровень – Анализ


1. Установите дробный формат для данных, представленных ниже, по образцу (табл. 3.7).

Таблица 3.7



Обычный формат	Формат дробный
5,6	5 3/5
30,3	30 3/10
25,5	25 1/2

2. В одну из ячеек листа должна вводиться дата рождения одного из учеников вашего класса (в формате **Дата**). Предусмотрите автоматическую проверку на правильность вводимых в эту ячейку значений.


Уровень – Синтез

Текстовый формат – это формат, отображающий данные точно так же, как они вводятся и обрабатываются как строки. Для примера введите число 456 и отформатируйте как текст, затем – как число. Текст выравнивается по левому краю, а число – по правому.

456	<-текст
456	<-число

После установки типа данных в поле выровняйте значение поля во вкладке **Выравнивание** диалогового окна **Формат ячейки** (рис. 3.23):



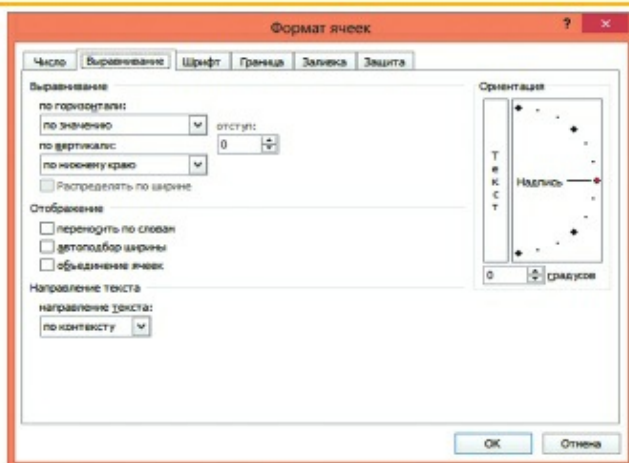


Рис. 3.23. Вкладка Выравнивание

Применяя три следующие вкладки, установите шрифт, границы, заливку ячеек таблицы и т. д.



Уровень – Оценивание

1. В одну из ячеек листа должно вводиться время начала урока (в формате **Время**). Предусмотрите автоматическую проверку на правильность вводимых в эту ячейку значений, если известно, что 1-й урок начинается в 8 часов 30 минут, а последний заканчивается в 15 часов 25 минут. Продолжительность урока 40 минут.

2. Применяя предыдущее задание, установите защиту ячейки, используя вкладку **Защита** из диалогового окна **Формат** (рис. 3.24). От кого и для чего нужно защищать информацию в ячейках электронной таблицы?



Рис. 3.24. Вкладка Защита



3.4. Условное форматирование в электронной таблице



Условное форматирование – один из самых полезных инструментов в электронной таблице, позволяет выполнять форматирование ячеек выборочно и автоматически, основываясь на их содержимом. Начнем изучение условного форматирования с проверки числовых значений *больше/меньше/равно/между* в сравнении с числовыми значениями.

Очень часто анализ данных подразумевает сравнение значений разных ячеек. В этом случае удобно ячейки, значения которых удовлетворяют определенным критериям, как-то выделить в документе. На рис. 3.25 показан рабочий документ со значениями ставок доходности по ценным бумагам по месяцам.

1	Условное форматирование на основе сравнения данных	
2		
3	месяц	ставка доходности (%)
4	январь	2,6
5	февраль	2,5
6	март	3,1
7	апрель	2,9
8	май	3,2
9	июнь	2,7
10	июль	2,4
11	август	2,3
12	сентябрь	2,7
13	октябрь	3,3
14	ноябрь	3,1
15	декабрь	2,9
16		

Рис. 3.25. Исходный документ с данными для условного форматирования



Зададим такое условное форматирование для ячеек **B4:B15**, чтобы те из ячеек, значения ставки доходности в которых меньше средней доходности 2,8, выделялись светло-красным цветом, а текст – красным цветом.

Для этого выделим диапазон ячеек **B4:B15**. Для использования условного форматирования применяем кнопку **Условное форматирование** на панели **Стили** ленты **Главная**. Далее выбираем команду **Правила выделения ячеек** и **Меньше** (рис. 3.26).

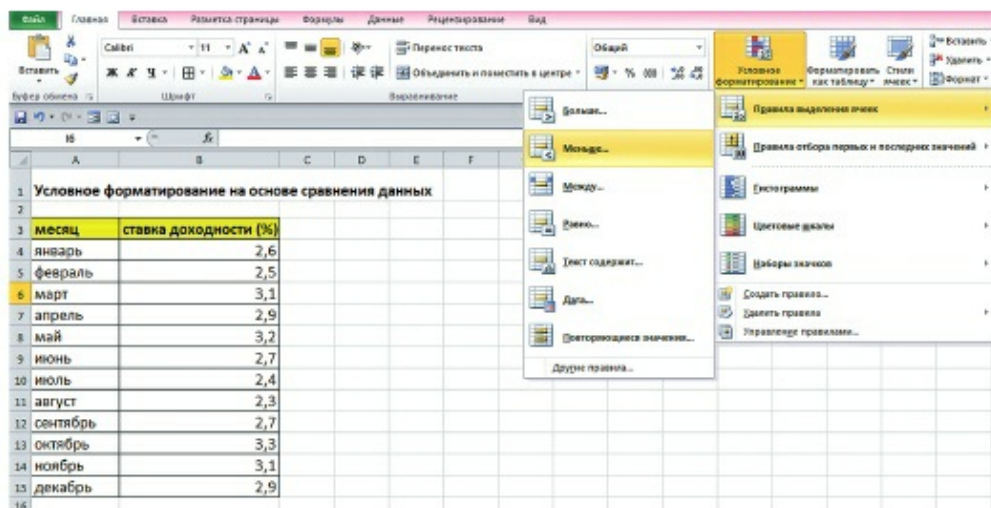


Рис. 3.26. Применение условного форматирования

Откроется диалоговое окно **Меньше** настройки параметров условного форматирования. В поле **Форматировать ячейки, которые Меньше** указываем значение, с которым следует сравнивать форматируемые ячейки (число или адрес ячейки). В данном случае указываем число 2,8. В раскрываемся списке справа выбираем один из предопределенных способов форматирования (закраска цветом ячейки и текста).

На рис. 3.27 видим, что текст отображается темно-красным цветом, а ячейка закрасена светло-красной заливкой.



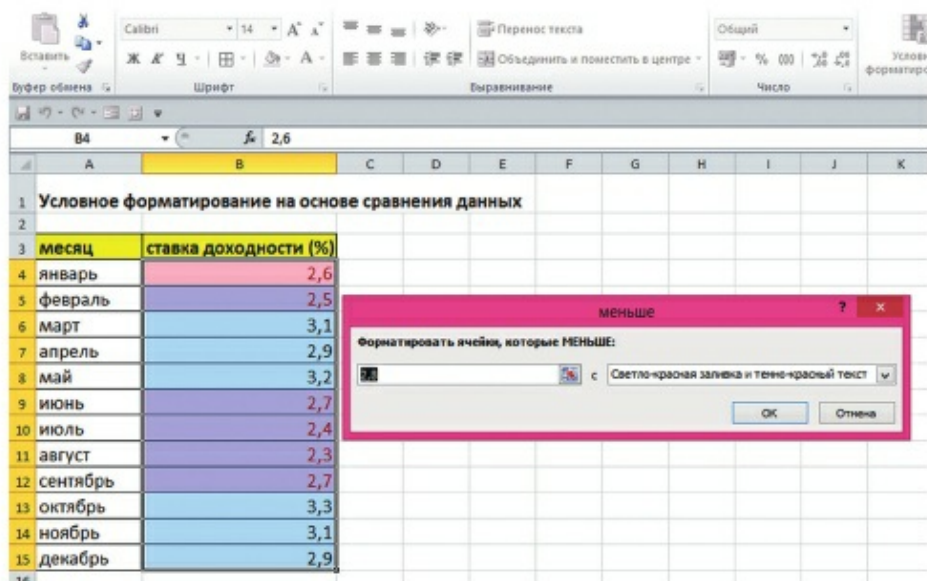


Рис. 3.27. Настройка условного форматирования

Это один из самых простых примеров выполнения условного форматирования.

ПОЭКСПЕРИМЕНТИРУЙТЕ

Чтобы лучше понять, как работает условное форматирование, выделите группу ячеек с уже введенными данными, нажмите кнопку **Условное форматирование** и посмотрите различные варианты форматирования.

При выборе **Наборы значков** (5 цветных стрелок) **Условного форматирования** в ячейках рядом с числовыми данными будут отображаться стрелки с разным направлением (рис. 3.28).

По умолчанию программа автоматически определяет больше среднего значения, меньше среднего, равное среднему значению в выделенном диапазоне.

Для формирования нужных правил форматирования можно воспользоваться пунктом **Другие правила**.

Если ячейка содержит текст, число или дату, то для применения условного форматирования необходимо выделить ячейку, нажать кнопку



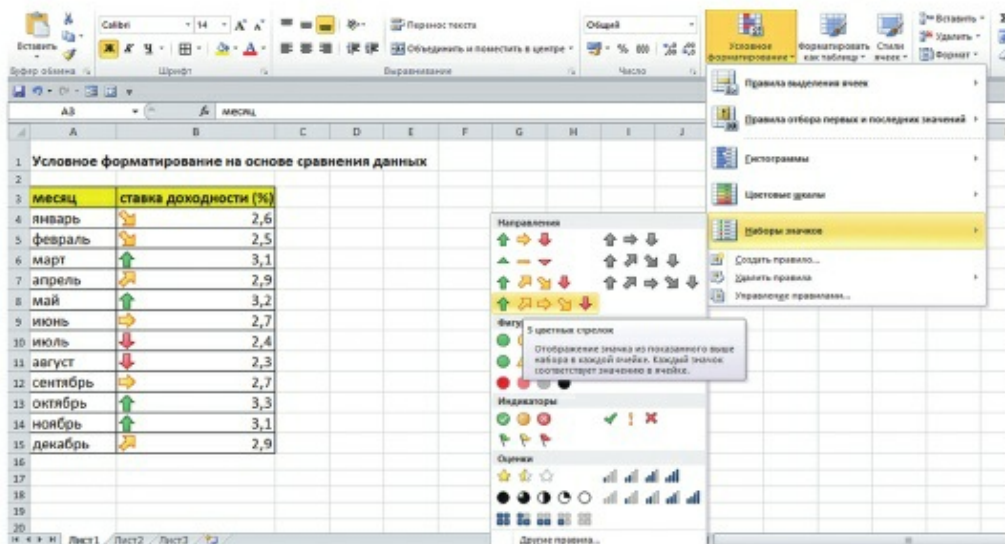


Рис. 3.28. Наборы значков

Условное форматирование и выбрать пункт Правила отбора первых и последних значений. Выберите пункт Выше среднего (рис. 3.29, 3.30).

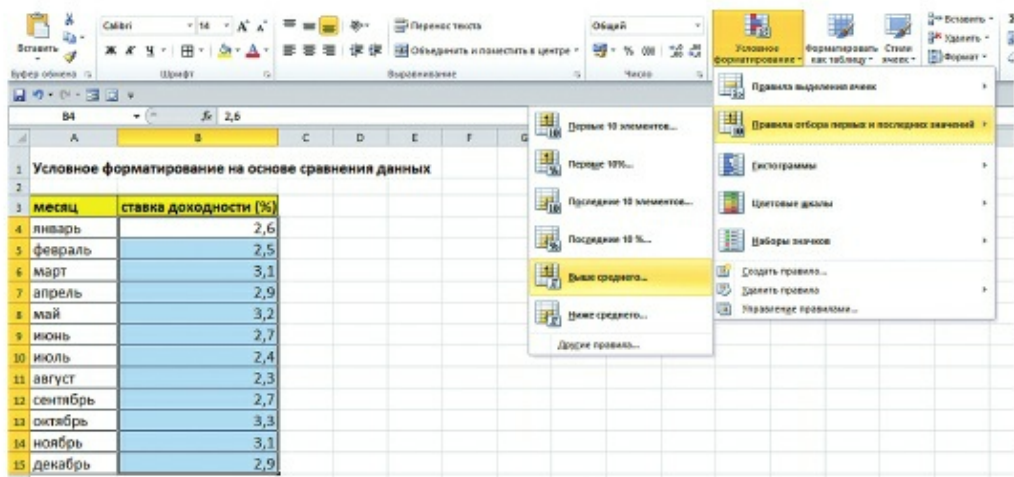


Рис. 3.29. Правила отбора первых и последних значений



Условное форматирование на основе сравнения данных

месяц	ставка доходности (%)
январь	2,6
февраль	2,5
март	3,1
апрель	2,9
май	3,2
июнь	2,7
июль	2,4
август	2,3
сентябрь	2,7
октябрь	3,3
ноябрь	3,1
декабрь	2,9

Выше среднего

Форматировать ячейки, которые находятся **ВЫШЕ СРЕДНЕГО**:

для выделенного диапазона с:

OK Отмена

Рис. 3.30. Правила отбора первых и последних значений **Выше среднего**

Уровень – Знание и понимание



1. С какой целью применяется условное форматирование?
2. Что нужно сделать с диапазоном ячеек, чтобы применить условное форматирование?
3. Каким образом можно вставить **Значки** в ячейки?
4. Какие еще способы применяются для условного форматирования?
5. Можно ли применить условное форматирование для определения ячеек с ошибочными записями?

Уровень – Применение



1. Создайте таблицу «Прогноз погоды». Введите температуру воздуха на неделю (с 20 по 27 февраля). Задайте условное форматирование таким образом, чтобы все ячейки с отрицательными значениями закрашивались в красный цвет.
2. Составьте таблицу расходов за неделю текущего месяца в тенге (рис. 3.31).



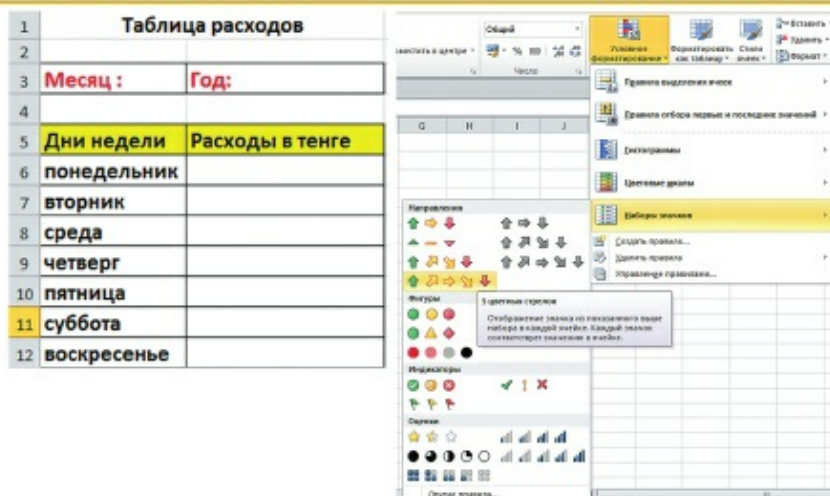


Рис. 3.31. Таблица расходов

Используя **Условное форматирование – Набор значков**, выберите понравившийся значок и вставьте туда, где будут введены данные расходов. Что отображают 5 разных значков в ячейках?

3. В списке учащихся на рис. 3.32 ячейки с оценкой «5» заливаются светло-красным цветом, «4» – желтым, «3» – зеленым, а «2» – красным цветом. Шрифт – пользовательский формат.

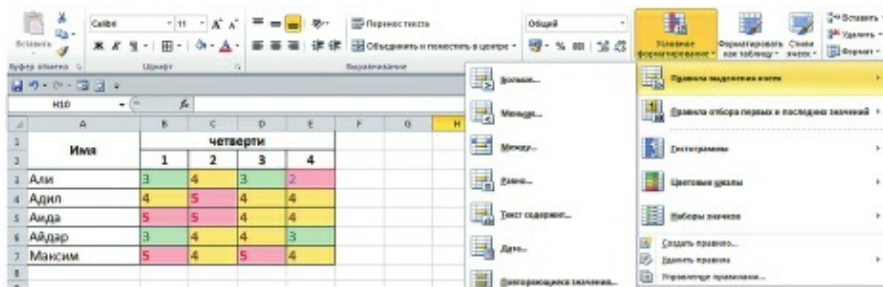


Рис. 3.32. Условное форматирование оценок





Уровень – Анализ, синтез

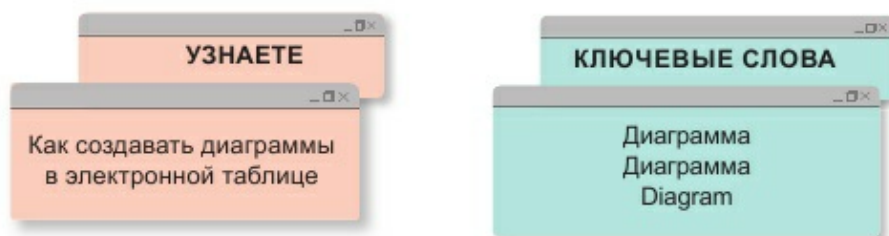
Выделите с помощью условного форматирования температурные отклонения в диапазоне значений в вашем регионе.

На рис. 3.33 показан образец температурных данных с условным форматированием, в котором для обозначения высоких, средних и низких значений использовалась цветовая шкала.

	A	B	C	D
1	Температура в городе Алматы			
2	температура	март	апрель	май
3	высокая	15	20	25
4	средняя	8	7	20
5	низкая	4	5	10
6				

Рис. 3.33. Температурные отклонения

3.5. Графическое представление табличных данных



Диаграммы – самый наглядный способ представления данных. Диаграмму качества знаний нагляднее показать графически, чем рассказать словами, что в классе из 25 человек 2 отличника, 12 хорошистов, а остальные – троечники. Диаграмма облегчает восприятие информации.

ЗАПОМНИТЕ



Диаграмма – это визуальное представление числовых значений.



Диаграммы бывают различного типа (рис. 3.34):

- гистограмма;
- круговая;
- линейчатая;
- с областями;
- график;
- точечная;
- кольцевая;
- лепестковая;
- пузырьковая.

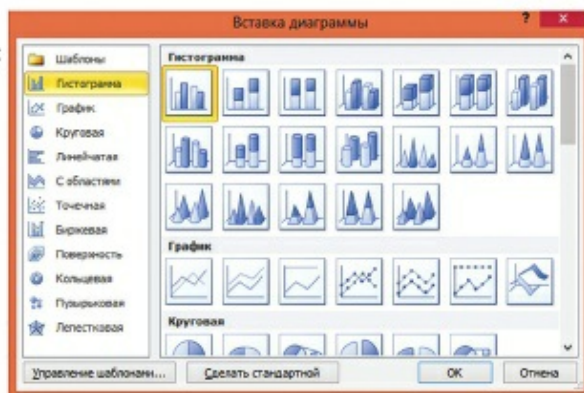


Рис. 3.34. Типы диаграмм

Каждый тип диаграммы подразделяется на **виды**. Например, гистограмма с группировкой показывает сравнение значений по категориям, гистограмма с накоплением – вклад каждого значения в общую сумму (с нарастающим итогом), нормированная гистограмма с накоплением – процент вклада каждого значения в общую сумму. Кроме того, можно выбрать форму элементов гистограммы: плоская (обычная), объемная, цилиндрическая, коническая, пирамидальная.

СОЗДАДИМ ВМЕСТЕ

Рассмотрим **технология построения диаграмм** на примере проведения графического анализа недельной продажи товара (газет) тремя продавцами.

Для этого построим диаграмму типа **Гистограмма** обычного вида, с отображением значений различных категорий.

1. Создайте таблицу с данными, как на рис. 3.35.

	Е9	f _x						
	A	B	C	D	E	F	G	H
1	Продажа газет							
2		Пн	Вт	Ср	Чт	Пт	Сб	Вс
3	Продавец 1	34	35	33	30	36	39	42
4	Продавец 2	28	29	21	24	26	24	24
5	Продавец 3	25	22	24	23	25	29	30

Рис. 3.35. Таблица продажи газет



2. Выделите область A2:H5 и выберите гистограмму вида с группировкой из группы Диаграммы на вкладке Вставка (рис. 3.36).

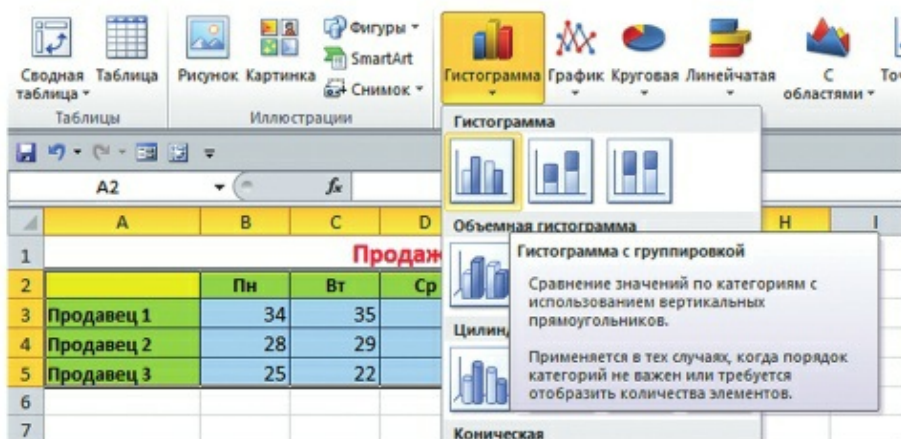


Рис. 3.36. Вставка диаграммы

3. Получите гистограмму, как на рис. 3.37.

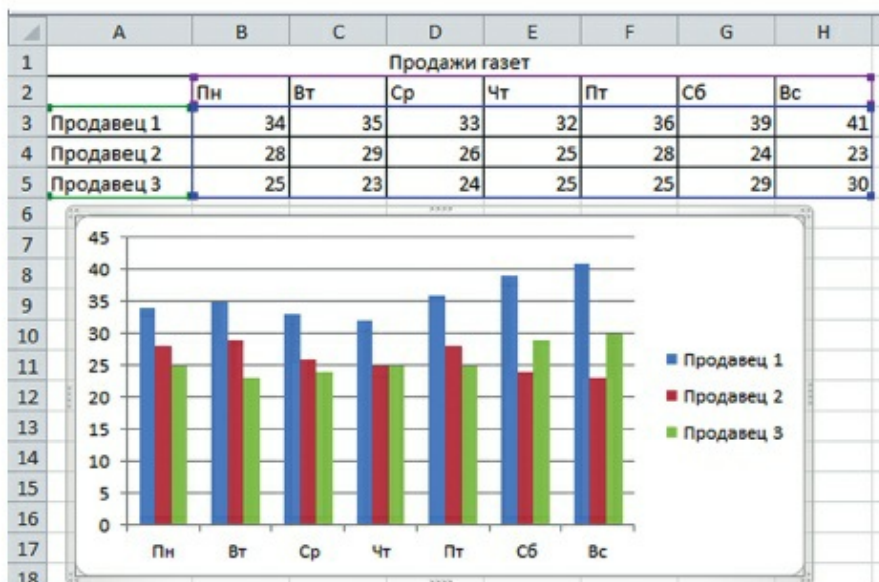


Рис. 3.37. Гистограмма



ПОЭКСПЕРИМЕНТИРУЙТЕ

При выделении диаграммы появятся дополнительные вкладки для работы с диаграммой: **Конструктор** и **Макет** (рис. 3.38).

Диаграмме необходимо задать разные параметры – это заголовок, оси, линии сетки, легенда, подписи данных, таблица данных для визуализации числовых данных.

Область диаграммы, где расположены условные обозначения (цвет и подписи Продавец №), называется **Легенда**.

Попробуйте с помощью вкладки **Конструктор** поменять стиль, тип диаграммы и переместить диаграмму на отдельный лист.

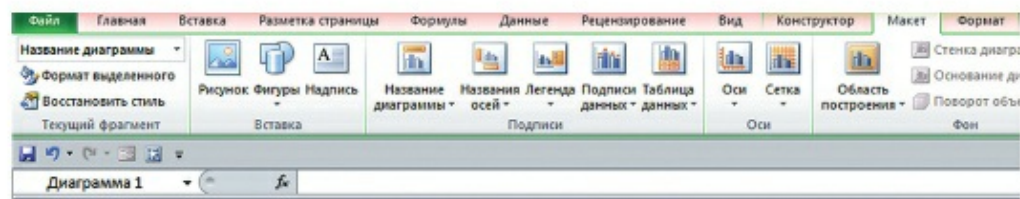


Рис. 3.38. Дополнительные вкладки **Конструктор** и **Макет**

С помощью вкладки **Макет** вставьте название диаграммы, название осей, переместите легенду в другое место, установите подписи данных (рис. 3.39).

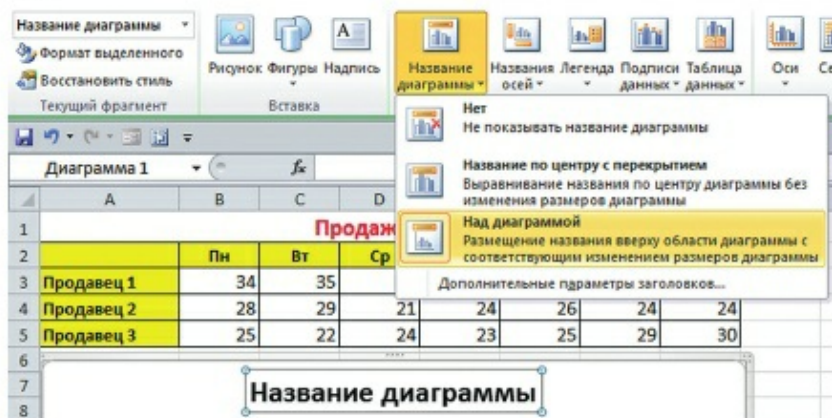


Рис. 3.39. Вставка названия диаграммы

Если выделите двойным щелчком левой кнопкой мыши что-либо на диаграмме, например ось, то можно с помощью команды **Формат оси** изменить ее параметры (рис. 3.40). Попробуйте изменить параметры оси.

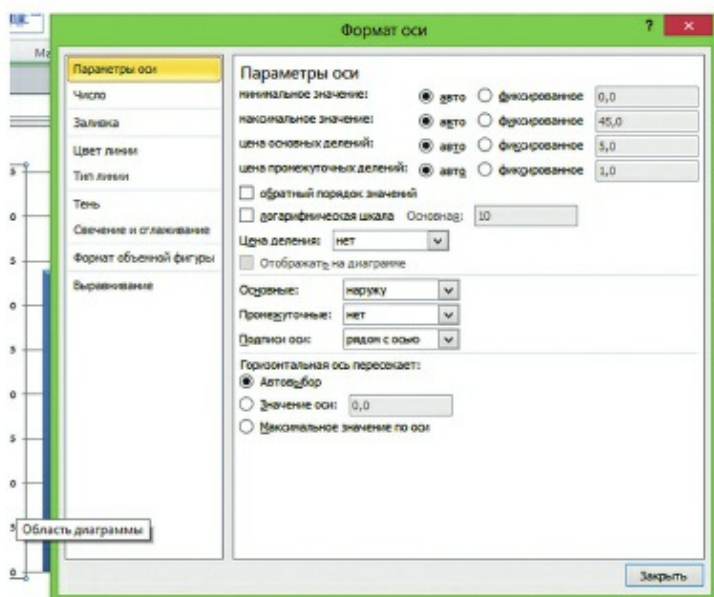


Рис. 3.40. Формат оси

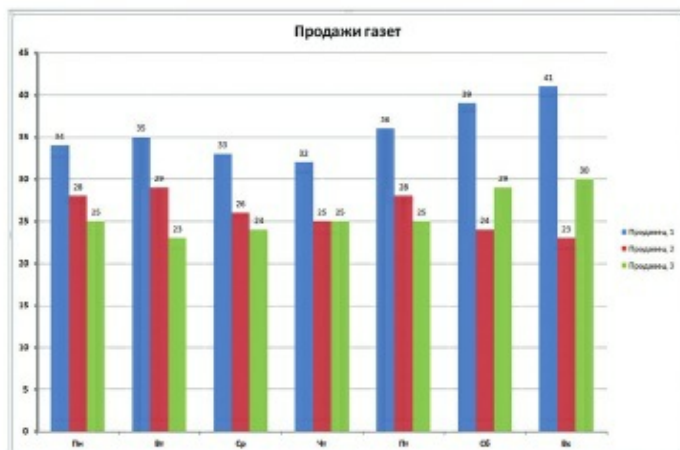


Рис. 3.41. Готовая диаграмма



В результате получите диаграмму, которая показывает, сколько товара продавал каждый продавец ежедневно (рис. 3.41).

Уровень – Знание и понимание



1. Какие типы диаграмм существуют?
2. Почему у одной диаграммы имеется несколько видов?
3. Можно ли поменять подписи данных у готовой диаграммы? Каким образом?
4. Как можно поменять тип диаграммы?
5. Какие дополнительные вкладки появляются в электронной таблице при выделении диаграммы?
6. Как настроить диаграмму, чтобы она могла максимально визуализировать данные?
7. Как удалить диаграмму?
8. Объясните, какие виды графиков изображены на рис. 3.42. Почему изображены два графика в одной координатной плоскости? Какой график наиболее информативный?

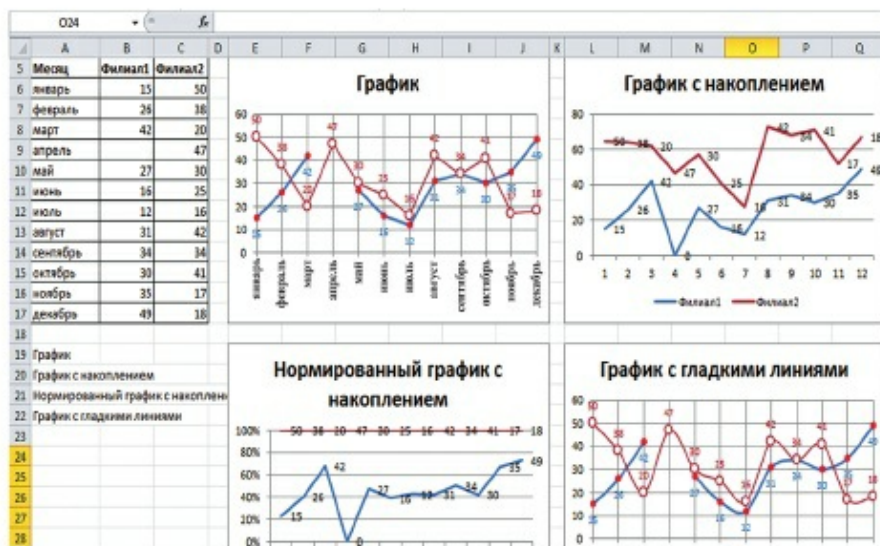


Рис. 3.42. Разные виды графиков

Уровень – Применение



Номера 1, 2, 3, 4, 5, 6, 7 на рис. 3.42 указывают на названия элементов диаграммы. Выпишите их и объясните, с какой целью эти элементы диаграммы используются.



Рис. 3.43. Элементы диаграммы



Уровень – Анализ



Постройте круговую диаграмму по данным из табл. 3.8.

Таблица 3.8

	А	В
1	Площадь континентов	
2	Название континента	Площадь (млн. кв. км)
3	Европа	10,00
4	Азия	43,40
5	Северная Америка	24,71
6	Южная Америка	17,84
7	Австралия	7,66
8	Антарктида	14,10
9	Африка	30,30
10	Всего:	148,01





Уровень – Синтез

Постройте круговую диаграмму распределения людей по группам крови, если людей с группой крови О в мире около 46%, с группой крови А около 34%, с группой крови В приблизительно 16,5%, а с самой редкой группой крови АВ всего 3,5%.



Уровень – Оценивание

Проблема экологии городов – это, в первую очередь, проблема уменьшения выбросов в окружающую среду различных загрязнителей.

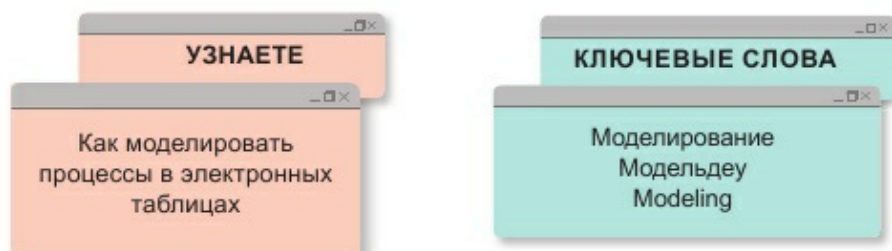
Для разложения в природной среде бумаги требуется до 10 лет, консервной банки – до 90 лет, фильтра от сигареты – 100 лет, полиэтиленового пакета – до 200 лет, пластмассы – до 500 лет, стекла – до 1000 лет.

Заполните таблицу в электронной таблице Excel, задайте ей имя.

Постройте 3 различные диаграммы и выберите наиболее информативную.

Помните об этом, не бросайте куда попало полиэтиленовые пакеты или бутылки.

3.6. Моделирование процессов в электронных таблицах



В своей деятельности человек очень часто использует модели, т. е. он создает образ того объекта, явления или процесса, с которым ему предстоит иметь дело.



Существуют разные виды моделирования: информационное, математическое, компьютерное, натурное, физическое, структурное и т. д. Любая модель создается и изменяется на основе имеющейся у человека информации. Модель создается для какого-либо объекта, процесса или явления. В этом случае говорят, что человек создает информационную модель объекта.

ЗАПОМНИТЕ



Информационная модель – это описание моделируемого объекта на одном из языков кодирования информации.

Прототип – это объект, процесс или явление, на основе которого создается модель.

Информационная модель содержит существенные для исследователя сведения об объекте.

ЗАПОМНИТЕ



Моделирование – это процесс построения моделей для исследования и изучения объектов, процессов или явлений.

Создание моделей для исследования экономических, экологических, биологических, физических процессов, развития Вселенной является актуальной задачей в моделировании процессов. Практически во всех науках о природе, живой и неживой, об обществе построение и использование моделей является мощным орудием познания.

В настоящее время для построения различных по форме моделей используется компьютер. Поэтому нас интересует, как создать компьютерную модель.

На рис. 3.44 представлена схема этапов моделирования при исследовании различных процессов.



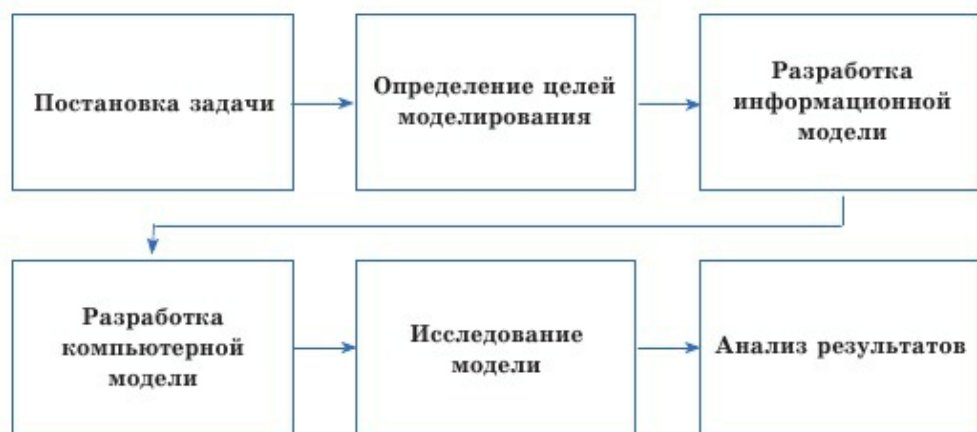


Рис. 3.44. Этапы моделирования

В качестве примера рассмотрим моделирование экологических процессов, связанных с загрязнением окружающей среды, в электронной таблице Excel.

ВЫПОЛНИМ ВМЕСТЕ

1. Постановка задачи.

В результате сброса промышленных стоков возрос уровень загрязнения реки. Каким он будет через сутки, двое, трое и т. д.? Когда он будет допустимым, если известно, что за сутки он уменьшается в определенное количество раз?

Провести исследование экологической модели при значениях параметров, представленных в табл. 3.9:

Таблица 3.9

Вещество	C_0 (мг/л)	$C_{\text{доп}}$ (мг/л)	K
Свинец	5	0,03	1,12
Мышьяк	1,5	0,05	1,05
Фтор	0,2	0,05	1,01

Условные обозначения:

C_0 – начальная концентрация вредных примесей;

$C_{\text{доп}}$ – предельно допустимая концентрация вредных примесей;

K – коэффициент суточного уменьшения концентрации вредных примесей.



2. Определение целей модели.

Объект исследования – концентрация вредных веществ. Для построения таблицы значений используем электронную таблицу Excel.

3. Разработка модели.

А) Построение математической модели:

– определение аргументов, результатов и связи между ними;

– определение математической модели задачи:

$$C_1 = C_0 / K$$

$$C_2 = C_1 / K = C_0 / K^2$$

$$C_3 = C_2 / K = C_0 / K^3$$

.....

$$C_n = C_0 / K^n \text{ или } C_n = C_{n-1} / K;$$

– определение целого значения n .

В формуле очередное значение C_i вычисляете по предыдущему значению C_{i-1} .

В) Построение компьютерной модели (рис. 3.45):

	A	B	C
1	Экологическая модель		
2	Исходные данные		
3	5	C ₀ - начальная концентрация вредных примесей	
4	0,03	C _д - допустимая концентрация	
5	1,12	K - коэффициент суточного уменьшения концентрации	
6	Расчетная таблица		
7	n	K	C₀
8	0	1,12	5,00
9			
10	=A8+1	=B8	=C8/B9
51	43	1,12	0,04
52	44	1,12	0,03

Рис. 3.45. Экологическая модель в электронной таблице

- представьте модель в электронной таблице;
- введите последовательно исходные данные, а также начальные данные и формулы для последующих вычислений;



- получите таблицу, с помощью которой произведете вычисления, скопировав формулы с помощью **Маркера заполнения** до нужного результата;
- проведите компьютерное тестирование модели, изменяя исходные данные веществ: *свинец, мышьяк, фтор*;
- установите **Формат ячеек** для числовых данных, установив *число десятичных знаков – 2 после запятой*;
- выполните форматирование элементов таблицы;
- используйте условное форматирование, закрасив ячейку и число уровня загрязненности при достижении допустимой нормы в другой цвет;
- постройте график функции по данным столбца начальной концентрации вредных примесей (рис. 3.46).

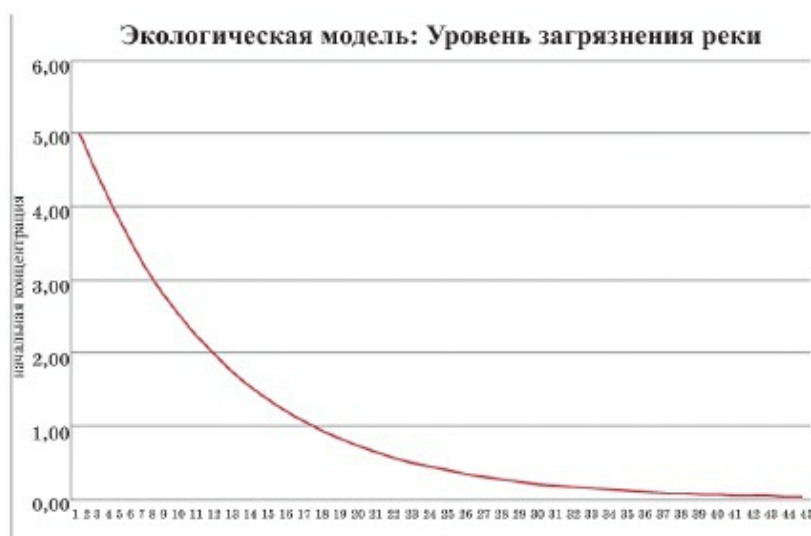


Рис. 3.46. График функции экологической модели

4. Исследование модели.

Модель построена в соответствии с формулами, представленными в условии задачи.

5. Анализ результатов.

Сравниваете значения столбца **C** со значениями ячейки **A4** допустимой нормой **Сдоп**. При этом в ячейке должны получить значение **C₁**. Это значение должно быть меньше или равно значению ячейки **A4**.



Если такая ячейка существует, то определяем соответствующее значение ячейки **AI**. Это будет номер суток. Если такой ячейки не окажется, следует продолжить копирование ячеек до получения нужного результата.

Экологическая интерпретация полученного решения.

Через 44 дня уровень загрязненности достигнет допустимой нормы.

Уровень – Знание и понимание



1. Что такое *моделирование*?
2. Из каких этапов состоит моделирование?
3. Чем отличается математическая модель от компьютерной?
4. Что такое *тестирование модели*? Как проводится компьютерное тестирование?
5. Почему необходимо проводить анализ результатов?
6. Для чего может быть предназначена каждая из моделей:
 - эскизы костюмов к театральному спектаклю;
 - макет книги или журнала;
 - глобус;
 - географический атлас;
 - модель (макет) строения молекулы воды;
 - уравнение химической реакции;
 - генеалогическое (родословное) древо семьи;
 - макет скелета человека;
 - формула определения площади квадрата со стороной h : $S = h^2$;
 - расписание движения поездов;
 - игрушечная модель автомобиля;
 - схема метрополитена;
 - график зависимости высоты тела, брошенного под углом к горизонту, от времени полета;
 - оглавление книги?

Уровень – Применение, анализ



Дана геометрическая модель (рис. 3.47), у которой длина стороны картонного листа обозначена a , а длина выреза – b . Вычислите объем геометрической модели.

Математическая модель

Расчетные формулы:

$c = a - 2b$ – длина стороны дна;

$S = c^2$ – площадь дна;

$V = Sb$ – объем.

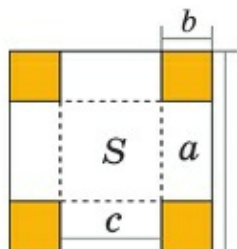


Рис. 3.47. Геометрическая модель



Компьютерная модель

Для моделирования используйте среду электронной таблицы, в которой информационная и математическая модели объединяются в таблицу, имеющую три области:

- исходные данные – управляемые параметры;
- промежуточные расчеты;
- результаты.

Задание

- Заполните область исходных данных по предложенному образцу (рис. 3.48).
- Составьте таблицу расчета со столбцами в электронной таблице, используя расчетные данные математической модели.
- Какие типы данных будут использоваться?
- Выполните форматирование данных в ячейках.
- Как можно отобразить данные из таблицы графически?

Задача о склеивании коробки

Исходные данные	
Длина стороны	40
Шаг увеличения выреза	1

Расчет

Промежуточные расчеты			Результаты
Длина выреза	Длина стороны дна	Площадь дна	Объем
1			

Рис. 3.48. Образец в электронной таблице

**Уровень – Синтез**

Необходимо покрасить стены кухни. Сколько потребуется банок краски, если известно, что:

- размеры кухни $405 \times 310 \times 285$ см³;
- 88% площади стен занимает кафельная плитка;
- 1 банка краски предназначена для покраски площади 5 м²?



Уровень – Оценивание

Вспомните, как был открыт закон всемирного тяготения. Легенда гласит, что Исаак Ньютон сидел под яблоней, когда на него упало яблоко, подсказавшее ему правильное решение. Как известно, закон всемирного тяготения говорит о том, что сила притяжения двух тел прямо пропорциональна их массам и обратно пропорциональна квадрату расстояния между ними:

$$F = \gamma \cdot M \cdot m / r^2.$$

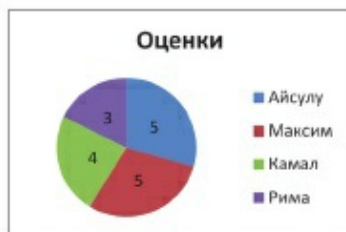
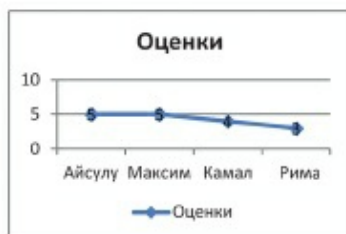
1. Является ли данная формула математической моделью такого физического явления, как тяготение?
2. Что нужно сделать, чтобы она стала адекватной моделью притяжения Земли и висящего на дереве яблока?

Электронные таблицы позволят создавать диаграммы нескольких типов, основными из которых являются графики, круговые диаграммы, гистограммы.

Графики используются для отображения зависимости значений одной величины (функции) от другой (аргумента). Они позволяют отслеживать динамику изменения данных.

Круговые диаграммы используются для отображения величин (размеров) частей некоторого целого; в них каждая часть целого представляется как сектор круга, угловой размер которого прямо пропорционален величине (размеру) части.

Гистограммы (столбчатые диаграммы) используются для сравнения нескольких величин. В них величины отображаются в виде вертикальных или горизонтальных столбцов. Высоты (длины) столбцов соответствуют отображаемым значениям величин.



ТЕСТОВЫЕ ЗАДАНИЯ К РАЗДЕЛУ 3

Из предложенных альтернативных ответов выберите только один верный. Каждый правильный ответ оценивается в 1 балл. Всего 15 баллов.

1. Текстовый редактор – это:

- а) программа для работы преимущественно с текстовой информацией;
- б) программа для создания мультфильмов;
- в) программа для обработки изображений;
- г) программа для управления ресурсами ПК при создании рисунков;
- д) художник-график.

2. Какая операция нарушает признак, по которому подобраны все остальные операции из приводимого ниже списка:

- а) сохранение текста;
- б) форматирование текста;
- в) перемещение фрагмента текста;
- г) удаление фрагмента текста;
- д) копирование фрагмента текста?

3. Продолжите логическую цепочку:

Библиотека – каталог; Книга – оглавление; Текстовый редактор – ?:

- а) окно;
- б) текст;
- в) меню;
- г) курсор;
- д) рабочее поле.

4. Электронная таблица – это:

- а) прикладная программа, предназначенная для обработки структурированных в виде таблицы данных;
- б) прикладная программа обработки кодовых таблиц;
- в) устройство персонального компьютера, управляющее процессом обработки данных в табличной форме;
- г) системная программа, управляющая ресурсами персонального компьютера при обработке таблиц;
- д) экранная форма представления записи базы данных.

5. Диапазон в электронной таблице – это:

- а) все ячейки одной строки;
- б) все ячейки одного столбца;
- в) множество допустимых значений;



г) совокупность клеток, образующих в таблице область прямоугольной формы;

д) область таблицы произвольной формы.

6. Сколько ячеек электронной таблицы в диапазоне A2:B4:

а) 16;

б) 2;

в) 8;

г) 4;

д) 6?

7. Электронная таблица представляет собой:

а) совокупность нумерованных строк и поименованных с использованием букв латинского алфавита столбцов;

б) совокупность поименованных с использованием букв латинского алфавита строк и нумерованных столбцов;

в) совокупность совокупность пронумерованных строк и столбцов;

г) совокупность строк и столбцов, именуемых пользователем произвольным образом.

8. Строки электронной таблицы:

а) именуется пользователем произвольным образом;

б) обозначаются буквами русского алфавита А ... Я;

в) обозначаются буквами латинского алфавита;

г) нумеруются.

9. Столбцы электронной таблицы:

а) обозначаются буквами латинского алфавита;

б) нумеруются;

в) обозначаются буквами русского алфавита А ... Я;

г) именуется пользователем произвольным образом.

10. Для пользователя ячейка электронной таблицы идентифицируется:

а) путем последовательного указания имени столбца и номера строки, на пересечении которых располагается ячейка;

б) адресом машинного слова оперативной памяти, отведенного под ячейку;

в) специальным кодовым словом;

г) именем, произвольно задаваемым пользователем.

11. Форма графического представления числовых значений, позволяющая облегчить восприятие и интерпретацию числовых данных, называется:

а) чертежом;



- б) блок-схемой;
- в) картой;
- г) таблицей;
- д) диаграммой.

12. Диаграмма, отдельные значения которой представлены точками в декартовой системе координат, называется:

- а) гистограммой;
- б) линейчатой диаграммой;
- в) круговой диаграммой;
- г) объемной диаграммой;
- д) точечной диаграммой.

13. Гистограмма – это диаграмма:

- а) в которой отдельные значения представлены вертикальными столбцами различной высоты;
- б) из параллелепипедов, размещенных вдоль оси X;
- в) в которой используется система координат с тремя координатными осями, что позволяет получить эффект пространственного представления рядов данных.
- г) в которой отдельные значения представлены полосами различной длины, расположенными горизонтально вдоль оси X;
- д) представленная в виде круга, разбитого на секторы.

14. Моделирование – это процесс:

- а) замены реального объекта (процесса, явления) его образом, создаваемым для решения конкретной задачи средствами языка в широком смысле слова;
- б) демонстрации моделей одежды в салоне мод;
- в) неформальной постановки конкретной задачи;
- г) замены реального объекта (процесса, явления) другим материальным или идеальным объектом, отражающим все без исключения признаки данного объекта;
- д) выявления существенных признаков рассматриваемого объекта.

15. К информационным моделям, описывающим организацию учебного процесса в школе, можно отнести:

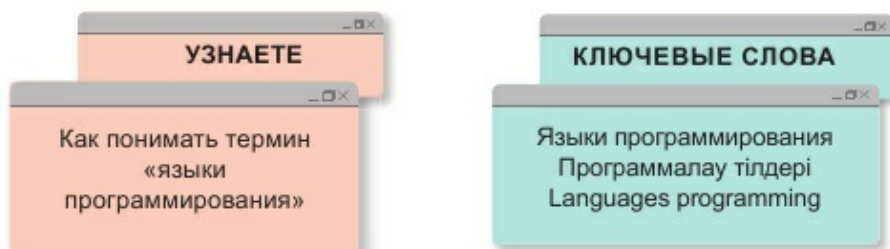
- а) классный журнал;
- б) расписание уроков;
- в) список учащихся школы;
- г) перечень учебников;
- д) список наглядных учебных пособий.



Раздел 4

ПРОГРАММИРОВАНИЕ РЕШЕНИЙ

4.1. Языки программирования



ПОДУМАЙТЕ И ОТВЕЬТЕ

В какой форме хранится вся информация в компьютере, в том числе и компьютерные программы?

Любые языки программирования относятся к искусственным языкам, созданным специально для разработки программ.

ЗАПОМНИТЕ



Язык программирования – формальная знаковая система, предназначенная для записи компьютерных программ. Язык программирования определяет набор определяющих внешний вид программы и действия правил, выполняемых исполнителем (обычно – компьютером) под ее управлением.

Существуют различные классификации языков программирования. Согласно одной из них все языки программирования делятся на две группы: **процедурные и непроцедурные** (рис. 4.1).





Рис. 4.1. Языки программирования

Процедурные (или алгоритмические) программы – это система предписаний для решения конкретной задачи. Здесь роль компьютера сводится к механическому выполнению этих предписаний.

Процедурные языки разделяют на языки **низкого** и **высокого** уровня. Разные типы процессоров имеют разные наборы команд. Если язык программирования ориентирован на конкретный тип процессора и учитывает его особенности, то он называется **языком программирования низкого уровня**. Он занимает меньше места в памяти и работает быстрее. С помощью этого языка создают системные программы, драйверы (программы для управления устройствами компьютера) и другие виды программ.

Все языки низкого уровня ориентированы на определенный тип компьютера. Для всех языков высокого уровня общим является то, что ориентированы они не на систему команд того или иного компьютера, а на систему операторов, характерных для записи определенного класса алгоритмов.

ПРОАНАЛИЗИРУЙТЕ И ОЦЕНИТЕ

Мы рассмотрели языки низкого уровня. Попробуйте проанализировать языки программирования высокого уровня по рис. 4.1, 4.2.





Рис. 4.2. Языки низкого и высокого уровней

Языки программирования строятся по тем же законам, что и человеческие: у них есть свой алфавит, свои слова (служебные), свои правила написания. **Программы**, написанные на таких языках, состоят из последовательности предложений, которые называются **командами**, или **операторами**.

Как нам известно, компьютер понимает только один язык – **язык машинных кодов**. Программы на машинном языке – очень длинные последовательности единиц и нулей – являются машиннозависимыми. Для каждой электронно-вычислительной машины (ЭВМ) необходимо было составлять свою программу.

Трансляторы – это программы или технические средства, переводящие информацию с языка, понятного человеку, на язык, понятный устройству. Трансляторы бывают двух типов: **компиляторы** и **интерпретаторы** (рис. 4.3).

Любой транслятор решает четыре основные задачи (рис. 4.3):

- анализирует транслируемую программу, ищет ошибки, исправляет их сам или выдает сообщения об ошибках;
- если ошибок нет, транслятор генерирует выходную программу на машинном языке;
- оптимизирует (улучшает) выходную программу;
- распределяет машинную память для выходной программы.



На английском языке **Translator** – программа-переводчик. Она преобразует программу, написанную на одном из языков высокого уровня, в программу, состоящую из машинных команд.

КОМПИЛЯТОРЫ

На английском языке **Compiler** – читает всю программу целиком, делает ее перевод и создает законченный вариант программы на машинном языке, который затем и выполняется. Примеры компиляторов: Free Pascal Compiler, C, C++ и др.

ИНТЕРПРЕТАТОРЫ

На английском языке **Interpreter** – переводит и выполняет программу строка за строкой тут же, не порождает на выходе программу на машинном языке. Примеры интерпретаторов: Java Script, VB Script и др. Недостатком является более низкая скорость выполнения программ, чем в машинных кодах.

Рис. 4.3. Трансляторы

Уровень – Знание и понимание



1. Что такое «язык программирования»?
2. Почему языки программирования делятся на машиннозависимые и машиннонезависимые?
3. Как вы понимаете термин «трансляторы»?
4. Чем отличаются компиляторы от интерпретаторов?
5. Какие четыре основные задачи решает транслятор?

Уровень – Применение



Классифицируйте по уровням языков программирования и объясните свои действия по представленной ниже схеме (рис. 4.4).



Рис. 4.4. Классификация по уровням программирования





Уровень – Анализ и синтез



Организируйте групповую поисковую деятельность с помощью учебников, справочников, Интернета для проведения мини-исследования «Еще больше о языках программирования». Информацию оформите в виде таблицы в текстовом процессоре.



Уровень – Оценивание

Совместная работа над интеллект-картой для лучшего усвоения темы. Для работы используйте полученные данные мини-исследования из предыдущего задания. Систематизируйте информацию в виде интеллект-карты, используя сервис визуализации mindomo. Описание и инструкция по работе с сервисом <https://www.mindomo.com>.

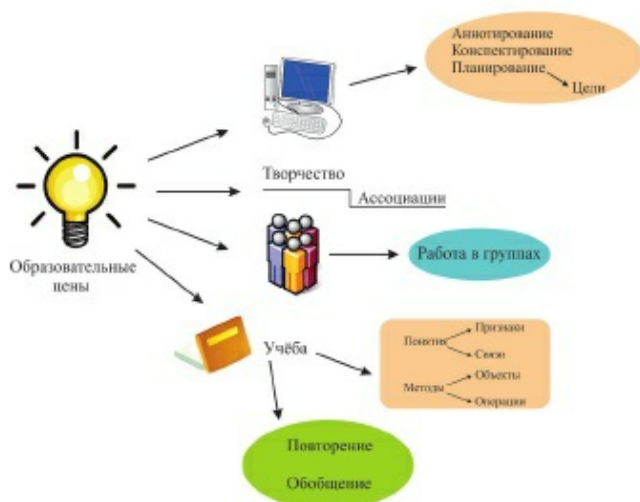
Визуализация в общем смысле – метод представления информации в виде оптического изображения (рисунков и фотографий, графиков, диаграмм, структурных схем, таблиц, карт и т. д.).



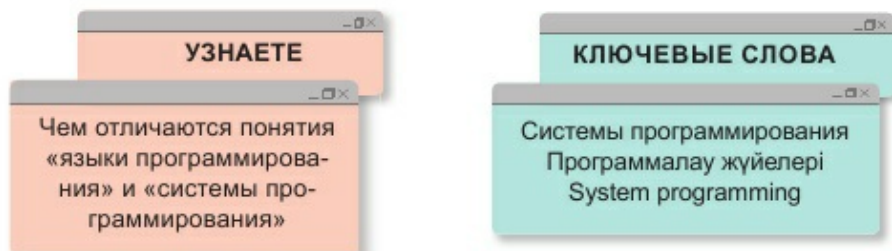
Интеллект-карта – техника представления любого процесса, события, мысли или идеи в систематизированной визуальной форме. Примеры создания таких карт можно встретить в научных трудах. Широкое применение техники Mind Maps началось благодаря английскому психологу Тони Бьюзену.

Термин Mind Maps на русский язык переводится по-разному: «интеллект-карты», «карты ума», «карты мыслей», «карты мышления», «ментальные карты», «карты памяти» и т. д. Суть интеллект-карт одна – наглядное отображение смысловых, причинно-следственных, ассоциативных связей между понятиями, событиями, идеями.





4.2. Системы программирования



Машинный код, который понимает компьютер, удобен только для него. Чтобы улучшить понимание между компьютером и пользователем, были разработаны системы программирования.

С появлением систем программирования пользователь освободился от необходимости написания программ, решения своих задач на неудобном для него языке низкого уровня – машинных команд.

Система программирования предоставляет возможность использовать специальные языки более высокого уровня. Для каждого из таких языков, называемых **входными**, или **исходными**, система программирования имеет программу. Эта программа осуществляет автоматический перевод (трансляцию) текстов программы с входного языка на язык машины.



С помощью системы программирования разрабатывают **новые программы** на конкретном языке программирования. Ядро системы программирования составляет язык.

ЗАПОМНИТЕ



Системой программирования называется комплекс программ, предназначенный для автоматизации программирования задач на ЭВМ (электронно-вычислительной машине).

Современные системы программирования обычно предоставляют пользователям мощные и удобные средства разработки программ. В составные системы программирования входят (рис. 4.5):

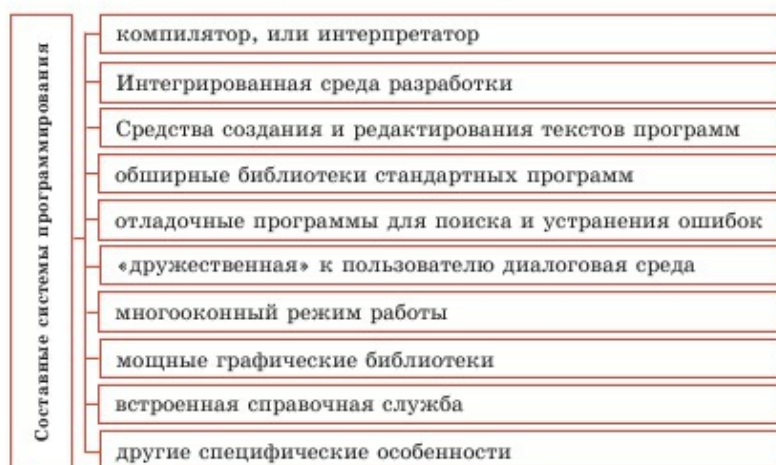


Рис. 4.5. Составные системы программирования

В настоящее время разработка любого системного и прикладного программного обеспечения осуществляется с помощью систем программирования, которые, как правило, включают в себя:

- **текстовый редактор (Edit)**, осуществляющий функции записи и редактирования исходного текста программы;
- **загрузчик программ (Load)**, позволяющий выбрать из директория нужный текстовый файл программы;
- **запускатель программ (Run)**, осуществляющий процесс выполнения программы;



- **компилятор (Compile)**, предназначенный для компиляции или интерпретации исходного текста программы в машинный код с диагностикой синтаксических и семантических (логических) ошибок;
- **отладчик (Debug)**, выполняющий сервисные функции по отладке и тестированию программы;
- **диспетчер файлов (File)**, предоставляющий возможность выполнять операции с файлами: сохранение, поиск, уничтожение и т. п.

Приведем популярные системы программирования – *Turbo Basic*, *Quick Basic*, *Turbo Pascal*, *Turbo C*.

Язык программирования C (Си) разработан Деннисом Ритчи в 1972 г. Язык C хорошо зарекомендовал себя тем, что на нем было написано более 90% всего кода операционной системы UNIX.

В настоящее время язык C стал очень популярен. В нем сочетаются удобство, краткость и мобильность языков высокого уровня. Язык не очень простой в изучении, но позволяет создавать сложные и эффективные программы.

В последнее время широкое распространение получили системы программирования, ориентированные на создание *Windows-приложений*:

- Пакет *Borland Delphi* (Дельфи) – семейства компиляторов Borland Pascal. Его быстрый компилятор позволяет эффективно и быстро решать практически любые задачи прикладного программирования.
- Пакет *Microsoft Visual Basic* – удобный и популярный инструмент для создания Windows-программ с использованием визуальных средств. Содержит инструментарий для создания *диаграмм* и *презентаций*.
- Пакет *Borland C++* – одно из самых распространенных средств для разработки DOS и Windows-приложений.

Уровень – Знание и понимание



1. Что называется системой программирования?
2. Что включают в себя системы программирования?
3. Какие популярные системы программирования вам известны?
4. Какие компоненты необходимы для создания программы?
5. Почему язык является ядром системы программирования?
6. Какие языки программирования вы знаете?
7. Что входит, как правило, в состав системы программирования?
8. В чем отличия языков программирования разных видов?
9. В чем отличие между языком программирования и системой программирования?





Уровень – Применение, анализ, синтез



- Исправьте ошибки в высказываниях.
 - Только алгоритмический язык Паскаль является языком низкого уровня.
 - Программирование на языке высокого уровня начинается с выбора алфавита и синтаксиса.
 - Компилятор – это интерпретатор программы с языка программирования в машинные коды.
 - Никогда компилятор не может быть транслятором.
- Составьте кроссворд со словами: алгоритм, программа, язык, транслятор, компилятор, код, интерпретатор.

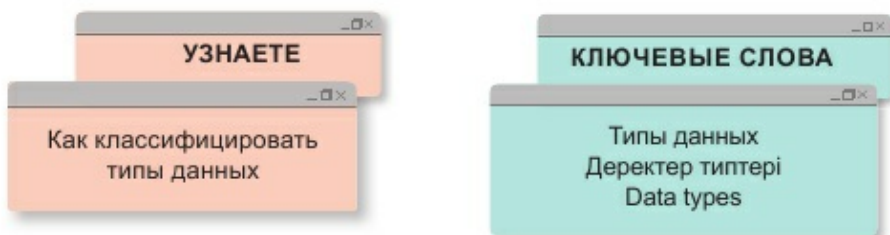


Уровень – Оценивание



Напишите эссе на тему: Чем отличаются понятия «система программирования» и «язык программирования»?

4.3. Типы данных



Для обработки данных их необходимо хранить в памяти. При этом к этим данным надо как-то обращаться. Обычно люди обращаются друг к другу по имени, такой же способ используется в программировании: каждой ячейке памяти (или группе ячеек) дается свое собственное **имя**. Используя это имя, можно прочитать информацию из ячейки и записать туда новую информацию.

ЗАПОМНИТЕ



Переменная – это ячейка в памяти компьютера, которая имеет имя и хранит некоторое значение. Значение переменной может меняться во время выполнения программы. При записи в ячейку нового значения старое стирается.



Имя переменной состоит из последовательности латинских букв и цифр. Причем имя должно начинаться с буквы. Имя переменной не должно включать следующие символы: `,` `/`, `:`, `*`, `?`, `*`, `<`, `>`, `|`.

Имена переменных подбираются со смыслом. Например, если нужно подсчитать пройденный путь, то имена целесообразно назначать в соответствии с уже знакомой формулой пути: *S*, *v*, *t*.

С каждым именем (**идентификатором**) в программе связан **тип**. Тип задает те операции, которые могут применяться к имени. Это означает, что все операции подвергаются строгому контролю со стороны компилятора на **соответствие типов**. Недопустимые операции не компилируются, что позволяет исключить ошибки и повысить надежность программ.

Мы с вами будем изучать классификацию типов данных в языке программирования C++. В данном языке программирования имеется набор встроенных типов данных для представления целых и вещественных чисел, символов, а также тип данных **символьный массив**, который служит для хранения символьных строк.

Типы данных в C++ еще называют **расширяемыми**. Понятие «расширяемый» означает, что кроме встроенных типов данных можно создавать свои типы данных. Поэтому в C++ существует огромное количество типов данных (табл. 4.1).

Таблица 4.1

Типы данных

Основные типы данных в C++	Объявление переменной	Диапазон принимаемых значений
1	2	3
int целочисленный тип данных	<code>int a;</code> //объявление переменной a целого типа	-2 147 483 648 / 2 147 483 647
Приставки к целочисленному типу данных: <ul style="list-style-type: none"> • short – укорачивает тип данных путем уменьшения размера занимаемой памяти; • long – удлиняет тип данных путем увеличения размера занимаемой памяти; • unsigned (без знака) – увеличивает диапазон положительных значений в два раза 		-32 768 / 32 767 -2 147 483 648 / 2 147 483 647 0 / 4 294 967 295



1	2	3
float тип данных с плавающей точкой	<code>float b;</code> // объявление переменной <code>b</code> типа данных с плавающей точкой	-2 147 483 648.0 / 2 147 483 647.0
long float		-9 223 372 036 854 775 808.0 / 9 223 372 036 854 775 807.0
double тип данных с плавающей точкой двойной точности	<code>double c = 14.2;</code> // инициализация переменной типа <code>double</code>	-9 223 372 036 854 775 808.0 / 9 223 372 036 854 775 807.0
char символьный тип данных	<code>char d = 's';</code> // инициализация переменной типа <code>char</code>	0 / 255
bool логический тип данных	<code>bool k = true;</code> // инициализация логической переменной <code>k</code> . У логического выражения может быть один из двух результатов – <code>true</code> или <code>false</code>	0 / 255
void тип данных с отсутствующим значением	С его помощью задаются указатели на объекты неизвестного типа или функции, не возвращающие значение	

Тип `char` служит для хранения отдельных символов и небольших целых чисел. Он занимает один машинный байт. Ниже представлено описание типа `char`:

```
{
  char symbol = 'A'; // объявление переменной типа char и инициализация ее символом 'A'
}
```



Объявленная переменная типа `bool` (логического типа) инициализирована значением `17`. Переменная типа `bool` может содержать два значения – `0` (ложь) или `1` (истина).

```
{
  bool boolean = 17; // переменная типа bool с именем boolean
}
```

ВЫПОЛНИМ ВМЕСТЕ

Даны типы данных `char`, `short`, `int` и `long`. Вместе они составляют *целые типы*, которые, в свою очередь, могут быть *знаковыми* (signed) и *беззнаковыми* (unsigned).

В беззнаковых типах все биты используются для значения. В знаковых типах самый левый бит служит для хранения знака (`0` – «плюс», `1` – «минус»), а оставшиеся биты содержат значение.

Попробуйте привести примеры и объяснить описание типов данных `signed char` (значения от `-128` до `127`) и `unsigned char` (от `0` до `255`).

Уровень – Знание и понимание



1. Что такое *переменная*?
2. Как подбираются имена переменных?
3. Как объявляется переменная?
4. Какие бывают характеристики у переменных?
5. Что такое *тип переменной*?
6. Какие типы данных называются встроенными?
7. По каким критериям выбирается тип для переменной?
8. Перечислите самые популярные типы переменных.
9. Для чего используется `int`? Какие приставки применяются к целому типу данных?
10. Для чего используются `float`, `double`, `char`, `bool`, `cin`?

Уровень – Применение, анализ



Допишите комментарии. Каким типом данных объявлены переменные?

```
int a; // объявление переменной a типа
float b; //
double c = 14.2; //
char d = 's'; //
bool k = true; //
```





Уровень – Синтез

Опишите переменные для размещения в них:

- пола человека (один символ);
- целого положительного двузначного числа;
- целого положительного трехзначного числа;
- целочисленного отрицательного четырехзначного числа;
- небольшого вещественного числа;
- названия цвета;
- вычисления расстояния.



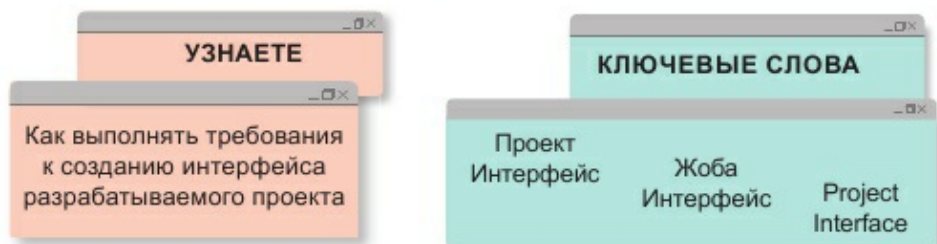
Уровень – Оценивание



Какие из приведенных ниже определений переменных содержат синтаксические ошибки?

- a) `int car = 1024, auto = 2048;`
- b) `int ival = ival;`
- c) `int ival(int());`
- d) `double salary = wage = 9999.99;`
- e) `cin >> int input_value;`

4.4. Интерфейс проекта



ПОДУМАЙТЕ И ОТВЕЬТЕ

Задумывались ли вы когда-нибудь об интерфейсе тех или иных приложений, с которыми работали? Каким должен быть интерфейс разрабатываемого проекта? На что должны обратить внимание при разработке дизайна проекта?

В понятие **пользовательского интерфейса** входит не только картинка на экране – трехмерная, анимированная, выполненная в модном дизайне, – но и способы взаимодействия пользователя с системой.



Разработка интерфейсов для проектов привела к появлению соответствующих стандартов. При написании программы программиста интересует, как осуществить **функциональность, эффективность и технологичность** проекта, не затрачивая существенные ресурсы компьютера, свои силы и время.

В дизайне интерфейса условно выделяют две составляющие – *декоративную* и *активную*. К *декоративному* интерфейсу относятся элементы, отвечающие за эстетическую привлекательность программного продукта. *Активные* элементы – это управляющие средства пользовательского интерфейса, посредством которых пользователь управляет программой.

Например, среда электронной таблицы Excel содержит удобный интерфейс для автоматического копирования данных. Интерфейс данного программного средства обеспечивает **комфортность** и **оперативность** реакции на действия пользователя, **удобство** и **скорость** манипулирования мышью и клавиатурой для эффективной работы.

Основу взаимодействия пользовательского интерфейса составляют диалоги – это регламентированный обмен информацией, осуществляемый в реальном времени. Диалог направлен на совместное решение конкретной задачи – обмен информацией и координация действий.

Различают *процедурно ориентированный* и *объектно ориентированный* подходы к разработке интерфейса проекта.

Процедурно ориентированный использует традиционные модели взаимодействия с пользователем, основанные на понятиях *процедура* и *операция*. В этой модели программное обеспечение предоставляет пользователю возможность выполнения некоторых действий, для которых пользователь определяет данные и следствием выполнения которых является желаемый результат.

Объектно ориентированный интерфейс – в этой модели пользователь напрямую взаимодействует с каждым объектом и инициирует выполнение операций, в процессе которых могут взаимодействовать несколько объектов. В этом случае пользователь может создавать объекты, изменять их параметры и связи между ними и инициировать взаимодействие между ними.

Процедурно ориентированные интерфейсы делят на три типа: примитивный, меню, со свободной навигацией (графический).

• **Примитивным** называют интерфейс, который организует взаимодействие человека с компьютером с помощью устройств ввода-вывода. Обычно этот интерфейс организует ввод данных, решение, вывод результатов.



- **Меню** – этот интерфейс позволяет пользователю выбирать необходимые операции из необходимого списка. Последовательность действий в данном интерфейсе определяется пользователем.

- **Графический** пользовательский интерфейс ориентирован на поддержку интерактивного взаимодействия с программным обеспечением, осуществляя обратную связь с пользователем.

При проектировании интерфейса необходимо учитывать **психофизические особенности** человека, связанные с *восприятием, запоминанием и обработкой информации*.

В процессе обработки информации мозг сравнивает поступающие данные с предыдущими, поэтому при **смене кадра** мозг на некоторое время блокируется. Краткосрочная память – это самое узкое место в системе обработки информации человека. Невостребованная информация хранится не более 30 секунд. Если необходимо ввести или запомнить число либо группу символов, то количество не должно превышать 7 ± 2 .

Цвет в сознании человека ассоциируется с фоном и является сильным раздражителем. Поэтому применять цвета в интерфейсе следует осторожно. Обилие оттенков привлекает внимание, но быстро утомляет. Интерфейс рекомендуют делать в единой цветовой гамме и учитывать индивидуальные особенности восприятия цветов человеком. Для этого по возможности пользователю нужно предоставить возможность настраивать цвета самому.

Звук в интерфейсе используют для привлечения внимания как фон и как источник дополнительной информации. Обязательно нужно предусмотреть отключение звука.

Основными критериями оценки интерфейса пользователя являются:

- 1) простота освоения и запоминания системы;
- 2) скорость достижения результатов при использовании системы;
- 3) субъективная удовлетворенность при эксплуатации системы.

Рассмотрим интерфейс **интегрированной среды разработки (ИСР)**. Эта система программных средств используется для разработки программного обеспечения. В английском языке такая среда называется **Integrated development environment** или сокращенно IDE.

Интерфейс в ИСР предусматривает **подсветку синтаксиса** – выделение синтаксических конструкций текста с использованием различных цветов, шрифтов и начертаний. Обычно применяется в текстовых редакторах для облегчения чтения исходного текста, улучшения визуального восприятия.



Одна из наиболее важных частей интерфейса ИСР – отладчик. Он представляет собой модуль среды разработки или отдельное приложение, предназначенное для поиска ошибок в программе. Отладчик позволяет отслеживать, устанавливать или изменять значения переменных в процессе выполнения программы, устанавливать и удалять контрольные точки или условия остановки и т. д.

ПРОАНАЛИЗИРУЙТЕ И ОЦЕНИТЕ

Для программирования решений будем использовать бесплатную среду разработки **Code::Blocks**, позволяющую писать программы не только на C/C++, но и обеспечивающую поддержку таких языков программирования, как Fortran и D с некоторыми ограничениями.



Компилятор **Code::Blocks** преобразует исходный код (файлов *.c, *.cpp) в работающую программу. **Code::Blocks** можно найти в Интернете и установить на компьютер.

*После установки ИСР **Code::Blocks** изучите интерфейс основного окна. Какой редактор используется для ввода исходного кода в ИСР? Обеспечен ли интерфейс ИСР **Code::Blocks** расширенной функциональностью – подсветкой синтаксиса, сортировкой строк, показом кодов символов и т. п.?*

При программировании решений в ИСР **Code::Blocks** будут применены перечисленные выше требования к созданию интерфейса разрабатываемого проекта.

Уровень – Знание и понимание



1. Что лежит в основе разработки интерфейса проекта?
2. Какие виды дизайна пользовательского интерфейса вы знаете?
3. Что характеризует качество интерфейса?
4. Какие новые особенности появились при разработке компьютерного интерфейса?
5. Какие три типа в процедурно ориентированном интерфейсе?
6. Какие аспекты необходимо учитывать при проектировании пользовательского интерфейса?
7. Перечислите основные критерии оценки интерфейса пользователя.
8. Чем отличается интерфейс настольного компьютера от ноутбука?



Уровень – Применение



В представленном кроссворде (рис. 4.6) дайте определение словам по горизонтали и вертикали.

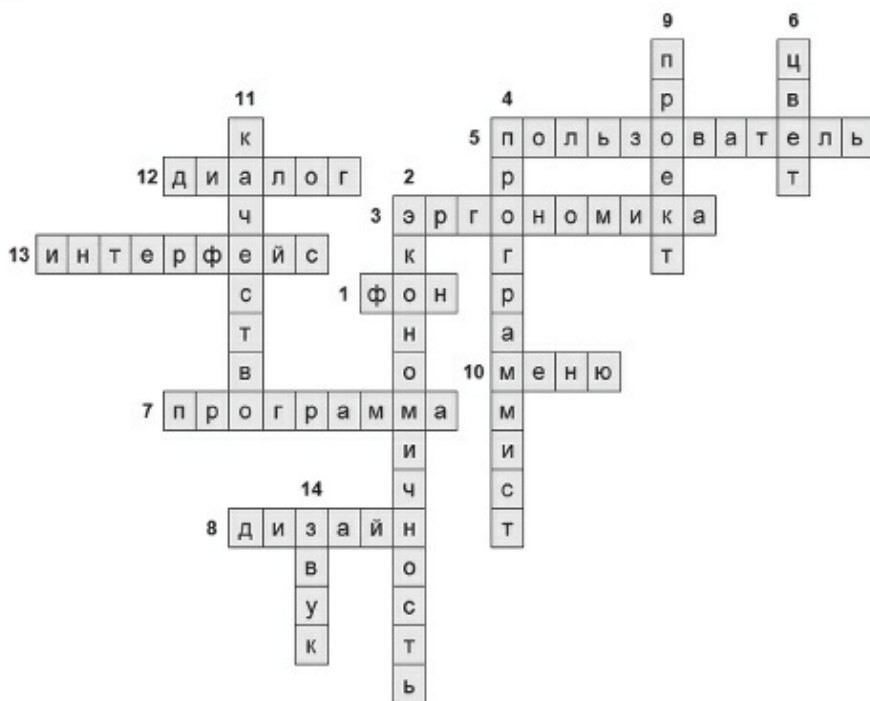


Рис. 4.6. Интерфейс проекта

Уровень – Анализ



Перечислите и проанализируйте особенности интерфейса интегрированной среды разработки Code::Blocks.

Уровень – Синтез

Разработайте проект анкетирования в электронной таблице, чтобы интерфейс соответствовал основным критериям оценки интерфейса пользователя: простота, комфортность, скорость работы.





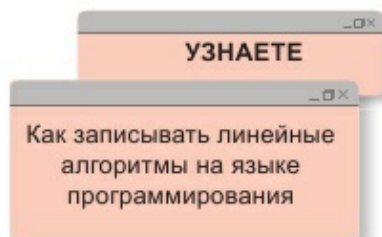
Уровень – Оценивание

Поисковая парная работа с интернет-источниками. Используйте табл. 4.2 для проведения классификации нескольких сайтов, соответствующих классу интерфейса: символьный или графический. Оцените из выбранных сайтов лучший интерфейс.

Таблица 4.2

Классы интерфейса	Подклассы	Примеры типов управляющих средств
Символьный	Командный интерфейс	Командная строка
Графический	Простой графический	Управляющие клавиши
	Графический двухмерный	Меню Графические элементы управления
	Трёхмерный	Конические деревья (визуализация информации в виде детской пирамидки)

4.5. Программирование линейных алгоритмов



Одним из самых распространенных программных языков является C++. Он был создан в начале 1980-х годов Бьерном Страуструпом. В своей работе автор языка программирования опирался на опыт создателей языка C.



Как и в описании естественных языков, таких как русский, английский, казахский и др., используются понятия: *алфавит*, *синтаксис* и *семантика*.

ЗАПОМНИТЕ



Алфавит языка – это множество простейших знаков, которые могут быть использованы в текстах этого языка.

Алфавит языка C++ состоит из:

- прописных и строчных букв латинского алфавита;
- арабских цифр от 0, 1,.....9;
- разделителей: , . ; : ? ! ' " | / \ ~ _ ^ () { } [] < > # % & - = + *;
- неизображаемых символов: пробел, табуляция, переход на новую строку и др.;

// – последовательности знаков, начинающих комментариев.

Из символов языка формируются лексемы – это:

- идентификаторы;
- константы;
- знаки операций;
- ключевые слова;
- разделители.

Для примера рассмотрим запись линейного алгоритма на языке программирования. **В линейном алгоритме операторы выполняются друг за другом.**

Ниже представлена программа, выполняющая арифметические операции: деление, сложение, вычитание и умножение друг за другом.

```
#include "stdafx.h"
#include "locale"
#include "iostream"
using namespace std;
int main()
{
    setlocale(LC_ALL, "Russian");
    float c, c1, c2, c3, a, b; // Числа с плавающей запятой
```




```

cout << "Введите коэффициенты a и b" << endl;
cin >> a >> b;
cout << endl; // переход на следующую строку
c = a / b; // также +, -, *.
c1 = a + b;
c2 = b - a;
c3 = a * b;
cout << "c = " << c << endl;
cout << "c1 = " << c1 << endl;
cout << "c2 = " << c2 << endl;
cout << "c3 = " << c3 << endl;
system("PAUSE");
return 0;
}

```

Приведем расшифровку программы. Слева даны команды программы, справа – комментарии, поясняющие команду.

<pre> #include "stdafx.h" #include "locale" #include "iostream" int main() setlocale (LC_ALL, "Russian"); { float c, c1, c2, c3, a, b; cout << "Введите ко- эффициенты a и b" << endl } </pre>	<pre> //используется команда для подключения за- головочных файлов. "stdafx.h", "locale", "io- stream" – имя заголовочного файла, внутри угловых скобок не должно быть пробелов. Для подключения нового заголовочного файла надо использовать новую команду #include; //сообщает компилятору, что есть функция с именем main и что функция возвращает це- лое число типа int. Пустые скобки означают, что main не имеет аргументов. //функция, выводящая на экран русский алфа- вит; //{фигурные скобки обозначают начало; // числа с плавающей запятой; //используется для отображения текста. Он ис- пользует символы <<, чтобы указать, что от- правляется к выводу на экран. //}и конец основной программы; </pre>
---	--



Ниже представлена программа вычисления арифметических операций, выполненная в Code::Blocks (рис. 4.7).

```

1 #include <iostream>
2 using namespace std;
3 int main()
4 {
5     setlocale(LC_ALL, "Russian");
6     float c, c1, c2, c3, a, b; // float - вещественное
7     cout << "Введите коэффициенты a и b" << endl;
8     cin >> a >> b;
9     cout << endl; // перевод строки
10    c = a / b; // операция деления
11    c1 = a + b;
12    c2 = b - a;
13    c3 = a * b;
14    cout << "c = " << c << endl;
15    cout << "c1 = " << c1 << endl;
16    cout << "c2 = " << c2 << endl;
17    cout << "c3 = " << c3 << endl;
18    return 0;
19 }

```

```

22.8
12.8
c = 1.9
c1 = 34.8
c2 = -10.8
c3 = 273.6
Process returned 0 (0x0)   execution time : 26.144 s
Press any key to continue.

```

Рис. 4.7. Результат выполнения арифметических операций

Уровень – Знание и понимание



1. Что такое *алфавит языка*?
2. Из чего формируются лексемы?
3. Что такое *интегрированная среда разработки*? Как ее запустить?
4. Какие алгоритмы называются линейными?
5. Какие основные этапы разработки программы существуют?
6. Приведите примеры задач, содержащих линейный алгоритм решения.
7. Оцените интерфейс консольного ввода данных в интегрированной среде разработки Code::Blocks.

Уровень – Применение



1. Дан фрагмент программы:

```

#include <iostream>
int main ()
{
    cout << "Hello, World!\n";
}

```

- Что означает строка `#include`?
- Будет ли иметь смысл выражение `<< "Hello, World!\n"` без `cout`?
- Куда будет выведена запись: `"Hello, World!\n"`?



- Верно ли будет высказывание "Строка – это последовательность символов, заключенная в двойные кавычки"?
- Почему каждый оператор языка C++ заканчивается точкой с запятой?

2. Что будет напечатано программой в результате ее выполнения?

```
#include <iostream>
using namespace std;
Setlocale (0, "Rus")
int main()
{
    cout << "Я буду программировать на C++";
    cin.get();
}
```

После запуска интегрированной среды разработки Code::Blocks откройте меню **File** → **New** → **Project** → **Console Application** → **Go** (рис. 4.8, 4.9).

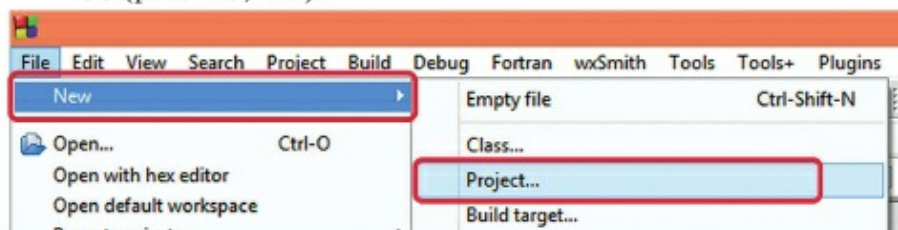


Рис. 4.8. Создание нового проекта



Рис. 4.9. Выбор Console Application и C++

Придумайте название проекту и нажмите **ОК** (рис. 4.10).

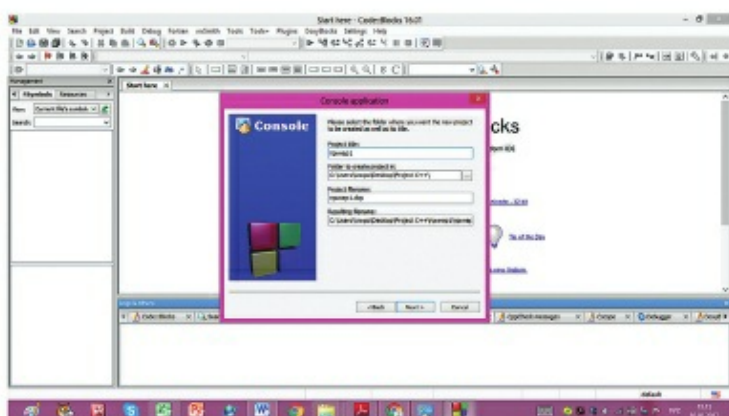


Рис. 4.10. Название проекта в Code::Blocks

Посмотрите видеофильм установки и создания первого проекта **Code::Blocks**.



Уровень – Анализ

1. Посмотрите видеофильм в Youtube по созданию проекта «Hello World!».
2. Проанализируйте задание, которое было выполнено совместно на уроке, и дайте комментарии командам, которые не описаны в учебнике.



Уровень – Синтез



Известны диагонали ромба d_1 и d_2 . Найти периметр P и площадь S . Составьте арифметические выражения, используя табл. 4.3. Запишите линейный алгоритм для задачи.



Уровень – Оценивание

Найдите ошибку в программе.

```
#include <iostream>
using namespace std;
```




```

int main()
{
int d1=10, d2=12, P, S, pold1, pold2; double a;
S=d1*d2/2;
cout<<"S="<<S<<\n;
pold1=d1/2;
pold2=d2/2;
a=pold1*pold1+pold2*pold2;
P=sqrt(a)*4;
cout<<"P="<<P<<\n;
}

```



Идентификатор – последовательность из букв латинского алфавита, цифр и символов подчеркивания, которая начинается с буквы или символа подчеркивания (например, `_alf_a1`). В языке C++ большие и маленькие буквы различаются и будут являться разными идентификаторами. Например, `book` и `BOOK` – **разные идентификаторы**.

Константа – величина, которая не изменяется в процессе решения задачи. Идентификаторы, зарезервированные в языке, называют **служебными словами**.

Алгоритм в программной среде редактора C++ записывается в виде **команд**. Команды, записанные на языке программирования, состоят из **операторов** и **параметров**.

Операторы – зарезервированные в языке программирования слова или символы, выполняющие определенные функции.

Параметры уточняют или конкретизируют выполнение необходимых действий команды. Каждый оператор языка C++ заканчивается **точкой с запятой (;)**. Она сообщает компилятору, что это конец команды.

Алгоритм, записанный на языке программирования, называется **программой**.



**Основные этапы разработки программы:**

1. Напишите программу на языке программирования C++.
2. Введите программу в текстовый редактор ИСР Code::Blocks.
3. Откомпилируйте в ИСР в объектный файл.
4. Если компьютер обнаруживает в программе ошибки, то сообщает об этом.
5. Исправьте ошибки и снова компилируйте программу.
6. Если ошибок нет, то запустите программу.

Для реализации различных алгебраических выражений в C++ применяются арифметические операции, математические функции из библиотеки `math.h` (табл. 4.3).

Таблица 4.3

Арифметические операции и математические функции

Математическая запись	Запись на C++	Назначение
+	+	сложение
-	-	вычитание
*	*	умножение
/	/	деление
%	%	остаток от деления
x^y	<code>pow(x,y)</code>	Число x в степени y
$ x $	<code>fabs(x)</code>	Модуль числа x
\sqrt{x}	<code>sqrt(x)</code>	Квадратный корень из x

Примеры записи математических выражений:

Математическая запись

$$1. x^2 - 7x + 6$$

$$2. \frac{|x| - |y|}{1 + |xy|}$$

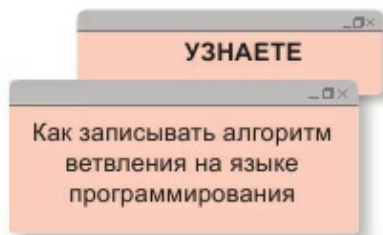
Запись на C++

$$\text{pow}(x,2) - 7 * x + 6$$

$$(\text{fabs}(x) - \text{fabs}(y)) / (\text{float})(1 + \text{fabs}(x * y))$$



4.6. Программирование алгоритмов ветвления



Далеко не все алгоритмы являются линейными. Встречаются такие ситуации, в которых при выполнении некоторого условия необходимо сделать одни действия, а в противном случае (если условие не выполнилось) необходимо выполнить другие действия или ничего не делать. В таком случае говорят, что в алгоритме использована алгоритмическая конструкция ветвления.

Для программирования алгоритмов ветвления в языке C++ используется **условный оператор if**, или **ветвления**. Условные операторы бывают полной и неполной формы. Структура условного оператора в полной форме представляется в виде:

ЗАПОМНИТЕ



if (условие) оператор1; else оператор2;

Данная структура условного оператора означает: *Если (if) выражение (условие) истинно, выполняется оператор1, иначе (else) выполняется оператор2.*

Ниже представлена структура **неполной формы** условного оператора:

ЗАПОМНИТЕ



if (условие) оператор;

Эта форма используется тогда, когда в случае невыполнения условия ничего делать не надо.

При записи синтаксиса условного оператора помните о следующих особенностях:



- записывается в круглых скобках;
- обязательно ставится точка с запятой после оператора.

Проверяемое условие может быть любым логическим выражением. Логические выражения могут быть образованы операциями равенства и отношения (табл. 4.4).

Таблица 4.4

Логические выражения C++

Операция в C++	Условие	Смысл записанных условий в C++
==	a == b	a равно b
!=	a != b	a не равно b
>	a > b	a больше b
<	a < b	a меньше b
>=	a >= b	a больше или равно b
<=	a <= b	a меньше или равно b

ВЫПОЛНИМ ВМЕСТЕ

Введите целое число с клавиатуры. Если это число больше 0, то программа должна выполнить действие «Это число отрицательное», иначе – «Это число положительное».

Запишем этот алгоритм на языке программирования C++ с помощью алгоритма ветвления.

```
#include <iostream>
using namespace std;
int main()
{
    setlocale(0, " ");
    int m;
    cout << "Введите произвольное число: ";
    cin >> m;

    if (m < 0) { // Если введенное число меньше 0.
        cout << "Это число отрицательное. \n"; } //Условный
    else { // иначе оператор
        cout << "Это число положительное. \n";}

    return 0;
}
```



Здесь говорится: «Если переменная **m** меньше 0 – вывести соответствующее ей сообщение. **Иначе** – вывести другое сообщение».

Если в блоке «if» или в блоке «else» только один оператор, то фигурные скобки можно не ставить. Ввод данных с клавиатуры с отображением их на экране в C++ осуществляется с помощью операторов – **cin**.

Уровень – Знание и понимание



1. Какой оператор используется для программирования алгоритмов ветвления в языке C++?
2. Перечислите структуру полного условного оператора.
3. Перечислите структуру неполного условного оператора.
4. Какие операции равенства и отношения записывают в логических выражениях C++?
5. Какие синтаксические правила нужно соблюдать для записи условного оператора?
6. Как выполняется структура условного оператора?

Уровень – Применение



1. Объясните порядок выполнения в представленной ниже программе. Какое условие в нем проверяется?

```
#include <iostream>
using namespace std;
int main()
{
    setlocale(0, " ");
    float temp;
    cout << "Введите произвольное число: ";
    cin >> temp;

    if (temp== 36.6) {
        cout << "У вас нормальная температура тела"; }
    else { // иначе
        cout << "У вас повышенная температура";}
    return 0;
}
```

2. Найдите ошибки, допущенные в программе.

```
#include <iostream>
```



```
using namespace std;
setlocale (0,"Rus")
int main ()
{
    int number
    cout <<"Введите число:"
    cin >> number;
    cout <<"Вы ввели:" <<number <<"\n";
    cin.get()
```



Уровень – Анализ

Запишите программу условного оператора, в котором сравнивают значения переменных a и b . Если число больше, то вывести его на печать.



Уровень – Синтез

1. Напишите программу для решения следующей задачи:
Дано целое число. Является ли оно четным?
2. Составьте программу вычисления y по заданному x .

$$y = \begin{cases} x, & \text{при } x > 0 \\ \frac{1}{x}, & \text{при } x \leq 0 \end{cases}$$



Уровень – Оценивание

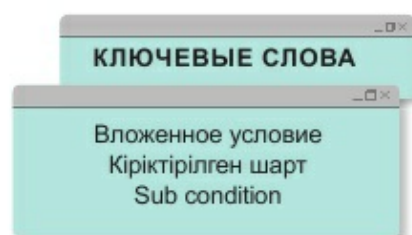
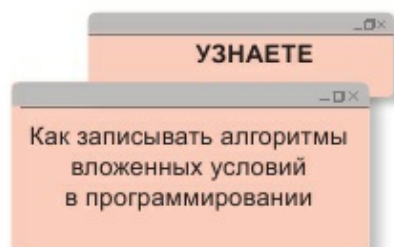


Проведите исследования на тему: «Основные концепции и конструкции языка C++».

Примечание. Для исследования примените книгу «Язык программирования C++». Автор: Бьёрн Страуструп. Интернет-ресурсы: https://vk.com/doc16756779_228187070?hash=-88e5b6ccd4f822602c&dl=ccc12f550fe4ac896f.pdf.



4.7. Программирование вложенных условий



При решении задач часто приходится рассматривать не два, а большее количество вариантов условий. Это можно реализовать, используя несколько условных операторов. Когда оператор **If** появляется внутри другого оператора **If**, они считаются **вложенными**. Такое вложение используется для уменьшения числа необходимых проверок.

Этот метод часто обеспечивает большую эффективность, однако одновременно он уменьшает наглядность программы. Не рекомендуется использовать более одного-двух уровней вложения **If**. За вторым уровнем вложения становится трудно восстановить последовательность проверки условий каждым условным оператором.

Структура вложенных операторов:

ЗАПОМНИТЕ



```
if (условие) оператор;
    else
if (условие) оператор;
    else
if (условие) оператор;
    else оператор;
```

В конструкции вложенных операторов каждому оператору **if** соответствует только один оператор **else**. Совокупность этих операторов **else if** означает, что если не выполняется предыдущее условие, то проверяется данное. Если ни одно из условий не верно, то выполняется тело оператора **else**.



ПРОАНАЛИЗИРУЙТЕ

Какое сообщение будет выводиться в приведенном ниже фрагменте программы в зависимости от вложенного условия? Сколько условий проверяется? Какие условия проверяются? Почему во втором условии применяется оператор равенства (`==`), а не оператор присваивания? Если поставить оператор присваивания в условии, то при проверке условия значение переменной изменится. Выполнится ли это условие после изменения?

```
if (a < 100) // Если введенное число меньше 100.
    cout << "Это число меньше 100.\n ";
else if (a == 100)
    cout << "Это число равно 100.\n ";
else // иначе cout << "Это число больше 100.\n ";
}
```

Вложенный
условный
оператор if

Таким образом, для выбора одного из нескольких вариантов в зависимости от значения некоторой переменной, можно использовать несколько вложенных операторов `if`. Ее альтернативной заменой является оператор `switch`. Структура `switch` значительно удобнее, чем структура вложенных операторов `if`.

ЗАПОМНИТЕ**Структура оператора `switch`:**

```
switch (/*переменная или выражение*/)
{ case /*константное выражение1*/:
  {
    /*группа операторов*/;
    break;
  }
  case /*константное выражение2*/:
  {
    /*группа операторов*/;
    break;
  }
  // . . .
  default:
  {
    /*группа операторов*/;
  }
}
```



Разберем структуру оператора множественного выбора **switch**:

- заголовок и тело оператора заключены в фигурные скобки;
 - в заголовке после ключевого слова **switch** в круглых скобках записывается имя переменной. В зависимости от значения этой переменной делается выбор между несколькими вариантами;
- каждому варианту соответствует метка **case**, после которой стоит одно из возможных значений этой переменной и двоеточие;
 - если значение переменной совпадает с одной из меток, то программа переходит на эту метку и выполняет все последующие операторы;
- оператор **break** служит для выхода из тела оператора **switch**;
- если значение переменной не совпадает ни с одной из меток, программа переходит на метку **default** (по умолчанию), т. е. если ничего другого не указано.

ВЫПОЛНИМ ВМЕСТЕ

Рассмотрим простой пример выполнения арифметических операций: сложения, вычитания, умножения и деления двух чисел, введенных с клавиатуры.

```
#include <iostream>
using namespace std;

int main(int argc, char* argv[]) // строка автоматически создается
при создании консольного приложения в языке программирова-
ния C++
{
    setlocale(0, "Rus");
    int d; // переменная для выбора в switch
    int a,b; // переменные для хранения операндов
    cout << "Введите первое число: ";
    cin >> a;
    cout << "Введите второе число: ";
    cin >> b;
    cout << "Выберите действие: 1 – сложение; 2 – вычитание; 3 –
    умножение; 4 – деление: ";
    cin >> d;
```



```

switch (d) // начало оператора switch
{
    case 1: // если d = 1
    {
        cout << a << " + " << b << " = " << a + b << "\n"; // вы-
        полнить сложение
        break;
    }
    case 2: // если d = 2
    {
        cout << a << " - " << b << " = " << a - b << "\n"; // вы-
        полнить вычитание
        break;
    }
    case 3: // если d = 3
    {
        cout << a << " * " << b << " = " << a * b << "\n"; // вы-
        полнить умножение
        break;
    }
    case 4: // если d = 4
    {
        cout << a << " / " << b << " = " << a / b << "\n"; // вы-
        полнить деление
        break;
    }
    default: // если d равно любому другому значению
        cout << "Неправильный ввод. \n";
    }
}

```

В
л
о
ж
е
н
н
ы
й

у
с
л
о
в
н
ы
й

о
п
е
р
а
т
о
р

ПРОКОММЕНТИРУЙТЕ

Рассмотрите программу и сделайте комментарии к ней. В программе объявлена переменная **d** целочисленного типа. Со значением этой переменной программа будет сравнивать константное выражение. Для хранения введенных чисел объявлены две переменные **a** и **b** целого типа.



В условном операторе множественного выбора **switch** на начальном этапе анализируется переменная **d**:

- если переменная **d** равна 1, выполняется блок операторов – ...
- если переменная **d** равна 2, выполняется блок операторов – ...
- если переменная **d** равна 3, выполняется блок операторов – ...
- если переменная **d** равна 4, выполняется блок операторов – ...

Если же значение переменной **d** не совпадает ни с одним константным выражением, то передается управление ветви программы, содержащей зарезервированное слово – ...

Какая строка будет выведена на экран?

Оператор **switch** может содержать и не содержать слово **default**. Если значение переменной не совпадет ни с одним константным выражением и не будет **default**, то программное управление в этом случае просто перейдет к первому оператору после **switch**. Если убрать оператор **break**, то программа будет дальше сравнивать значение переменной с константными выражениями.

Уровень – Знание и понимание



1. Какие условия называются вложенными?
2. Перечислите структуру вложенных операторов **if**.
3. Каким образом выполняются вложенные операторы?
4. Почему удобнее использовать специальный оператор **switch**?
5. Для чего используется **default**?
6. Для чего используется **break**?
7. Отсутствие **default** и **break** будет влиять на выполнение вложенного оператора?
8. Чем отличаются вложенные условные операторы **if** и **switch**?

Уровень – Применение



1. Для разветвляющегося алгоритма рассмотрим задачу вложенного цикла условного оператора, в котором в зависимости от радиуса окружности будет выводиться сообщение: «точка внутри круга», «точка вне круга», «точка на границе» (рис. 4.11).

Введите программу в Code::Blocks, откомпилируйте ее. Дайте объяснения сообщениям, которые выводятся на экран в зависимости от введенного радиуса.



```

1  #include <locale>
2  #include <iostream>
3  using namespace std;
4  int main()
5  {
6  setlocale(LC_ALL, "Russian");
7  float R, x, y;
8  cout << "Введите значение радиуса: ";
9  cin >> R;
10 cout << "Введите значение координат: ";
11 cin >> x >> y;
12 if ( (x*x + y*y) < R*R ) cout << "Точка внутри круга";
13 if ( (x*x + y*y) > R*R ) cout << "Точка вне круга";
14 else cout << "Точка на границе";
15 return 0;
16 }

```

```

"C:\Users\user\Desktop\Project C++\final\bin\
Введите значение радиуса:
5
Введите значение координат:
18
29
Точка вне круга.
Process returned 0 (0x0)   execution time : 21.220 s
Press any key to continue.

```

Рис. 4.11. Компиляция и выполнение программы

2. Найдите ошибку в программе.

```

#include <iostream>
using namespace std;
int main()
{
    setlocale(0, "Rus");
    int osenka;
    cout << "Введите произвольную цифру от 2 до 5: ";
    cin >> osenka;
    if (osenka = 5) // Если введенная цифра равна 5.
        cout << "Вы получили оценку «5»\n ";
    else if (osenka = 4) //Если введенная цифра равна 4
        cout << "Вы получили оценку «4».\n ";
    else (osenka = 3) //Если введенная цифра равна 3
        cout << "Вы получили оценку «3».\n ";
        else // иначе cout << "Вы получили оценку «2».\n ";
}

```

3. Вычислите значение функции y по заданным условиям:

$$x - 12, x > 0$$

$$y = \begin{cases} 5, & x = 0 \\ x^2, & x < 0 \end{cases}$$



При решении этой задачи вложенный условный оператор будет иметь алгоритм:

```
если  $x > 0$ 
то
 $y$  вычислить по формуле  $y = x - 12$ 
иначе
если  $x = 0$ 
то
 $y$  вычислить по формуле  $y = 5$ 
иначе
 $y$  вычислить по формуле  $y = \text{sqr}(x)$ ;
```

Напишите фрагмент программы в C++, используя вложенный оператор `if, else`.

Уровень – Анализ

У вас есть 1000 тенге. Если стоимость билета в кино – 1000 тенге, тогда вы купите билет, если стоимость билета в кино – 700 тенге, то купите билет и колу. Если стоимость билета в кино – 1200 тенге, тогда кино посмотрите в другой раз. Напишите фрагмент программы в C++, используя вложенный оператор `switch`.

Уровень – Синтез

Напишите программу определения дня недели по введенному номеру. Является он рабочим днем или выходным?

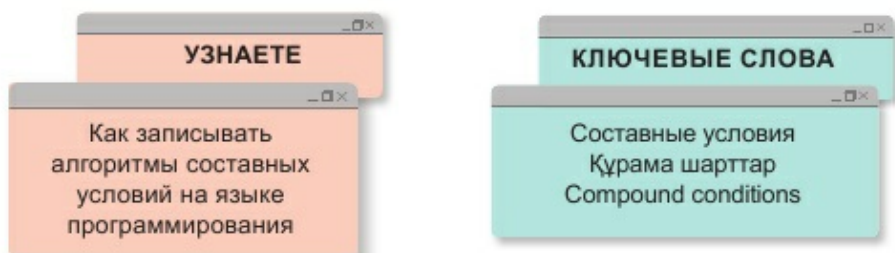
Уровень – Оценивание

Напишите программу пересчета из одной величины в другую.

- 1 верста – 10668,8 м
- 1 верста – 500 саженьей
- 1 сажень – 3 аршина
- 1 аршин – 16 вершков
- 1 вершок – 4,45 см
- 1 фут – 12 дюймов
- 1 дюйм – 0,254 см
- 1 морская миля – 1852,2 м
- 1 пуд – 16,38 кг



4.8. Программирование составных условий



В условном операторе `if` указывается условие, в зависимости от истинности которого выполняется тот или другой оператор. Различают два вида условий – *простые* и *составные*.

ЗАПОМНИТЕ



Простым условием называется выражение, составленное из двух арифметических выражений или двух текстовых величин (иначе их еще называют операндами), связанных одним из знаков:

- < – меньше, чем...;
- > – больше, чем...;
- <= – меньше, чем... или равно;
- >= – больше, чем... или равно;
- <> – не равно;
- = – равно.

Пример простого условия:

$A \geq D$

$X \leq 10$

$5 * X + 2 = 4$

ЗАПОМНИТЕ



Составное условие – это условие, состоящее из нескольких простых условий, связанных с помощью логических операций:

- not** – НЕ (отрицание, инверсия);
- and** – И (одновременное выполнение условий);
- or** – ИЛИ (выполнение хотя бы одного из условий).



В составном условии каждое простое условие заключается в скобки.
Пример составного условия:

$(X > 5) \text{ Or } (X < = -10);$

$(A > B) \text{ And } (B > C);$

$(Y > 20) \text{ And } (Y < 50).$

В языке C++ существует три логические операции: **И**, **ИЛИ**, **НЕ**. Логические операции образуют **составное условие** из нескольких простых (два или более) условий, которые упрощают структуру программы.

Для построения логических условий в табл. 4.5 приведены все логические операции языка программирования C++:

Таблица 4.5

Логические операции C++

Операции	Обозначение	Условие	Краткое описание
И	&&	$a == 3 \ \&\& \ b > 4$	Составное условие истинно, если истинны оба простых условия
ИЛИ		$a == 3 \ \ b > 4$	Составное условие истинно, если истинно хотя бы одно из простых условий
НЕ	!	$!(a == 3)$	Условие истинно, если a не равно 3

ПОЭКСПЕРИМЕНТИРУЙТЕ

На примере предыдущей программы примените логический оператор И. Какое значение выводит программа – **true** или **false**?

```
#include <iostream>
using namespace std;
int main()
{setlocale(0, "Rus");
  Int one, two;
  cin>>one>>two;

  if ((one==1) && (two==2))
  cout<<"true";
  else cout<<"false";
}
```

Составное
условие



Примените логический оператор **ИЛИ**.

```
#include <iostream>
using namespace std;
int main()
{setlocale(0,"Rus");
int one, two;
cin>>one>>two;
if ((one==1) || (two==2))
cout<<"true";
else cout<<"false";
}
```

Составное
условие

Логический оператор **НЕ**.

```
#include <iostream>
using namespace std;
int main()
{setlocale(0,"Rus");
int one;
cin>>one;
if (one!=1)
cout<<one<<" - не единица ";
else cout<<one<<" - единица ";
}
```

Составное
условие

Когда выполняется в программе условие?

Уровень – Знание и понимание




1. Какое условие называется простым?
2. Какое условие называется составным?
3. Какие логические операции существуют в C++?
4. Если оба простых условия верны, какая логическая операция используется?
5. Если верно одно из условий, какая логическая операция используется?
6. Приведите примеры из окружающей среды с составным условием.




**Уровень – Применение**

Составьте четыре простых логических выражения с помощью оператора И, ИЛИ, НЕ.

**Уровень – Анализ**

Составьте программу, которая определяет, принадлежит ли точка X отрезку $[A;B]$. Если точка принадлежит отрезку $[A;B]$, то выводится ответ – Yes, а иначе – No.

**Уровень – Синтез**

Напишите программу, определяющую по координатам точки, в какой четверти системы координат она находится.

**Уровень – Оценивание**

Дано a, b, c – стороны предполагаемого треугольника. Требуется сравнить длину каждого отрезка с суммой двух других. Если хотя бы в одном случае отрезок окажется больше суммы двух других, *то треугольника с такими сторонами не существует.*



ТЕСТОВЫЕ ЗАДАНИЯ К РАЗДЕЛУ 4

Из предложенных альтернативных ответов выберите только один верный. Каждый правильный ответ оценивается в 1 балл. Всего 15 баллов.

1. Системы программирования:

- а) обеспечивают непосредственное решение пользовательских задач;
- б) позволяют создавать новые программы на языках программирования;
- в) обеспечивают работу всех аппаратных устройств компьютера и доступ пользователя к ним;
- г) представляют собой совокупность программ, использующихся для различных операций с документами;
- д) обеспечивают защиту от компьютерных вирусов.

2. Язык программирования:

- а) язык, предназначенный для записи компьютерных программ;
- б) естественный язык, предназначенный для записи компьютерных программ;
- в) формальный язык, предназначенный для записи компьютерных программ;
- г) машинный язык, предназначенный для записи компьютерных программ;
- д) английский язык, предназначенный для записи компьютерных программ.

3. Алгоритм – это:

- а) правила выполнения определенных действий;
- б) ориентированный граф, указывающий порядок исполнения некоторого набора команд;
- в) понятное и точное предписание исполнителю совершить последовательность действий, направленных на достижение поставленных целей;
- г) набор команд для компьютера;
- д) протокол вычислительной сети.

4. Алгоритм называется линейным:

- а) если он составлен так, что его выполнение предполагает многократное повторение одних и тех же действий;
- б) если ход его выполнения зависит от истинности тех или иных условий;



- в) если его команды выполняются в порядке их естественного следования друг за другом независимо от каких-либо условий;
- г) если он представим в табличной форме;
- д) если он включает в себя вспомогательный алгоритм.

5. Трансляторы – это:

- а) техническое средство, выполняющее трансляцию программы;
- б) программа или техническое средство;
- в) программа, переводящая информацию;
- г) программа или техническое средство, выполняющее трансляцию программы;
- д) трансляция программы.

6. Переменная в программировании полностью характеризуется:

- а) именем и типом;
- б) именем;
- в) именем и значением;
- г) значением;
- д) именем, значением и типом.

7. Данные, которые во время работы программы могут меняться многократно, называются:

- а) метками;
- б) переменными;
- в) константами;
- г) операций;
- д) функциями.

8. Характерным признаком линейной программы является:

- а) присутствие в ней операторов цикла;
- б) наличие в программной строке только одного оператора;
- в) использование в ней исключительно операторов присваивания;
- г) присутствие в ней операторов условного перехода;
- д) строго последовательное выполнение операторов в порядке их записи.

9. Термин «интерфейс» широко используется в технике. Интерфейсом называют:

- а) совокупность технических и программных средств, обеспечивающих сопряжение различных устройств между собой, а также возможность работать с этими устройствами;



- б) любая совокупность технических и программных средств;
- в) совокупность программных средств, обеспечивающая процесс общения человека с компьютером;
- г) определенная совокупность технических и программных средств;
- д) совокупность технических и программных средств, обеспечивающих возможность работать с компьютером.

10. Программа включает в себя ветвление, если:

- а) она составлена так, что ее выполнение предполагает многократное повторение одних и тех же действий;
- б) ход ее выполнения зависит от истинности тех или иных условий;
- в) ее команды выполняются в порядке их естественного следования друг за другом независимо от каких-либо условий;
- г) она представима в табличной форме;
- д) она включает в себя вспомогательный алгоритм.

11. Алфавит языка – это:

- а) множество простейших знаков, которые могут быть использованы в текстах этого языка;
- б) английские буквы, которые могут быть использованы в текстах этого языка;
- в) специальные символы, которые могут быть использованы в текстах этого языка;
- г) ключевые слова, которые могут быть использованы в текстах этого языка;
- д) идентификаторы, которые могут быть использованы в текстах этого языка.

12. Какое условие называют простым:

- а) выражение, составленное из трех арифметических выражений или трех текстовых величин;
- б) выражение, составленное из двух арифметических выражений или двух текстовых величин;
- в) выражение, составленное из арифметического выражения или текстовой величины;
- г) выражение, составленное из четырех арифметических выражений или четырех текстовых величин;
- д) выражение, составленное из арифметических и текстовых величин?



13. Составное условие – это:

- а) условие, состоящее из нескольких простых условий, связанных с помощью логических операций;
- б) условие, состоящее из простого условия;
- в) условие, состоящее из нескольких простых условий;
- г) выражение, состоящее из двух арифметических выражений или двух текстовых величин;
- д) выражение, состоящее из нескольких арифметических выражений или нескольких текстовых величин.

14. Интегрированная среда разработки (ИСР) – это:

- а) система программных средств, используемая программистами;
- б) программная среда, используемая программистам для разработки программного обеспечения;
- в) система программных средств;
- г) система программных средств для игр;
- д) система программных средств, используемая программистами для разработки программного обеспечения.

15. Укажите символьный тип данных:

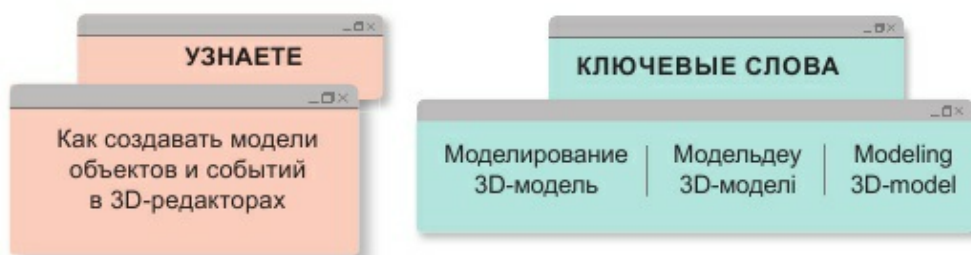
- а) Float;
- б) Int;
- в) Char;
- г) Bool;
- д) Double.



Раздел 5

МОДЕЛИРОВАНИЕ ОБЪЕКТОВ
И СОБЫТИЙ

5.1. Трехмерные модели



С понятием «моделирование» вы знакомы из темы «Моделирование процессов в электронных таблицах». Вам известно, что когда исследование некоторого **объекта** вызывает затруднения, требует длительного времени, стоит дорого, то возникает потребность в **моделировании**.

ЗАПОМНИТЕ



Объект – это некоторая часть окружающего мира.
Модель – упрощенное представление об объекте.

Между моделью и объектом, интересующим исследователя, должно существовать подобие:

- в сходстве **физических характеристик** модели и объекта,
- в сходстве **функций**, осуществляемых моделью и объектом,
- в **тождестве математического описания** «поведения» объекта и его модели.

Окружающий нас мир многообразен. Это 3D-мир, который прочно вошел во многие сферы нашей жизни. 3D (от английского слова **3-Dimensional**) – объект, имеющий три измерения. Поэтому актуальность 3D-технологий очевидна. Такие модели, как здания, самолеты, автомо-



били, бытовая техника, ручки, микросхемы, компьютеры, – это модели, созданные с использованием 3D-технологий.

ЗАПОМНИТЕ



3D-моделирование – это процесс создания трехмерной модели объекта. Задача 3D-моделирования – разработать визуальный объемный образ желаемого объекта.

Сегодня современные технологии позволяют нам легко перейти к моделированию окружающего нас мира – 3D-мира.

ПОДУМАЙТЕ И ОТВЕТЬТЕ

Почему 3D-мир интересен для детей разного возраста? Какую дополнительную мотивацию дает учащимся 3D-моделирование?

Объекты, выполненные с помощью 3D, позволят увидеть то, что еще не существует. Как вы думаете, будет ли 3D отличным инструментом:

- для виртуальных музеев;
- для игровой индустрии;
- для строительства;
- для телевидения;
- для студий дизайна интерьера;
- для рекламы;
- для полиграфии;
- для промышленных предприятий;
- в самолетостроении, кораблестроении?

3D-графика, или трехмерная графика, – это один из разделов компьютерной графики. С помощью комплекса приемов и инструментов можно создать объемные объекты, имеющие форму и цвет. 3D-графика подразумевает построение трехмерной модели на плоскости. Выполняется это с помощью специализированных компьютерных программ:

- Punch Home Design
- SketchUp (Google SketchUp)
- ArCon
- Realtime Landscaping Plus
- Sweet Home 3D
- Terragen



- Turbo FLOORPLAN Landscape and Deck 12
- FloorPlan 3D 12
- Autocad
- 3ds Max
- Sierra Land Designer 3D 7.0
- ArchiCad 17
- Blended и др.

Для моделирования объектов трехмерной графики вы можете выбрать любой из указанных 3D-редакторов.

ЗАПОМНИТЕ



3D-редактор – программная среда, обладающая возможностями для трехмерного моделирования, анимации и визуализации.

Мы познакомимся с основными приемами моделирования простых трехмерных объектов окружающей среды. Для моделирования трехмерного изображения создадим информационную модель с ее описанием в программной среде. При этом необходимо знать **этапы** моделирования (рис. 5.1):



Рис. 5.1. Этапы выполнения модели трехмерного изображения



Сегодня мы видим, как быстро развиваются технологии. 3D-графика, 3D-принтер становятся неотъемлемой частью науки и техники. В настоящее время мы можем не только создавать трехмерную графику, но и напечатать абсолютно все, используя 3D-принтеры. И даже можно придавать напечатанному вкус яблока, мяты, арбуза и др.

Уровень – Знание и понимание



1. Вспомните, каким образом вы проводили моделирование в электронных таблицах.
2. Что вы понимаете под словами «модель», «объект»?
3. Что означает 3D-моделирование?
4. В каких случаях применяется программная среда 3D-редактор?
5. Какие существуют программы для 3D-моделирования?
6. Перечислите основные этапы трехмерного моделирования.
7. Что вы знаете о 3D-принтерах?

Уровень – Применение



1. Угадайте слово в ребусе (рис. 5.2).



Рис. 5.2. Ребус

2. Выберите один правильный ответ из теста.

1. Модель – это:

- а) изображение объекта, процесса или явления на бумаге;
 - б) уменьшенное представление о реальном объекте, процессе или явлении;
 - в) упрощенное представление о реальном процессе, объекте или явлении;
 - г) уменьшенный объект, выполненный из любого материала.
3. На каком этапе моделирования производится анализ объекта и определение цели моделирования:
 - а) разработка модели;



- b) постановка задачи;
- c) компьютерный эксперимент;
- d) анализ результатов моделирования?

4. Совокупность сведений об объекте, процессе или явлении называется:

- a) знаковой моделью;
- b) компьютерной моделью;
- c) информационной моделью;
- d) вербальной моделью.

5. Рассмотрите рис. 5.3.

Какого вида модель представлена на нем?

- a) игровая;
- b) учебная;
- c) имитационная;
- d) научно-техническая;
- e) опытная?



Рис. 5.3. Модель объекта

6. Этапы построения модели трехмерного изображения, как правило, предполагают:

- a) описание всех пространственно-временных характеристик изучаемого объекта;
- b) моделирование, раскраску, освещение, съемку, рендеринг;
- c) выделение свойств объекта относительно раскраски, освещения и рендеринга;
- d) описание свойств съемки и раскраски исследуемого объекта;
- e) выделение не более трех существенных признаков объекта.



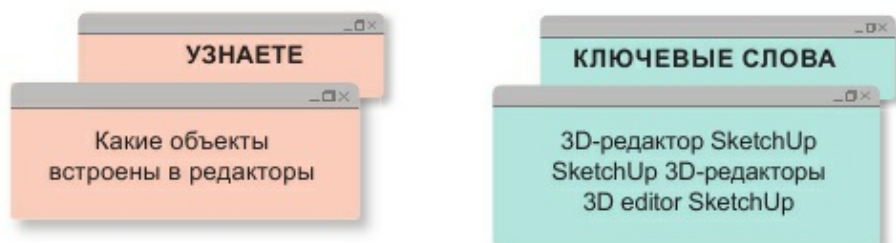
Уровень – Анализ, синтез и оценивание



Выполните поисковую работу в группе, используя различные источники информации: энциклопедии, интернет-ресурсы. Напишите эссе по теме «Все самое интересное о 3D-мире». Проведите анализ, синтез и оценивание новой информации. Классифицируйте компьютерные программы для трехмерной графики.



5.2. Объекты, встроенные в редактор



Для создания и редактирования трехмерной графики мы будем изучать программу **Google SketchUp**. Программная среда **SketchUp** – простой инструмент для создания, обработки и презентации трехмерных моделей. Кроме того, Google SketchUP предоставляет возможность создавать многостраничные документы и презентации. Множество моделей можно раскладывать на одной странице. Программная среда идеально подходит для моделирования 3D-объектов.

ЗАПОМНИТЕ



Sketch Up 3D можно использовать для:

- эскизного моделирования в архитектуре;
- моделирования существующих зданий;
- моделирования исчезнувших зданий;
- дизайна интерьера;
- ландшафтного дизайна;
- дизайна наружной рекламы;
- моделирования изделий для печати на 3D-принтере и т. д.

ВЫПОЛНИМ ВМЕСТЕ

Программу **SketchUp** скачайте с сайта и установите на компьютер.

Посмотрите видеофильмы о том, как работать в SketchUp.

Запустите программную среду по алгоритму:

1. Нажмите на кнопку **Пуск** и выберите **Все программы** (или с помощью ярлыка на рабочем столе).



2. В диалоговом окне **Добро пожаловать в SketchUp** выберите **Выбрать шаблон**. Из набора шаблонов выберите наиболее подходящий из представленных вариантов, например простой шаблон (метры).

3. Нажмите на кнопку **Начать использование SketchUp**.

Изучите интерфейс программы **SketchUp**. Рабочее окно программы (рис. 5.4) разбито на несколько областей:

- **меню** – панель команд или панель управления;
- **область моделирования** занимает основную часть рабочего стола программы, где и происходит весь процесс моделирования;
- **строка заголовка**, расположенная в верхней части экрана, показывает имя текущего открытого файла;
- **панели инструментов**, расположенные горизонтально ниже меню, содержат группы кнопок самых «востребованных» в работе инструментов и опций;
- **строка состояния** находится ниже области моделирования;
- в левой части панели находятся кнопки нескольких опций идентификации проекта и **окно инструктора**, справочное окно по основным приемам работы с программой SketchUp.

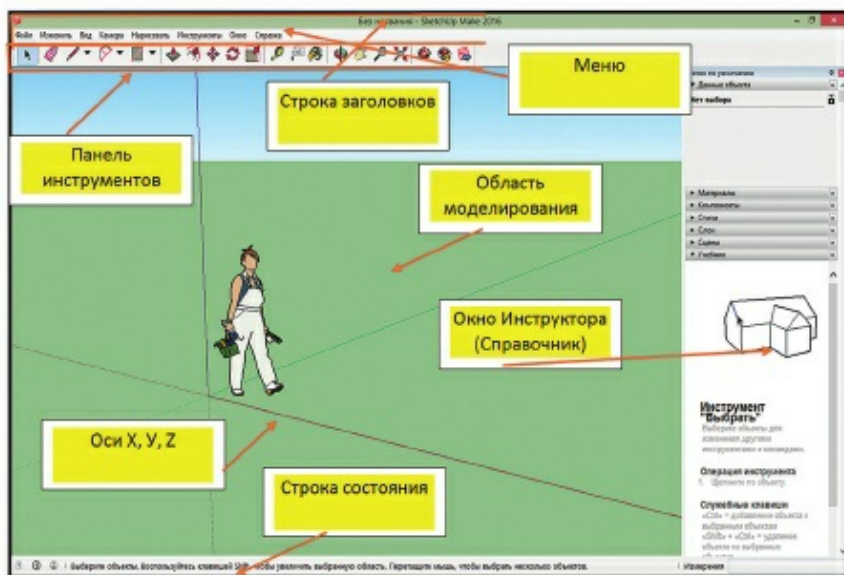


Рис. 5.4. Рабочее окно программы SketchUp



В трехмерном пространстве координаты объектов задаются в привычной декартовой системе координат. Оси **X**, **Y** и **Z** в программе обозначены соответственно красной, зеленой и синей линиями в области рисования. Они помогают во время работы контролировать положение в 3D-пространстве сцены.






На пересечении оси появляется **человек**, который позволяет оценить реальные размеры конструкции. Человека можно скрыть, с помощью контекстного меню.

ПОЭКСПЕРИМЕНТИРУЙТЕ


Панель инструментов содержит 20 кнопок (табл. 5.1). Назначение каждой кнопки можно прочесть на всплывающей подсказке, появляющейся при наведении на нее указателя мыши. Попробуйте использовать инструменты (вдавить, переместить, ластик, дуги, измерить расстояние и т. д.), чтобы лучше узнать их назначение.

Таблица 5.1

Панель инструментов

Инструмент	Назначение	Выполняемая операция
1	2	3
	Выбрать	Выбрать объекты для изменения другими инструментами и командами
	Ластик	Стереть, сгладить или смягчить выбранные модели объекты
	Линии	Рисовать линии
	Дуги	Рисовать дуги
	Фигуры	Выбрать фигуры: прямоугольник, повернутый прямоугольник, круг, многоугольник
	Вдавить и вытянуть	Вдавливать и вытягивать грани объектов
	Сдвиг	Сместить выбранные ребра в плоскости



1	2	3
	Переместить	Позволяет перемещать, растягивать, копировать и упорядочивать объекты
	Повернуть	Повернуть, растянуть, скопировать или упорядочить объекты относительно оси
	Масштаб	Масштабирование и растягивание выбранных объектов
	Рулетка	Измерение расстояний, создание направляющих линий или точек, а также масштабирование модели
	Текст	Рисует текстовые подписи
	Заливка	Применить цвет и материал к объектам в модели
	Вращение	Вращение поля зрения камеры вокруг модели
	Панорама	Поворот точки обзора камеры по вертикали или горизонтали
	Масштаб	Приближение или удаление точки камеры
	Масштабирование	Приближение или удаление точки камеры для показа всей модели
	3D Warehouse	Открыть 3D Warehouse
	Extension Warehouse	Добавить расширение в SketchUp
	Layout	Отправить в layout



ВЫПОЛНИТЕ

Создайте 3D-модель «прямоугольника». Создание начните с этапа моделирования.

1. Для этого постройте геометрический объект на сцене в трехмерном пространстве. На Панели инструментов выберите **Фигуры – Прямоугольник**. Нарисуйте прямоугольник от центра сцены, растягивая до подходящей величины (рис. 5.5).

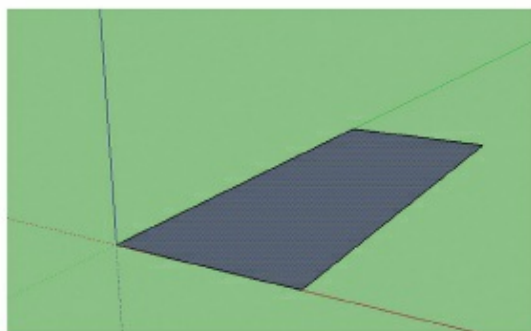


Рис. 5.5. Инструмент **Прямоугольник**

2. Поверхность **Прямоугольника** поднимите инструментом **Вдавить и вытянуть** на нужную высоту – получите объем прямоугольника (рис. 5.6).

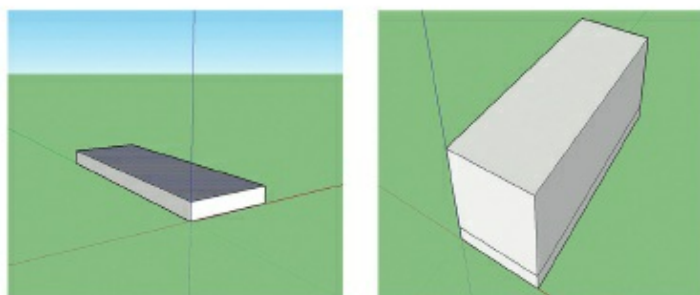


Рис. 5.6. Построение цоколя и стен

3. Попробуйте инструментом **Смещение** движением внутрь обозначить внутреннюю границу прямоугольника.

4. Инструментом **Вдавить и вытянуть** продавите внутреннюю поверхность вниз.



5. Инструментом **Вращение** посмотрите модель прямоугольника. Лишние объекты можно удалить инструментом **Ластик**.
6. Попробуйте с помощью геометрических фигур нарисовать «робота».
7. Сохраните через меню **Файл**. Выберите команду **Сохранить**, введите название модели и нажмите на кнопку **Сохранить**.
Проекты SketchUp сохраняются в формате *.skp.

Уровень – Знание и понимание



1. Какая программная среда используется для создания и редактирования трехмерной графики?
2. Какие возможности программы SketchUp вы знаете?
3. Как можно запустить программу SketchUp?
4. Чем отличается интерфейс программы SketchUp от графического редактора Paint?
5. На какие области делится окно редактора SketchUp?
6. Почему оси X, Y и Z в программе обозначены разными цветами?
7. Какие инструменты вы считаете наиболее часто используемыми?
8. Почему возникла необходимость создания 3D-редакторов в современном мире?
9. Как вы представляете использование 3D-редакторов в своей жизни?
10. С каким расширением сохраняются проекты в SketchUp?




Уровень – Применение



1. Установите соответствие стрелочками между инструментом и выполняемой операцией в табл. 5.2. Впишите в столбец «Назначение» назначение инструмента.

Таблица 5.2

Инструменты программы SketchU

Инструмент	Назначение	Выполняемая операция
1	2	3
		Рисовать линии
		Выбрать объекты для изменения другими инструментами и командами
		Стереть, сгладить или смягчить выбранные модели объектов



Продолжение

1	2	3
		Вдавливать и вытягивать грани объектов
		Сместить выбранные ребра в плоскости
		Рисовать дуги
		Выбрать фигуры: прямоугольник, повернутый прямоугольник, круг, многоугольник
		Позволяет перемещать, растягивать, копировать и упорядочивать объекты
		Повернуть, растянуть, скопировать или упорядочить объекты относительно оси
		Масштабирование и растягивание выбранных объектов
		Измерение расстояний, создание направляющих линий или точек, а также масштабирование модели
		Рисует текстовые подписи
		Применить цвет и материал к объектам в модели
		Приближение или удаление точки камеры
		Поворот точки обзора камеры по вертикали или горизонтали
		Вращение поля зрения камеры вокруг модели



Окончание

1	2	3
		Добавить расширение в SketchUp
		Отправить в layout
		Приближение или удаление точки камеры для показа всей модели
		Открыть 3D Warehouse

2. В программной среде SketchUp нарисуйте «домик» (рис. 5.7), предварительно просмотрев видеофильм по адресу: <https://www.youtube.com/watch?v=Bfk8gGHVbDs>.

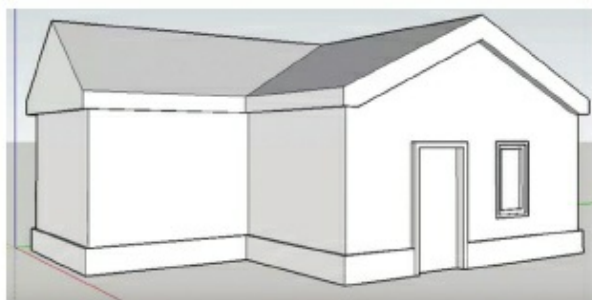


Рис. 5.7. Домик

Уровень – Анализ



Сделайте анализ изученного интерфейса программной среды SketchUp. В чем преимущества и недостатки пользовательского интерфейса программы?

Уровень – Синтез

Проведите сравнение (рис. 5.8): чем отличается 2D-графика от 3D-графики из представленных рисунков?



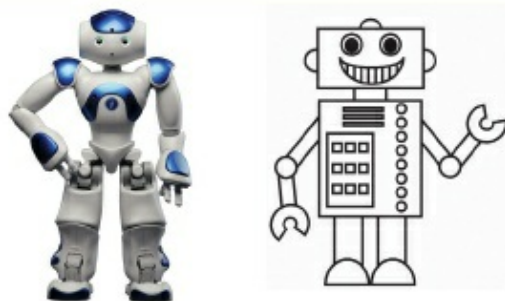
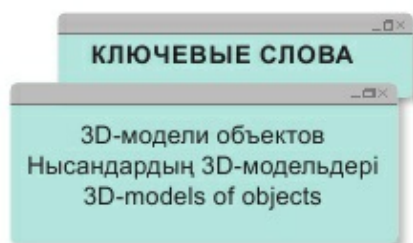
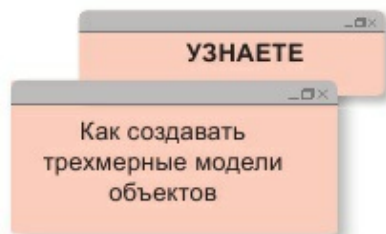


Рис. 5.8. Изображение роботов

Уровень – Оценивание

Для эффективной работы с 3D-графикой вам необходимо выбрать компьютер. В представленном списке: корпус, блок питания, процессор, видеокарта, системный блок, оперативная память, система охлаждения компьютера, винчестер вам нужно ранжировать устройства по степени важности при работе с трехмерной графикой.

5.3. Трехмерные модели объектов



ВЫПОЛНИМ ВМЕСТЕ

Создадим модель – 3D-дом.
Строим стены, окна, дверь.

Для моделирования дома первоначально создайте геометрическую фигуру прямоугольник с помощью Карандаша на Панели инструментов и сделайте его объемным.



Проемы (окна, двери) нарисуйте по наружной поверхности инструментом **Прямоугольник**, а затем инструментом **Вдавить и вытянуть** продавите их до внутренней поверхности (рис. 5.9).

Инструментом **Вращение** рассмотрите модель. Излишние объекты удалите инструментом **Ластик**.

Для копирования одинаковых объектов выберите рамкой слева направо все элементы и, используя возможности опции **перемещение + Ctrl**, получите копии.

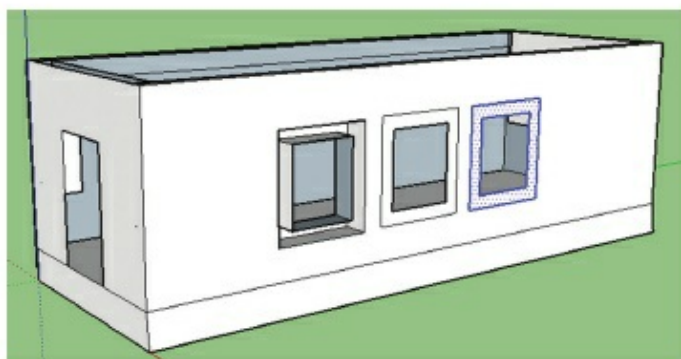


Рис. 5.9. Рисование проемов

Строим крышу.

Строим прямоугольник потолка. Для этого делаем **Смещение**, расширяя на величину карниза (рис. 5.10, 5.11). Поднимаете инструментом **Вдавить и вытянуть**. Удаляете лишние ребра с использованием инструмента **Ластик**.

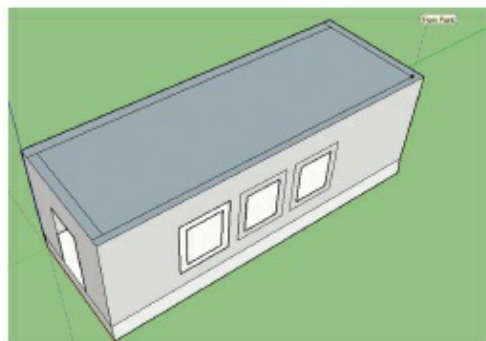


Рис. 5.10. Стройка прямоугольника потолка

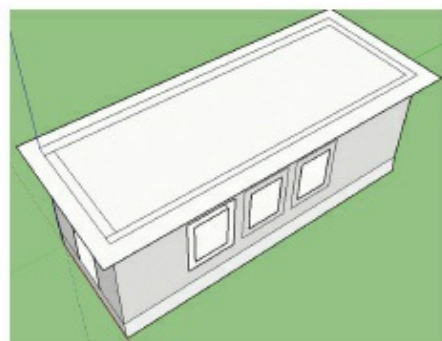


Рис. 5.11. Величина карниза



Используя инструмент **Линия**, проведите отрезок ровно посередине верхней грани, на это укажет появившаяся голубая точка, взамен красной. Ориентируясь по подсказке, разделите поверхность на две равные части. Используя инструмент **Перемещение**, поднимите кровлю вверх (рис. 5.12, 5.13).

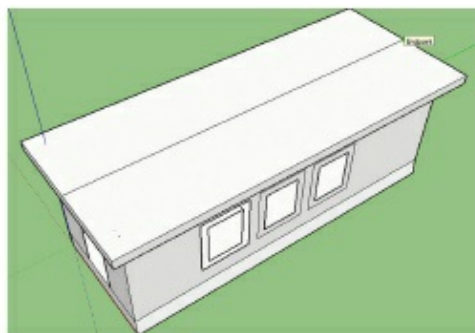


Рис. 5.12. Разделение на две части

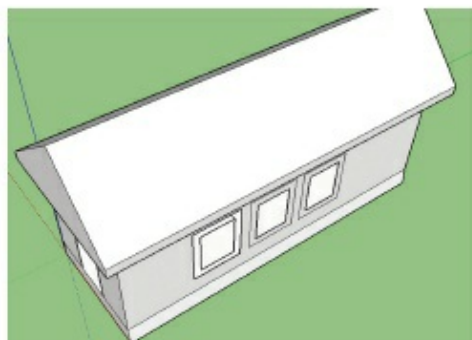


Рис. 5.13. Кровля

Используя инструменты **Линия**, **Прямоугольник**, **Вдавить и вытянуть**, **Перемещение**, добавьте недостающие элементы дома. Например, лестницу, крыльцо и т. д.

Строим дымоход и измеряем дом.

Выберите инструмент **Дуга** – щелкните в начальной точке и растяните на длину хорды. Щелкните в конечной точке. Переместите курсор перпендикулярно хорде на нужную высоту подъема и третьим щелчком завершите построение.

Дополните линию, используя инструмент **Линия**. Применяя инструмент **Вдавить и вытянуть**, вытяните образовавшийся объект вверх (рис. 5.14).

Чтобы измерить дом, подведите указатель мыши к инструменту **Рулетка**. Нажмите кнопку мыши в начальной точке измерения. Переместите указатель. Нажмите кнопку мыши в конечной точке измерения.

После выбора инструмента **Повернуть** стандартный курсор изменится на изображение инструмента **Транспортир** с круговыми стрелками. Работа **Транспортира**, как и других инструментов модификаций, контролируется точными числовыми измерениями, точнее – угловыми величинами.



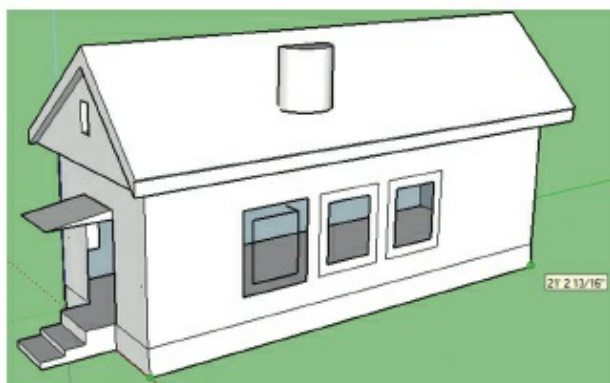


Рис. 5.14. Дымоход

Инструмент **масштабирование** используется для изменения размеров и пропорций дома целиком и их отдельных элементов для искажения.

Уровень – Знание и понимание



Вопросы	Ответы
Сколько координат имеет точка в трехмерном пространстве?	
Как называется плоский объект, ограниченный ребрами?	

Уровень – Применение



1. Дальнейшее оформление и дизайн «дома» напрямую зависят от творческого подхода. Фигура приобретает те цвета и ориентацию, которые ей придадут с помощью доступного и удобного инструментария.

Для выбора стиля дома выберите команду **Стили** и сам стиль. На рис. 5.15 выбраны несколько видов стилей.

Под **Материалами** в SketchUp понимается окраска поверхностей в однотонные цвета и помещение на них текстур.



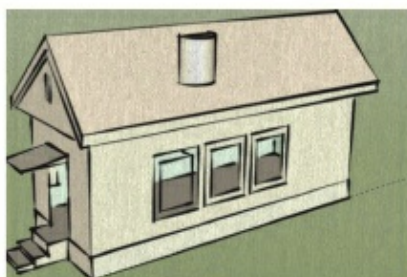


Рис. 5.15. Стиль



Рис. 5.16. Материал

Для подбора материала постройки дома выберите инструмент **Заливка**. Откроется диалоговое окно **Материалы** (рис. 5.16). В нем содержится коллекция включенных в программу материалов, используемых в текущем проекте.

Управление слоями осуществляется из диалогового окна, которое открывается через меню **Layers** и показывает все слои, созданные на данном этапе в проекте. В окне **Layers** в нашем примере в списке слоев – *Layer 0 и стены* (рис. 5.17).

Слои в SketchUp удобно использовать для:

- 1) управления видимостью объектов;
- 2) отображения различных объектов в сцене по цвету слоя;
- 3) экспорта 2D-геометрии в dwg формат, а также 3D-форматы.

Сначала моделирование геометрических фигур проходит в *Layer0*. В каждой сцене сохраняют различные настройки: положение камеры, тени, стиль отображения, видимость слоев, видимость конкретных объектов, положение осей координат и т. п.

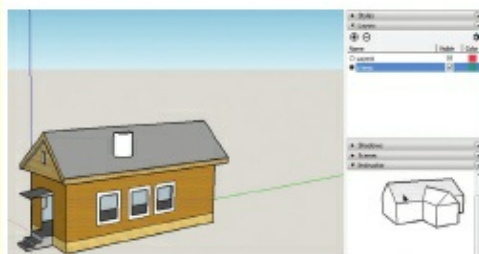


Рис. 5.17. Слой

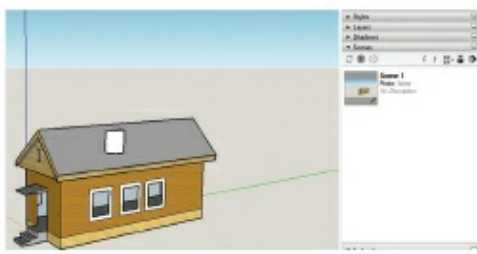


Рис. 5.18. Сцена



После открытия диалогового окна **Сцена** появится панель списка сцен в проекте – в нашем случае *одна сцена* (рис. 5.18). В проекте можно создавать несколько сцен. В каждой из них кроме общего вида можно сохранить фундамент, стены, крышу и др. В каждой из сцен сохраните нужные виды проекции, назначьте стили рендеринга, тени и т. д.

Вопросы	Ответы
<p>Инструмент  Тяни/Толкай используется для того, чтобы подтянуть или вытолкнуть грани, для увеличения или уменьшения объема модели. Сколько на рис. 5.19 граней, которые можно «вдавить» или «вытолкнуть»:</p>	
	
<p>Рис. 5.19. Грани</p>	
Чтобы увеличить масштаб просмотра изображения, нужно...	
Чтобы изменить угол просмотра изображения, нужно...	



Уровень – Анализ



Вопросы	Ответы
Что входит в основные элементы интерфейса Google SketchUp?	
Выбери наиболее часто используемые инструменты в программе Google SketchUp	



Какие можно получить фигуры из первичной фигуры с помощью инструмента  Тяни/Толкай (рис. 5.20)?

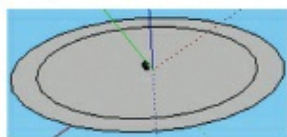


Рис. 5.20. Первичная фигура

Во время работы модель дома выпала из зоны видимости дизайнера. Какой инструмент ему лучше использовать в данной ситуации?



Уровень – Синтез

Напишите, какие инструменты нужно использовать, чтобы создать модель дома, изображенную на рис. 5.21.



Рис. 5.21. Модель дома



**Уровень – Оценивание**

1. Компьютерные игры в большинстве имеют трехмерный графический интерфейс. Приведите примеры игр, где компьютерная графика достаточно сложная, но дружелюбный пользовательский интерфейс.
2. Создайте 3D-модели геометрических фигур, изображенных на рис. 5.22.

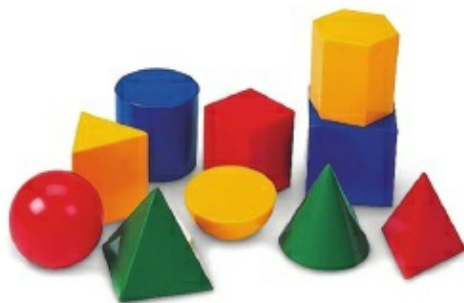


Рис. 5.22. Объемные геометрические фигуры

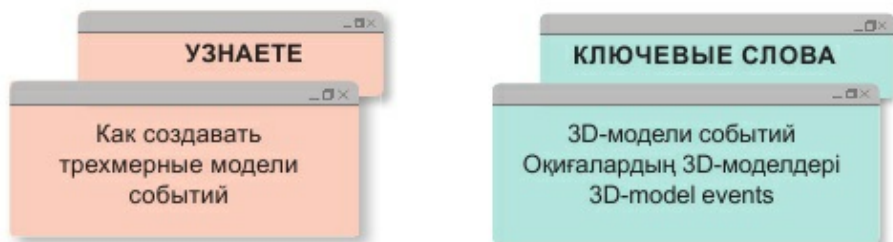
3. Создайте модель набора карандашей (рис. 5.23).



Рис. 5.23. Набор карандашей



5.4. Трехмерные модели событий



Работа с компьютерной графикой – одно из самых популярных направлений использования персонального компьютера. Для сферы обучения средства компьютерной графики открывают принципиально новые возможности. В процессе анализа изображений можно динамически управлять их содержанием, формой, размерами и цветом, добиваясь наибольшей наглядности.

Например, моделирование в **Google Sketch Up** ландшафта (рис. 5.24) с помощью инструмента **Песочница (Sandbox)**, основано на технологии создания и манипулирования поверхностями (рельефами). Поверхности моделируются в виде сетки с плоскими треугольными ячейками.

В **Песочницу (Sandbox)** входят следующие инструменты:



Из контуров (from Contours) – используется для создания рельефа, формируемого линиями, расположенными на разной высоте.



Из царапины (From Scratch) – используется для создания новой сетки «с нуля».



Штамп (Stamp) – решает задачу «посадки» или «врезки» в рельеф объекта.



Драпировка (Drape) – «прорезает» поверхность рельефа, образуя новые поверхности.



Добавить детали (Add Detail) – для выборочной детализации за счет дробления ячеек сетки на более мелкие треугольные ячейки.





Присоска (Stooon e) – для вытягивания (лепки) рельефа за счет деформации заранее подготовленной сетки.



Отразить ребро (Flip Edge) – для исправления нежелательных «переломов» рельефа.

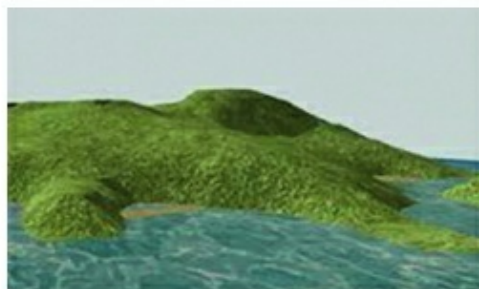


Рис. 5.24. Ландшафтный дизайн

ВЫПОЛНИТЕ

Как из 2D-графики получить 3D-графику? Например, из рисунков снежинки или дерева (без фона) с расширением .jpg создайте 3D-модель. Предварительно просмотрите видеофильм из Youtube.com – как сделать из 2D-графики 3D-графику в SketchUp Google.

Мы живем в трехмерном пространстве. Для создания трехмерной модели требуются специальные программные и аппаратные средства. К программным средствам относятся приложения 3D-визуализации, о которых пойдет речь ниже. К аппаратным относят то, с помощью чего создается и отображается модель (компьютер, 3D-мониторы, 3D-принтеры).

ЗАПОМНИТЕ



Событие (в информатике, в программном обеспечении) – действие, которое может быть инициировано пользователем, устройством типа таймера, клавиатурой или даже операционной системой.



Данные, связанные с событием, как минимум, определяют, что это за событие, но могут включать и другую информацию. Например, когда оно произошло, кто или что заставило его произойти, как событие должно быть обработано.

Компьютерная программа, меняющая свое поведение в ответ на события, управляется событием и является **интерактивной**.

В настоящее время создано множество интерактивных моделей. Например, интерактивная 3D-модель Солнечной системы **Solar System Scope** (рис. 5.25) позволяет совершать виртуальные путешествия, измерять расстояния и размеры планет, используется в качестве практического источника для показа и объяснения событий в нашей Солнечной системе.



Рис. 5.25. 3D-модель Солнечной системы

Как можно самостоятельно разработать трехмерные модели событий? Какими 3D-редакторами можно воспользоваться?

Практически всем, кто занимается трехмерной визуализацией, приходится сталкиваться с необходимостью наполнять сцену растительностью. В этом случае можно использовать специальные программы для генерации растений. Программа **Xfrog** позволяет генерировать и анимировать растения, объекты, архитектурные формы, изменяя отдельные их параметры. То есть можно получить анимацию роста любого растения: цветка, дерева, архитектурного объекта.

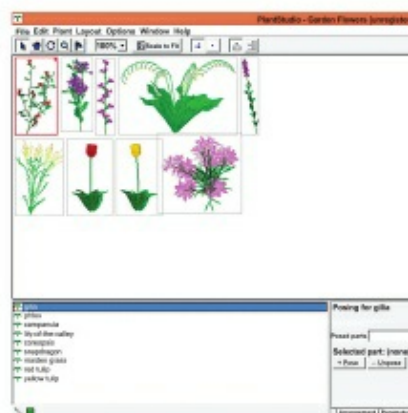


Рис. 5.26. 3D-редактор PlantStudio

PlantStudio имитирует растения, полевые цветы, овощи, водоросли, травы и др. С помощью данной программы можно «вырастить» растение, создавая реалистичные изображения в любом возрасте (рис. 5.25).

Программа **ngPlant** (интерактивный инструмент) используется для создания 3D-модели различных растений и деревьев. Интерфейс программы довольно прост. Вы можете **поворачивать/панорамировать** вид кнопками мыши, **выбирать/добавлять/удалять** слои ветвей из списка.

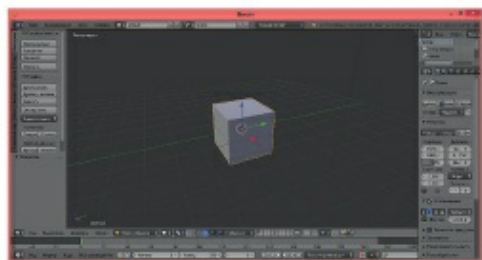


Рис. 5.27. Интерфейс программы Blender

Одной из популярных компьютерных программ для создания трехмерной графики является **Blender** (рис. 5.27). С ее помощью также создают анимацию и интерактивные приложения. Blender обладает двумя видимыми преимуществами. Во-первых, это свободно распространяемый под лицензией GNU (General Public License) продукт. Во-вторых, в нем есть движок, позволяющий создавать анимацию реального времени (интерактивные модели). Чтобы рассмотреть, как создавать интерактивные модели, попробуйте выполнить самостоятельное исследование в приложении Blender.

Во-вторых, в нем есть движок, позволяющий создавать анимацию реального времени (интерактивные модели). Чтобы рассмотреть, как создавать интерактивные модели, попробуйте выполнить самостоятельное исследование в приложении Blender.

Уровни – Знание, Понимание, Применение, Анализ, Синтез, Оценивание

Темы проектов по 3D-моделированию

Тема 1

- Для создания 3D-модели выберите любую программу: Google SketchUp, Sweet Home 3D, Terragen и др. Постройте юрту по рис. 5.28 (отделку подберите из имеющейся текстуры).
- Сохраните модель в рабочий файл.
- Сделайте две-три визуализации наиболее удачного ракурса созданной работы, сохраните их в формате jpg.
- Объясните, чем эта программа интересна для вас.



Рис. 5.28. Юрта



Тема 2

– Для создания 3D-модели выберите любую программу: Google SketchUp, Sweet Home 3D, Terragen и др. Постройте модель символа Астаны – монумента Байтерек (рис. 5.29). Байтерек олицетворяет древо жизни. В его высокой кроне находится золотое яйцо – Солнце, дающее жизнь и надежду.

- Сохраните модель в рабочий файл.
- Сделайте две-три визуализации наиболее удачного ракурса созданной работы, сохраните их в формате jpg.
- Объясните, чем эта программа интересна для вас.



Рис. 5.29. Монумент Байтерек

Тема 3

– Для создания 3D-модели выберите любую программу: Google SketchUp, Sweet Home 3D, Terragen и др. По заданной планировке разработайте дизайн-интерьер коттеджа (рис. 5.30).

- Сохраните модель в рабочий файл.
- Сделайте две-три визуализации наиболее удачного ракурса созданной работы, сохраните их в формате jpg.
- Объясните, чем эта программа интересна для вас.



Рис. 5.30. Планировка коттеджа



Тема 4

– Для создания 3D-модели выберите любую программу: Google SketchUp, Sweet Home 3D, Terragen и др. Создайте красочный ландшафт каньона Чарын, искусственный и без растительности (рис. 5.31).



Рис. 5.31. Каньон Чарын

- Сохраните модель в рабочий файл.
- Сделайте две-три визуализации наиболее удачного ракурса созданной работы, сохраните их в формате jpg.
- Объясните, чем эта программа интересна для вас.

Тема 5

– Для создания 3D-модели выберите любую программу: Google SketchUp, Sweet Home 3D, Terragen и др. Создайте красочный ландшафт Большого Алматинского озера (рис. 5.32).



Рис. 5.32. Большое Алматинское озеро

- Сохраните модель в рабочий файл.
- Сделайте две-три визуализации наиболее удачного ракурса созданной работы, сохраните их в формате jpg.
- Объясните, чем эта программа интересна для вас.

Требования к отчету по проектам 3D-модели

Отчет по проекту должен содержать:

- цель работы;
- задачи работы;
- использованное программное обеспечение;
- описание использованных инструментов;
- результаты работы с приложением графического материала;
- в качестве отчета по данному исследовательскому заданию подготовить компьютерную презентацию из 3–5 слайдов, содержащих минимум текста.





Анимация

Анимация – это очень широкое понятие. Существует классическая компьютерная анимация – анимация по ключевым кадрам. Отдельный раздел трехмерной графики – персонажная анимация, моделирование физических процессов. В Blender представлено такое разнообразие инструментов, что их хватит для решения любой задачи, связанной с анимацией.

Классический прием создания анимации – по ключевым кадрам. Чтобы анимировать сцену таким способом, выберите положение объекта в сцене, установите положение бегунка на временной шкале **Timeline** и нажмите клавишу **I**. Далее в контекстном меню выберите анимируемый параметр и укажите команду для создания ключа анимации (рис. 5.33).

Простой способ анимировать объекты сцены – в режиме автоматического создания ключевых кадров. Этот режим активируется нажатием красной кнопки, которая расположена возле кнопок для управления воспроизведением анимации. Более тонкая настройка анимации выполняется в окне Graph Editor. Выделив маркер, в настройках можно изменить тип излома кривой, оказав влияние на характер протекания анимации в ключевой точке.



Рис. 5.33. Классический прием





Средство для управления связанными параметрами – driver. Настройка драйверов выполняется с помощью редактора графов, переключенного в режим Drivers (рис. 5.34).



Рис. 5.34. Drivers

Аниматор имеет возможность моделировать взаимодействия между «твердыми» объектами, а также составлять конструкции. Например, соединять повторяющиеся звенья цепи. Решение физических задач может также включать такие особенности взаимодействий, как разрушение конструкций (опция Breakable), геометрия поверхности соударения, определение внешних сил и пр. Для этого нужно выделить объект, перейти на вкладку Physics, нажать на кнопку **Rigid Body** и указать тип объекта: подвижный – active или неподвижный – passive (рис. 5.35). Простейшая анимация готова.



Рис. 5.35. Rigid Body



ГЛОССАРИЙ

- ⇓ **3D-моделирование** – процесс создания трехмерной модели объекта. Задача 3D-моделирования – разработать визуальный объемный образ желаемого объекта.
- ⇓ **3D-редактор** – программная среда, обладающая возможностями для трехмерного моделирования, анимации и визуализации.
- ⇓ **BIOS** – постоянная память, т. е. память, хранящая информацию при отключенном питании. В нее данные заносятся при изготовлении.
- ⇓ **Google Sketch Up 3D** – программная среда для создания и редактирования трехмерной графики.
- ⇓ **CMOS-память** – предназначена для длительного хранения данных о конфигурации и настройке компьютера (дата, время, пароль).
- ⇓ **Абзац** – набор символов, заключенных между знаками абзаца (¶). Ставятся при нажатии клавиши Enter.
- ⇓ **Автозаполнение** – операция ускоренного автоматического заполнения ячеек данными, текстом, числами или формулами без использования обычных способов ввода.
- ⇓ **Алфавит языка** – множество простейших знаков, которые могут быть использованы в текстах этого языка.
- ⇓ **Антивирус** – программа для обнаружения и «лечения» программ, зараженных компьютерным вирусом, а также для предотвращения заражения файлов вирусом.
- ⇓ **Архив** – файл, содержащий в себе один или несколько файлов в сжатом виде.
- ⇓ **Архиваторы или средства сжатия данных** – служебные программы для создания архивов, т. е. файлов, имеющих повышенную плотность записи информации.
- ⇓ **Беспроводные локальные сети** – сети для передачи и приема данных с помощью радиоволн. Область действия сети может ограничиваться лишь одной комнатой, но может охватывать и большее пространство.
- ⇓ **Беспроводное подключение** – очень редко используется в стационарных компьютерах. Идеальное решение для ноутбуков, «умных» телефонов и других подобных устройств.
- ⇓ **Бит** – количество информации, содержащееся в сообщении, уменьшающем неопределенность знаний в два раза.



- ⇒ **Видеоинформация** – графическое изображение (кадр) или последовательность кадров, сменяющих друг друга.
- ⇒ **Видеопамять (VRAM)** – это разновидность оперативного запоминающего устройства, в котором хранятся закодированные изображения.
- ⇒ **Вложенные циклы** – циклическая конструкция, где неоднократно повторяется та часть программы, в которой уже есть повторение.
- ⇒ **Вредоносная программа** – любое программное обеспечение, предназначенное для получения несанкционированного доступа к вычислительным ресурсам компьютера или к информации, хранимой на компьютере, с целью несанкционированного использования ресурсов компьютера или причинения вреда владельцу информации путем копирования, искажения, удаления или подмены информации.
- ⇒ **Графический пользовательский интерфейс** – ориентирован для поддержки интерактивного взаимодействия с программным обеспечением, осуществляя обратную связь с пользователем.
- ⇒ **Данные** – информация, используемая в программах в процессе решения поставленных задач.
- ⇒ **Диаграмма** – визуальное представление числовых значений.
- ⇒ **Идентификатор** – последовательность из букв латинского алфавита, цифр и символов подчеркивания, начинается с буквы или символа подчеркивания.
- ⇒ **Интегрированная среда разработки (ИСР)** – система программных средств, используемая программистами для разработки программного обеспечения.
- ⇒ **Интерпретатор (Interpreter)** – переводит и выполняет программу строка за строкой тут же, не порождает на выходе программу на машинном языке.
- ⇒ **Интерфейс** – совокупность возможностей, способов и методов одновременного воздействия, в том числе посредством обмена информацией между объектами взаимодействия. Интерфейс бывает аппаратный, программный, пользовательский.
- ⇒ **Информация** – знания, сведения или команды, которые мы получаем из окружающего мира.
- ⇒ **Информационная модель** – описание моделируемого объекта на одном из языков кодирования информации. Прототип – объект, процесс или явление, на основе которого создается модель.



- ⇓ **Канал связи** – техническое средство, позволяющее осуществлять передачу данных на расстоянии. Компьютеры могут обмениваться информацией с использованием каналов связи различной физической природы: кабельных, оптоволоконных, радиоканалов и др.
- ⇓ **Код** – система условных знаков для представления информации.
- ⇓ **Компилятор (Compiler)** – читает всю программу целиком, делает ее перевод и создает законченный вариант программы на машинном языке, который затем выполняется.
- ⇓ **Компьютерная безопасность** – меры безопасности, применяемые для защиты вычислительных устройств (компьютеры, смартфоны и др.), а также компьютерных сетей (частных и публичных сетей, включая Интернет).
- ⇓ **Компьютерная память** – устройство хранения информации или запоминающее устройство. Память является обязательной частью компьютера.
- ⇓ **Компьютерное моделирование** – особый вид информационного моделирования, при котором используется вычислительная техника.
- ⇓ **Компьютерные сети** – объединение компьютеров для обмена информацией и совместного использования ресурсов.
- ⇓ **Компьютер-сервер** – центральный компьютер, распределяющий общие ресурсы между многими пользователями.
- ⇓ **Константа** – зарезервированная, определенная величина, используемая в процессе выполнения программ.
- ⇓ **Кэш-память, или сверхоперативная память** – очень быстрое запоминающее устройство небольшого объема. Используется при обмене данными между микропроцессором и оперативной памятью.
- ⇓ **Маркер заполнения** – небольшой черный квадрат в правом нижнем углу выделенной ячейки или выделенного диапазона.
- ⇓ **Моделирование** – процесс построения моделей для исследования и изучения объектов, процессов и явлений.
- ⇓ **Модель** – некий новый упрощенный объект, который отражает существенные особенности реального объекта, процесса или явления.
- ⇓ **Объект** – некоторая часть окружающего мира.
- ⇓ **Оперативная память (Random Access Memory – RAM)** – быстрое запоминающее устройство небольшого объема, связанное с процессором и предназначенное для записи, считывания и хранения выполняемых программ и данных, обрабатываемых этими программами.



- ⇒ **Переменная** – ячейка в памяти компьютера, которая имеет имя и хранит некоторое значение. Значение переменной может меняться во время выполнения программы. При записи в ячейку нового значения старое стирается.
- ⇒ **Пользовательский интерфейс** – взаимодействие пользователя с компьютером.
- ⇒ **Последовательности** – введенные в соседние ячейки данные, представляющие собой некоторую закономерную последовательность.
- ⇒ **Постоянная память (ПЗУ, англ. ROM, Read Only Memory – память только для чтения)** – энергонезависимая память, используется для хранения данных, которые никогда не потребуют изменения.
- ⇒ **Прикладное программное обеспечение** – совокупность программ различного назначения для автоматизации обработки различного рода информации, выполнения расчетов в разных сферах деятельности человека, обучения и развлечения.
- ⇒ **Программирование** – процесс создания программ для компьютера.
- ⇒ **Программное обеспечение (ПО или Software)** – совокупность всех программ, обеспечивающих функционирование аппаратных средств, разработку, отладку и выполнение задач пользователя.
- ⇒ **Программный интерфейс** – взаимодействие между различными программами компьютера.
- ⇒ **Распаковка архива** – процесс восстановления сжатых файлов.
- ⇒ **Сервер** – абонент сети, отдающий в сеть свой ресурс и имеющий или не имеющий доступ к ресурсам сети. Используется для хранения файлов, веб-страниц, для управления почтой, для хранения копий файлов.
- ⇒ **Сжатие данных** – используется при работе с файлами больших форматов.
- ⇒ **Синтаксис** – набор правил, определяющих возможные сочетания (конструкции) из букв алфавита.
- ⇒ **Система программирования** – комплекс программ, предназначенный для автоматизации программирования задач на ЭВМ (электронно-вычислительной машине).
- ⇒ **Составное условие** – условие, состоящее из нескольких простых условий, связанных с помощью логических операций.
- ⇒ **Спутниковое подключение** – имеет самый широкий радиус подключения, но используется только в стационарных компьютерах.
- ⇒ **Таблица** – объект, который добавляется в документ для организации текстов и других элементов таблицы.



- ⇓ **Текстовый процессор** – класс программ, позволяющий вводить, корректировать, форматировать простые и комплексные текстовые документы, а также управлять выводом на бумагу и экран.
- ⇓ **Текстовый файл** – файл, компонентами которого являются данные символьного типа.
- ⇓ **Тип данных** – характеристика группы данных, определяющая способ хранения данных в памяти компьютера, множество допустимых значений и допустимые операции, применимых к данному типу.
- ⇓ **Топология** – характеризует физическое расположение компьютеров, кабелей и других компонентов сети.
- ⇓ **Транслятор (Translator)** – программа-переводчик. Преобразует программу, написанную на одном из языков высокого уровня, в программу, состоящую из машинных команд.
- ⇓ **Условное форматирование** – один из самых полезных инструментов в электронной таблице, позволяет выполнять форматирование ячеек выборочно и автоматически, основываясь на их содержанием.
- ⇓ **Форматирование** – изменение свойств текста, определяющих его внешний вид, без изменения содержимого.
- ⇓ **Формула** – специальный инструмент в электронной таблице Excel, предназначенный для расчетов, вычислений и анализа данных.
- ⇓ **Электронная таблица Excel** – приложение, обрабатывающее информацию, которую можно представить в виде таблиц и автоматизировать обработку таких данных.
- ⇓ **Язык программирования** – формальная знаковая система, предназначенная для записи компьютерных программ. Язык программирования определяет набор правил, определяющих внешний вид программы и действия, которые выполнит исполнитель (обычно компьютер) под ее управлением.
- ⇓ **Ячейка** – основная единица хранения данных.



Учебное издание

**Мухамбетжанова Сауле Талапеденовна
Тен Алира Сунтаковна
Исабаева Даража
Сербин Василий Валерьевич**

ИНФОРМАТИКА

Учебник для 7 классов общеобразовательных школ

Зав. редакцией *Н. Жиенгалиев*
Редактор *А. Раймкулова*
Художественный редактор *З. Аульбекова*
Технический редактор *О. Рысалиева*
Корректор *И. Кротов*
Компьютерная верстка *Э. Омаровой*

ИБ № 049

Сдано в набор 11.01.2017. Подписано в печать 09.06.2017. Формат 70x90^{1/16}.
Бумага офсетная. Гарнитура «Школьная». Печать офсетная.
Усл. печ. л. 12,87. Уч.-изд. л. 8,57. Тираж 50 000 экз. Заказ №2499.
ТОО «Корпорация «Атамұра», 050000, г. Алматы, пр. Абылай хана, 75.
Полиграфкомбинат ТОО «Корпорация «Атамұра», Республика Казахстан.
050002, г. Алматы, ул. М. Макатаева, 41.