



Л.А. Габдуллина

Informatica

ПРАКТИКУМ ПО ИНФОРМАТИКЕ



Ижевск 2014

Министерство образования и науки Российской Федерации

ФГБОУ ВПО «Удмуртский государственный университет»

Географический факультет

Кафедра экологии и природопользования

ПРАКТИКУМ ПО ИНФОРМАТИКЕ

Учебно-методическое пособие



Ижевск 2014

УДК 076.5
ББК 32.81
М74

Рекомендовано к изданию учебно-методическим советом УдГУ

Рецензент: к. мат. н. Т.С. Быкова

Составитель: к. г. н. Л.А. Габдуллина

М74 Практикум по информатике: Учеб.-метод. пособие / сост. Л.А. Габдуллина. Ижевск: Изд-во «Удмуртский университет», 2014. 66 с.

Пособие представляет собой краткие сведения и сборник заданий по вузовскому курсу «Информатики» для студентов бакалавриата географического факультета направления «Экология и природопользования». В учебно-методическом пособии кратко изложены основы работы в текстовых редакторах Microsoft Word (пакет Microsoft Office), Writer (пакет OpenOffice.org); работа с электронными таблицами в программах MS Excel и OO Calc; работа с мастером диаграмм в MS Excel и OO Calc; программирование на языке Си; основы работы в Интернете.

Пособие адресовано преподавателям кафедр вузов, которые могут использовать приведенные задания для рубежного и итогового контроля знаний студентов. Пособие может быть использовано и студентами для самоконтроля. Приведенный материал может быть полезен и преподавателям техникумов, профессиональных училищ и общеобразовательных школ.

УДК 076.5
ББК 32.81

© Л.А. Габдуллина, 2014
© ФГБОУ ВПО «Удмуртский государственный университет», 2014

ВВЕДЕНИЕ

Ключевой проблемой образовательной политики России является проблема повышения качества образования. На решение этой проблемы нацелено создание Национальной системы оценки качества образования, ориентированной как на федеральные государственные образовательные стандарты (ФГОС), так и на признанные международные нормы по качеству образования. Важнейшим этапом этой системы должен стать мониторинг качества образования, как механизм единой и регулярной оценки соответствия качества подготовки студентов требованиям ФГОС по каждому направлению подготовки. Реализация этого механизма требует создания набора тренировочных заданий и фонда педагогических контрольных материалов для диагностики результатов образования.

При выполнении работ по информатике доля самостоятельной деятельности студентов должна быть существенно выше, чем при других видах учебной работы; преподаватель в этой ситуации довольно часто выступает в роли консультанта. Это помогает будущему специалисту научиться самостоятельно осваивать новые знания и навыки, что является одной из важнейших целей обучения.

Кроме того, любой современный человек должен иметь элементарные навыки работы с компьютером. К ним относятся: умение запустить программу, скопировать файл и т.д., работая в среде какой-либо операционной системы или оболочки. Выпускник вуза высшей квалификации должен уметь набирать, редактировать и выводить на печать текст документа в одном из текстовых редакторов. Он должен освоить искусство программирования и работу с пакетами программ, используемых в той или иной области деятельности. Кроме того образованный человек должен уметь уверенно пользоваться основными информационными услугами, предоставляемыми, такими сетями, как Интернет.

По соответствующим темам пособие включает контрольные материалы, тестовые задания, задачи, упражнения, которые располагаются по мере возрастания трудности. Задания адаптированы к применению автоматизированного контроля, кафедра располагает персональными компьютерами, что сегодня стало общераспространенно. Все это позволяет оперативно, объективно и экономично измерять обученность студентов.

Изложение материала в пособии основано на операционной системе *Windows* и офисных пакетах *MS Office* и *OpenOffice.org*. Высокая стоимость пакета *Microsoft Office* и отсутствие значимых льгот у учебных заведений являются препятствием для его

использования в учебном процессе. В связи с этим многие учебные заведения (и не только они) переходят на альтернативные открытые офисные пакеты. В расчете на эту категорию читателей в пособие включено описание открытого офисного пакета *OpenOffice.org*. Бесплатный дистрибутив пакета можно найти по адресу <http://www.openoffice.org>.

1. РАБОТА С ТЕКСТОМ

Работа с текстовыми документами включает в себя их создание, редактирование, оформление по специальным правилам, включение в текст различных объектов: таблиц, диаграмм, формул, чертежей и т.д., а также сохранение на электронных носителях, пересылку по почте и печать.

1.1. Программы для работы с текстом

Программные средства для работы с текстовыми документами включают в себя:

- просмотрщики (вьюеры), предназначенные только для просмотра ранее созданных текстовых документов;
- текстовые редакторы, обеспечивающие создание, редактирование, просмотр, сохранение на электронных носителях и печать текстовых документов различной сложности;
- издательские системы, с помощью которых газеты, журналы и книги готовятся к массовому тиражированию в издательствах.

К группе просмотрщиков относится программа *Adobe Acrobat Reader* и др.

Одним из самых мощных текстовых редакторов является входящий в состав пакета *Microsoft Office* редактор *Microsoft Word*. Ему под силу создание документов практически любой степени сложности. С редактором *Microsoft Word* сопоставима по своим возможностям программа *WordPerfect* из офисного пакета *Corel WordPerfect Office*.

К этому же классу профессиональных инструментов подготовки текстов можно отнести бесплатный и достаточно мощный редактор *Writer*, входящий в разработанный фирмой *Apache Software Foundation* (ранее *Sun Microsystems*) пакет *OpenOffice.org*.

Все упомянутые текстовые редакторы работают по принципу «что вы видите, то вы и получаете» (принцип WYSIWYG), то есть пользователь видит документ в том виде и в том оформлении, в котором он будет напечатан.

Издательские системы, служащие для подготовки к тиражированию в типографских условиях газет, журналов, книг и др. отличаются от текстовых редакторов возможностями формировать

текст до измеряемых долями миллиметров деталей его оформления. Популярными системами издательских систем *Ventura Publisher*, *Adobe PageMaker*, *Adobe InDesign* и *QuarkXPress*.

1.2. Основные возможности текстовых редакторов

Практически любой текстовый редактор обладает стандартными средствами: ввод текста, внесение изменений в текст (на различных языках – русском, немецком, английском и т.д.); оформление текста по определенным правилам; сохранение документов в виде файлов на внешних устройствах; печать подготовленных документов.

В редакторы *Word* и *Writer* включены широкие возможности по оформлению текста документа, которое принято называть форматированием.

Форматированием называется оформление всего текста или какого-либо его участка по определенным правилам. Форматировать можно символы, слова, абзацы, страницы, главы и т.д. до всего текста включительно. Форматирование включает в себя выбор начертаний и размеров шрифтов, оформление абзацев, установку отступов, межстрочных расстояний, полей страниц, колонтитулов и многое другое.

Форматирование осуществляется с помощью параметров форматирования, таких как: размер листа бумаги, размер полей, наличие и способ нумерации, положение номера, количество колонок текста на странице и некоторые другие. Конкретный набор параметров форматирования объекта называется стилем.

Разделом принято называть часть документа с одними и теми же значениями параметров форматирования страниц. Изменение хотя бы одного из параметров приводит к возникновению нового раздела документа.

К основным возможностям текстовых редакторов *Microsoft Word* и *OpenOffice.org Writer* относятся:

- наличие встроенной мощной справочной системы;
- ввод и редактирование текста на различных языках (более 100 языков);
- изменение форм текстового курсора и указателя мыши при изменении текущей ситуации в редакторе;

-богатый набор шаблонов документов, наличие мастеров – подсистем, автоматизирующих работу над документами в стандартных ситуациях;

-ввод текста с автоматическим форматированием строк, абзацев и страниц;

-работа с документами практически произвольных размеров, ограниченных только объемом свободной дисковой памяти;

-возможность импорта, т.е. преобразование файлов из форматов других редакторов в собственный формат;

-возможность экспорта, т.е. преобразование файлов из собственного формата в форматы других редакторов;

-наличие различных режимов работы с документом;

-большой выбор шрифтов, размеров символов;

-большой выбор возможностей форматирования абзацев и страниц документа;

-поиск и замена подстрок;

-доступ к буферу обмена;

-механизмы отмены и восстановления после отмены последних выполненных действий (откат и накат);

-возможности включения в документ специальных символов, рисунков, графиков, диаграмм и т.д.;

-средства автоматизации работы с документами – автотекст, автозамена, автоперенос, автоформат и т.д.;

-возможности создания фона, подчеркивания, обрамления;

-автоматизация создания таблиц;

-возможности создания текстовых эффектов – наклон, поворот, переворот и т.д.;

-возможности подготовки рисунков средствами самого редактора;

-средства ввода математических формул произвольной степени сложности;

-работа с многоколоночными документами (газетными статьями);

-проверка орфографии по встроенным словарям на разных языках;

-проверка синтаксиса;

-подбор синонимов;

-подготовка конвертов, наклеек, визиток и т.д.;

-одновременная работа с несколькими документами;

-широкие возможности управления печатью документов;

-рассылка документов по локальной сети;

-работа с электронной почтой и *Web* в Интернете;
-создание макросов – средств, обеспечивающих автоматизацию часто выполняемых однотипных операций.

В версии редактора *Word2002* впервые появились следующие возможности:

-специальная область задач, облегчающая выполнение операций создания документа, форматирования, поиска, работу с буфером и т.д.;

-многоэлементный буфер обмена (до 24 элементов);

-распознавание речи и рукописных текстов;

-автоматическое восстановление документа после сбоя.

Файлы, содержащие подготовленные в редакторе *Word* документы, сохраняются на дисковых носителях со стандартным расширением *.doc*, а файлам, которые созданы редактором *Writer*, автоматически приписывается стандартное расширение *.odt* (от *Open Document Text*). Такие файлы записываются в специальных формах, которые не воспринимаются, например, в текстовых редакторах *Блокнот*, *WordPad* и их аналогах.

1.3. Создание таблиц

Существенно упрощается обзор и анализ больших массивов числовой или текстовой информации с некоторой повторяющейся структурой, если использовать табличную форму представления.

Ячейки таблиц образуются на пересечениях ее строк и столбцов. Обычно ячейки таблицы отделены друг от друга горизонтальными и вертикальными линиями. Эти линии могут быть различной толщины, цвета и стиля. Столбцы таблицы принято обозначать буквами латинского алфавита А, В, С, D, ..., а строки – последовательными числами 1, 2, 3, Каждая ячейка обозначается буквой и числом, соответствующей столбцу и строке.

Ячейки таблиц можно заполнять текстами, числами, различными графическими объектами (чертежи, рисунки, фотографии и др.).

В редакторе предусмотрены два основных способа создания таблиц:

-формирование пустой таблицы с последующим заполнением ее ячеек;

-преобразование в таблицу ранее введенных фрагментов документа – при этом сразу получается заполненная таблица.

Создавать пустую таблицу, как и выполнять преобразование существующих фрагментов в таблицу можно разными способами.

В редакторе *Word* предусмотрена еще и возможность рисования пустой таблицы "от руки" и последующее заполнение ее ячеек.

Создание пустой таблицы начинается с выбора места ее размещения в документе и перемещения текстового курсора в выбранную позицию. Проще всего создавать таблицы с автоматически определяемой шириной столбцов. Пользователь указывает только количество строк и столбцов, из которых должна состоять таблица. Остальные операции выполняются автоматически. Щелчок по кнопке **Вставить таблицу** в панели **Стандартная** приводит к развертыванию панели **выбора размерности** таблицы. Для определения размерности (то есть количества строк и столбцов) указатель мыши следует совместить с левым верхним квадратом и перемещать по строкам и столбцам панели направо и вниз. В момент достижения планируемой размерности следует выполнить щелчок. В том месте документа, в котором находится текстовый курсор, отобразится созданная таблица.

Аналогичный по простоте и результатам способ основан на команде **Таблица> Вставить> Таблица**, с помощью которой открывается окно **Вставка таблицы**. Количество строк и столбцов таблицы указывается в настраиваемых полях **Число столбцов (Столбцы)** и **Число строк (Строки)** окна, и выполняется щелчок по кнопке **ОК**.

Преобразуемый в таблицу текст разбивается пользователем на участки, каждый из которых должен занимать в создаваемой таблице отдельную ячейку. Деление фрагмента документа на такие участки происходит по символам разделителям, в качестве которых могут использоваться признаки конца абзаца, точки с запятой, позиции табуляции или любые другие, назначенные редактором символы.

Чтобы преобразовать текст в таблицу нужно выделить фрагмент документа, в котором уже присутствуют символы-разделители, а затем щелкнуть по кнопке **Добавить таблицу (Таблица)** или выбрать команду **Таблица> Преобразовать> Текст в таблицу** из меню редактора. Щелчок по кнопке приводит к преобразованию текста в таблицу с параметрами, установленными пользователем по умолчанию. Во втором случае открывается окно **Преобразовать в таблицу (Преобразовать текст в таблицу)**, в котором можно задать параметры формируемой таблицы.

Диалоговое окно преобразования похоже на окно **Вставка таблицы**. Оно имеет схожий набор элементов управления, причем они могут быть использованы точно так же, как и в окне **Вставка таблицы**. Вместе с тем, некоторые элементы управления заблокированы, кроме того, в окно добавлена группа переключателей **Разделитель (Разделитель текста)**, с помощью которых можно указать, по какому из одновременно имеющихся во фрагменте документа символов-разделителей следует выделять преобразуемые в ячейки таблицы части текста.

Для выбора другого переключателя, пользователь может указать в расположенном справа от переключателя поле любой символ, который и будет фактически использован как символ-разделитель.

В редакторе *Word* еще одним важным для формирования таблицы параметром является количество столбцов, которое указывается в настраиваемом поле **Число столбцов**. Поле ввода количества строк таблицы заблокировано, оно зависит от количества имеющихся в выделенном тексте символов-разделителей и указанного пользователем количества столбцов. После указания количества столбцов в заблокированном поле отображается подсчитанное редактором количество строк.

В редакторе *Writer* при преобразовании текста в таблицу ширина столбцов выбирается пропорциональной содержимому ячеек. Включение флажка **Одинаковая ширина всех столбцов** приводит к формированию таблицы, в которой все столбцы имеют одинаковую, независимую от содержания ячеек ширину.

В редакторе *Word* таблицу можно нарисовать так, как это делается с помощью линейки и карандаша на бумажном листе. Эти возможности доступны в режиме **рисования таблицы**, переключение в который происходит с помощью щелчка по кнопке **Таблицы и границы** в панели **Стандартная** либо с помощью команды **Таблица > Нарисовать таблицу** меню редактора. В режиме рисования таблицы указатель мыши приобретает форму карандаша, открывается окно, содержащее панель **Таблицы и границы**, и происходит автоматическое переключение в режим разметки документа. С помощью развертываемых списков **Тип линии**, **Толщина линии**, **Цвет линии** панели **Таблицы и границы** устанавливаются характеристики линий, которыми должна быть нарисована таблица. Далее указатель мыши располагается в той точке страницы, где должен находиться левый верхний угол таблицы. Затем нажимается левая клавиша мыши и выполняется перетаскивание контура будущей

таблицы в направлении ее правого нижнего угла. После достижения требуемых геометрических размеров клавиша мыши отпускается. Получаем внешний контур таблицы.

Для проведения вертикальных, горизонтальных или диагональных разделительных линий указатель мыши нужно совместить с одной из внешних сторон таблицы или уже с существующей внутренней линией и выполнить перетаскивание следа до пересечения с противоположной стороной или линией.

В момент начала ввода текста происходит выход из режима рисования, о чем свидетельствует изменение формы курсора мыши. Выйти из режима рисования можно также с помощью клавиши Esc. Для возвращения в режим рисования следует щелкнуть по кнопке **Нарисовать таблицу** в панели **Таблицы и границы**.

1.4. Упражнения

Задание 1. Знакомство с основными настройками документа и технологией его сохранения в среде текстового процессора *Word*.

1. Запустите текстовый процессор *Word*.
2. Задайте основные параметры (тип и размеры шрифта, способ выравнивания текста), которые будут устанавливаться по умолчанию при вводе текста.
3. Наберите текст, отображенный на рисунке 1.
4. Сохраните набранный текст в своем файле.
5. Закройте документ.
6. Откройте документ.
7. Закройте документ.
8. Научитесь выходить из среды *Word*.

<p>Уважаемые господа! Приглашаем вас на юбилейную презентацию компьютерной фирмы «МАКУЛШИРС». Мы работаем на российском рынке много лет. Информационные продукты нашей фирмы знают и любят многие пользователи страны. Наши компьютеры работают без рекламаций! Будем рады видеть вас. Запомните адрес и время нашей презентации – Компьютерная улица, д.5, 18-00. Справки по телефону 123-45-67.</p>

Рис.1. Текст задания 1.

Задание 2. Выполните основные операции по редактированию текстовых документов *Word*, воспользовавшись ранее созданным текстом.

1. Разбейте текст на абзацы и сравните с изображением на рис.2.
2. Выделите различные фрагменты текста.
3. Скопируйте абзац.
4. Удалите абзац.
5. Прделайте ряд упражнений по перестановке абзацев, предложений в абзацах, слов в предложениях.
6. Скопируйте весь текст задания и исправьте в нем обращение к одному человеку – «Дорогой друг!».
7. Замените одно слово на другое, например «вас» на «Вас».
8. Сохраните файл.

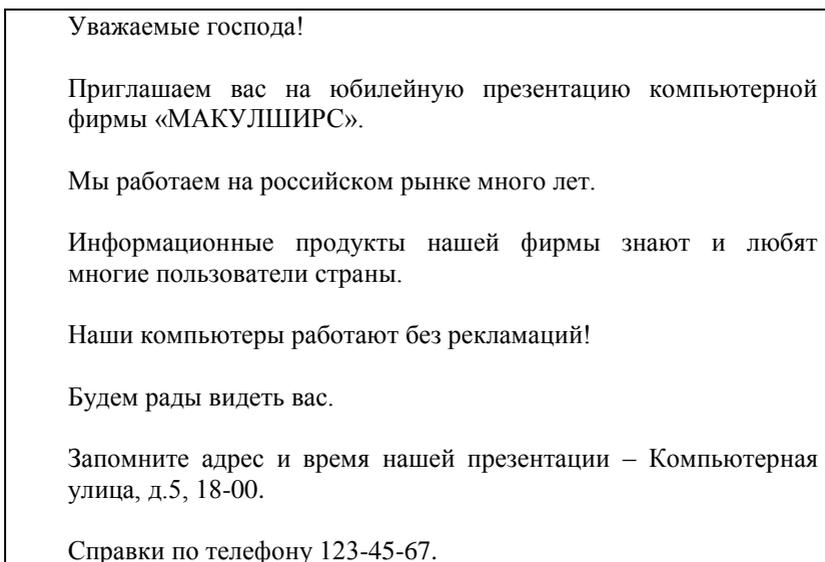


Рис.2. Текст задания 2.

Задание 3. Наберите текст и оформите его фрагменты как нумерованные и маркированные списки в соответствии с рис. 3.

Запомните!

1. Прежде чем начать форматирование фрагмента текста, надо его выделить.
2. Форматировать можно с помощью меню или панели инструментов Форматирование.

Форматировать текст – это значит уметь выполнять следующие операции:

- устанавливать шрифт, т.е. задавать параметры:
 - тип,
 - начертание,
 - размер,
 - подчеркивание,
 - цвет;
- определять эффекты в шрифтах:
 - верхний индекс,
 - нижний индекс,
 - зачеркнутый,
 - утопленный,
 - приподнятый и пр.;
- устанавливать межсимвольные интервалы;
- устанавливать межстрочные интервалы;
- делать первую строку абзаца красной или висячей;
- и еще множество других операций,

Рис. 3. Текст задания 3.

Задание 4. Создайте таблицу, соблюдая все детали оформления.

Группа 1921. Итоги зимней и летней сессий.

№	Фамилия И.О.	Зимняя сессия 2012				Летняя сессия 2012			
		Аппар. ср-ва	Мат. логика	История	Мат. анализ	Языки прог-я	Алгебра	Криптография	Мат. анализ
1	Лазарев А.В.	4	4	5	4	5	5	4	4
2	Синявский И.П.	5	4	4	4	5	4	5	4
3	Афанасьев А.Е.	3	4	3	4	3	4	3	3
4	Илюшкин Д.И.	4	5	5	3	4	5	5	5
5	Гончаров Е.С.	3	3	3	4	4	3	4	3
6	Трофимова Т.А.	5	5	5	5	4	4	5	5

Задание 5. Создайте таблицу, соблюдая все детали оформления.

	a	b	c	d	e	f	g	h	
8		■		■		■		■	8
7	■		■		■		■		7
6		■		■		■		■	6
5	■		■		■		■		5
4		■		■		■		■	4
3	■		■		■		■		3
2		■		■		■		■	2
1	■		■		■		■		1
	a	b	c	d	e	f	g	h	

Задание 6. Наберите текст и оформите его фрагменты как нумерованные и маркированные списки в соответствии с нижеприведенной таблицей.

Маркированный список	Нумерованный список	Многоуровневый список
<ul style="list-style-type: none"> ▪ Аппаратное обеспечение: <ul style="list-style-type: none"> • Системный блок • Монитор • Клавиатура • Принтер ▪ Программное обеспечение: <ul style="list-style-type: none"> • Системное • Прикладное ▪ Инструментарий программирования 	<p>I. Аппаратное обеспечение:</p> <p>Системный блок Монитор Клавиатура Принтер</p> <p>II. Программное обеспечение:</p> <p>Системное Прикладное</p> <p>III. Инструментарий программирования</p>	<p>1. Аппаратное обеспечение:</p> <p>1.1. Системный блок 1.2. Монитор 1.3. Клавиатура 1.4. Принтер</p> <p>2. Программное обеспечение:</p> <p>2.1. Системное 2.2. Прикладное</p> <p>3. Инструментарий программирования</p>

Задание 7. Наберите текст (свойства степеней), соблюдая все детали оформления.

$$a^n \cdot a^m = a^{n+m}$$

$$a^n : a^m = a^{n-m}$$

$$(a^n)^m = a^{n \cdot m}$$

$$a^0 = 1$$

$$(a/b)^k = a^k / b^k$$

Задание 8.

А. Подготовьте таблицу подсчета количества студентов, сдавших экзамены на «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Сведения об успеваемости студентов							
	Учебная дисциплина	Группа	Всего сдавало	Отлично	Хорошо	Удовл.	Неудовл.
	Информатика						
1.		13		13	14	3	2
2.		14		7	7	5	5
3.		15		9	8	9	8
4.		16		8	8	8	3
5.		17					
	ИТОГО						
	Философия						
1.		13		10	5	15	2
2.		14		3	18	2	1
3.		15		5	5	10	3
4.		16		7	7	7	7
5.		17		8	8	8	8
	ИТОГО						
	Высшая математика						
1.		13		7	11	15	1
2.		14		16	9	6	2
3.		15		8	8	5	5
4.		16		3	8	8	3
5.		17		5	5	9	9
	ИТОГО						

Б. Произведите расчеты:

- числа студентов каждой учебной группе, сдававших экзамен по определенной дисциплине;
- числа студентов, сдавших экзамен по каждой дисциплине на «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».
- числа студентов, сдававших экзамен по определенной дисциплине.

Задание 9. Научитесь рисовать схемы, используя стандартные графические объекты панели.

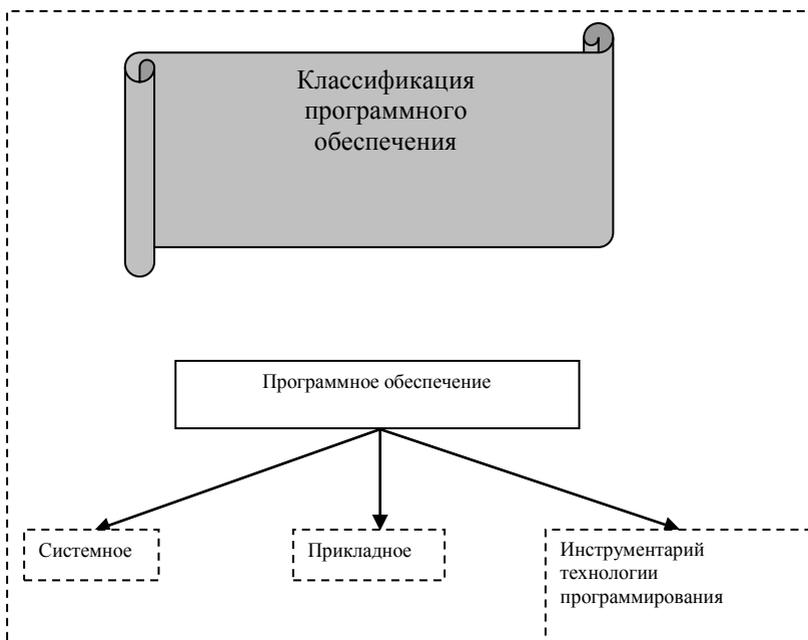


Рис. 4. Текст задания 9.

Задание 10. Создайте текст, соблюдая все детали оформления.

Цветное фото

(материалы KODAK)

Только здесь бесплатно при заказе:

100 руб.- проявка и фотоальбом

200 руб.- проявка, фотоальбом и пленка

Мы ждем Вас по адресам:

- Кафе «Адам», проспект Мира, 53
- Гостиница «Красноярск»
- Фотосалон, ул.Весны, 5
- Магазин «Веселые ребята»
- Магазин «Первый тайм»
- Магазин «Локомотив»
- Магазин «Бирюсинка»
- Магазин «Воскресенский»
- Магазин «Сосновоборск»
- Магазин «KODAK»
- Торговый центр
- ЦУМ
- «Детский мир».

Рис. 5. Текст задания 10.

Задание 11. Создайте текст, соблюдая все детали оформления.

Предлагаю услуги репетитора по *английскому языку*

Контактный телефон: 12-34-56.

12-34-56

12-34-56

12-34-56

12-34-56

12-34-56

12-34-56

12-34-56

12-34-56

12-34-56

12-34-56

12-34-56

Задание 12. Создайте текст, соблюдая все детали оформления.

Мой e-mail: ab@instic.ru.

Моя зарплата: xxx \$.

По истории задали §100.

По физике мы проходили α , β , γ частицы.

Простейшим типом периодических колебаний являются гармонические колебания:

$$x=A \sin (\omega t+\alpha_0)$$

или

$$x=A \cos (\omega t+\alpha_1),$$

где A , ω , α_0 , α_1 – постоянные величины, причем $A>0$, $\omega>0$ и $\alpha_1 = \alpha_0 - \pi/2$.

Задание 13. Освойте технологию создания документов, в которых используются графические объекты:

- стандартные рисунки;
- графические объекты из коллекции WordArt;
- графические объекты из коллекции Автофигуры.

Уважаемые господа!

Приглашаем вас на юбилейную презентацию компьютерной фирмы

«МАКУЛШИРС».

Справки по телефону 123-45-67.

Рады видеть Вас по адресам:

- г. Ижевск, ул. Мира, д.5,
- г. Сарапул, ул. Гоголя, д.7,
- г. Можга, ул. Ленина, д.9.

Рис. 6. Текст задания 13.

1.5. Тестовые задания

1. Текстовый редактор *Word* – это:
 - 1) прикладная программа;
 - 2) базовое программное обеспечение;
 - 3) сервисная программа;
 - 4) редактор шрифтов.
2. Под термином «кегель» понимают:
 - 1) размер полосы набора;
 - 2) размер шрифта;
 - 3) расстояние между строками;
 - 4) начертание шрифта.
3. В текстовых редакторах и настольных издательских системах, как правило, с помощью клавиш Alt+F4 происходит:
 - 1) переход в окно с предыдущей программой;
 - 2) переход в окно со следующей программой;
 - 3) открытие файла;
 - 4) выход из программы.
4. В текстовых редакторах и настольных издательских системах, как правило, с помощью клавиш Ctrl+S происходит:
 - 1) создание нового документа;
 - 2) открытие файла;
 - 3) сохранение файла;
 - 4) печать файла.
5. Следующая последовательность действий:
установить указатель мыши на полосе выделения рядом с текстом; нажать левую клавишу мыши и, удерживая ее, передвигать мышь в нужном направлении
в *Word* приведет:
 - 1) к выделению текста;
 - 2) к удалению текста;
 - 3) к перемещению текста;
 - 4) к копированию текста в буфер.
6. Следующая последовательность действий:
выделить нужный участок текста; нажать на нем левую клавишу мыши и, удерживая ее, передвигать мышь до нужного места
в *Word* приведет:
 - 1) к копированию выделенного участка текста;
 - 2) к переносу выделенного участка текста;
 - 3) к замене текущего текста на выделенный;
 - 4) к удалению выделенного участка текста в буфер.

7. Абзацные отступы и ширина колонок могут изменяться в *Word* с помощью:

- 1) линейки прокрутки;
- 2) координатной линейки;
- 3) строки состояния;
- 4) поля пиктограмм.

8. При нажатии на кнопку с изображением изогнутой влево стрелки на панели пиктографического меню в *Word*:

- 1) появляется диалоговое окно для добавления гиперссылки
- 2) отменяется последняя команда
- 3) происходит разрыв страницы
- 4) повторяется последняя команда.

9. При нажатии на кнопку с изображением дискеты на панели пиктографического меню в *Word* происходит:

- 1) считывание информации с дискеты;
- 2) запись документа на дискету;
- 3) сохранение документа;
- 4) печать документа.

10. При нажатии на кнопку с изображением ножниц на панели пиктографического меню в *Word*:

- 1) происходит разрыв страницы;
- 2) вставляется вырезанный ранее текст;
- 3) удаляется выделенный текст;
- 4) появляется схема документа, разбитого на страницы.

11. Какую комбинацию «горячих клавиш» нужно нажать в *Word*, чтобы вставить скопированный блок текста без использования пиктограмм:

- 1) Ctrl+C;
- 2) Shift+Enter;
- 3) Ctrl+E;
- 4) Ctrl+V?

12. Какую комбинацию «горячих клавиш» нужно нажать в *Word*, чтобы выделить весь файл без использования пиктограмм:

- 1) Ctrl+S;
- 2) Ctrl+B;
- 3) Shift+Insert;
- 4) Ctrl+A?

13. Какую комбинацию «горячих клавиш» нужно нажать в *Word*, чтобы вставить в текст гиперссылку без использования пиктограмм:

- 1) Alt+G;
- 2) Ctrl+K;

- 3) Shift+V;
 - 4) Ctrl+C?
14. В редакторе *Word* предусмотрены способы создания таблиц:
- 1) формирование пустой таблицы;
 - 2) преобразование в таблицу ранее введенных фрагментов документа;
 - 3) рисование пустой таблицы от руки и последующее заполнение ее ячеек;
 - 4) все перечисленное в пунктах 1-3.
15. Ячейки таблиц в *Word* и *Writer* можно заполнять:
- 1) только числами;
 - 2) только текстами;
 - 3) только числами и текстами;
 - 4) текстами, числами, формулами, различными графическими объектами (рисунки, чертежи, фотографии и т.д.).

1.6. Контрольные вопросы

1. Классифицируйте программные средства работы с текстом. Приведите примеры программ различных групп.
2. В чем заключается принцип WYSIWYG?
3. В чем заключается назначение и основные особенности текстового редактора?
4. Что понимается под форматированием документа?
5. Назовите основные атрибуты шрифта.
6. Назовите основные атрибуты абзаца.
7. Какую роль играют параметры форматирования? Приведите примеры таких параметров.
8. Что понимается под разделом документа в текстовом редакторе?
9. Охарактеризуйте основные возможности текстовых редакторов.
10. Как скопировать блок текста?
11. Какими особенностями отличаются файлы документов, создаваемые текстовыми редакторами *Word* и *Writer*?
12. Опишите порядок запуска и завершения работы текстового редактора.
13. В чем преимущества использования табличного способа представления информации?
14. Что может находиться в ячейках таблицы?

15. Охарактеризуйте основные действия с таблицами в редакторе.
16. Опишите способы создания таблиц в редакторе.
17. Опишите способ преобразования текста в таблицу.
18. Опишите способы перемещения по таблице.
19. Опишите способы выделения ячеек, строк и столбцов таблицы.
20. Опишите особенности и способы выполнения перемещения, копирования и удаления содержимого ячеек, строк или столбцов.
21. Как можно объединить ячейки в таблице?
22. Как можно удалить из таблицы отдельные ячейки, строки и столбцы?
23. Как удалить всю таблицу целиком?
24. Как автоматизировать нумерацию строк в таблице?
25. Как можно подсчитать сумму чисел, находящихся в ячейках таблицы?
26. Опишите способ рисования таблиц в редакторе *Word*.

2. РАБОТА С ЭЛЕКТРОННЫМИ ТАБЛИЦАМИ

Электронная таблица, так же как и обычная, состоит из строк и столбцов, на пересечении которых располагаются ячейки. Преимущество электронной таблицы от обычной – возможность размещения расчетных формул в ее ячейках, а также автоматизация обработки данных, представленных в табличном виде.

2.1. Общие сведения о табличных процессорах

Табличные процессоры – мощные и простые в использовании программы, предназначенные для решения широкого круга планово-экономических, научно-технических, учетно-статистических и других задач, в которых числовая, текстовая или графическая информация с некоторой повторяющейся, регулярной структурой представлена в табличном виде.

Эти программы предоставляют широкие возможности создания и изменения таблиц, которые могут содержать числа, тексты, даты, денежные единицы, графику, а также математические и иные формулы для выполнения вычислений. Ввод данных в таблицы, их форматирование и выполнение различного рода вычислений в значительной степени автоматизированы. Предусмотрены средства статистического анализа данных и решения оптимизационных задач. К основным возможностям и средствам табличных процессоров относятся:

- создание таблиц произвольной структуры и возможность их изменения;
- средства автоматизации заполнения ячеек таблиц данными;
- набор разнообразных функций для выполнения вычислений;
- средства автоматизации построения диаграмм;
- механизмы создания и обработки списков (баз данных): сортировка, фильтрация, поиск;
- средства автоматизации создания отчетов.

В стандартном случае запуск программ *MS Excel* и *OO Calc* выполняется с помощью команд основного меню **Все программы > Microsoft Excel** и **Все программы > Open Office.org > Open Office.org Calc** соответственно. Запуск возможен также с помощью значков, которые могут быть размещены на **Рабочем столе**, в

инструментальных панелях или в любой выбранной пользователем папке.

2.2. Интерфейс табличного процессора

Операционное меню табличного процессора имеет состав сходный с составом меню текстового редактора, только вместо пункта **Таблица** включен пункт **Данные** с командами, обеспечивающими сложные виды обработки данных. Набор инструментальных панелей табличного процессора значительно отличается от набора, предусмотренного в текстовом редакторе. Но при стандартной настройке в его окне присутствуют уже знакомые читателю панели **Стандартная** и **Форматирование**. Под ними расположена специализированная панель **Строка формул (Панель формул)**. Ее правая часть занята полем ввода с названием **строка формул (строка ввода)**, которое служит для отображения и редактирования содержимого ячеек таблицы. Строка состояния предназначена не только для отображения текущей справочной информации. В ряде случаев в ней отображаются результаты автоматически выполняемых вычислений. А контекстное меню строки может использоваться для управления этими вычислениями.

При стандартном запуске табличного процессора в рабочей зоне автоматически открывается окно документа с названием Книга 1 (Безымянный 1). В окне находится шаблон, т.е. заготовка документа, удовлетворяющая стандартным требованиям к его оформлению.

Документом табличного процессора является рабочая книга, состоящая из некоторого количества страниц, которые называются рабочими листами. Рабочий лист табличного процессора можно представлять как страницу, заранее подготовленную к созданию таблиц, т.е. разграфленную на строки и столбцы. Рабочий лист – это электронный аналог листа бумаги в клетку, где могут размещаться таблицы и текстовые пояснения к ним, а также различные графики, диаграммы, рисунки.

В нижней части каждого рабочего листа расположен специальный ярлычок с названием листа. По умолчанию рабочие листы имеют названия Лист1, Лист2, Лист3 и т.д., но их всегда можно изменить. Рабочий лист состоит из 65536 строк и 256 столбцов. Строки нумеруются последовательными натуральными числами (от 1 до 65536), а столбцы обозначаются буквами латинского алфавита и их парными комбинациями (A, B, C и т.д. до Z, AA, AB, AC и т.д. до AZ,

BA, ... BZ, ...). Номера или заголовки строк находятся в расположенном слева столбце заголовков строк, а обозначения или заголовки столбцов находятся в расположенной сверху строке заголовков столбцов. На пересечении строки и столбца находится основной структурный элемент таблицы – ячейка.

Адрес ячейки – определяется обычным координатным способом, например, ячейка B3, ячейка C7 и т.д. В каждую из ячеек можно занести число, формулу (арифметическое выражение) или текст. Операндами формулы могут быть математические функции, константы, адреса ячеек; в последнем случае реальный операнд – содержимое ячейки с указанным адресом.

Тексты и числовые константы, занесенные в электронную таблицу, сами по себе в ходе работы таблицы никогда не изменяются. Что же касается формул, то программа по специальной команде производит вычисление их значений и отображает его на экране (в той же ячейке, куда занесена формула).

Данные, входящие в таблицы, можно автоматически обрабатывать в графическом виде. Их также можно подготовить и распечатать в удобном для пользователя виде, сохранять и использовать многократно.

Задаваемый обычным образом адрес ячейки называется относительным адресом или относительной ссылкой.

При некоторых операциях копирования, удаления, вставки электронная таблица автоматически изменяет адреса ячеек. Иногда возникает необходимость не менять адрес ячейки. В таких случаях используют абсолютный адрес. Абсолютный адрес ячейки создается с помощью знака \$.

Группа ячеек (диапазон) задается через двоеточие, например, B3:D4 (или B3..D4), и образует прямоугольник, включающий ячейки B3, C3, D3, B4, C4, D4.

Над выделенной ячейкой можно совершать различные операции: выполнять ввод данных, редактирование, копирование, перемещение, форматирование, удалять данные из ячейки и т.д. При выполнении операций над содержимым ячейки в ней находится текстовый курсор, такой же, как в текстовом редакторе. Редактировать содержимое текущей ячейки можно не только в ней самой, но и в поле ввода панели **Строка формул (Панель формул)**.

2.3. Общие сведения о диаграммах в табличном процессоре

Числовые данные, находящиеся в ячейках таблицы, можно представлять в наглядной графической форме – в виде диаграммы, изображающей изменение числовых данных по строке или столбцу таблицы. Средства табличного процессора могут использоваться для быстрого и качественного создания и включения в рабочий лист диаграмм самых различных типов – линейных, столбчатых, круговых, гистограммных, объемных и т.д.

На одной диаграмме можно разместить несколько зависимостей, соответствующих нескольким строкам или столбцам таблицы. Такие зависимости принято называть рядами. Все диаграммы, за исключением круговой и объемной, имеют две оси: горизонтальную (ось категорий) и вертикальную (ось значений). Круговая диаграмма не имеет осей, у объемной диаграммы три оси. Дополнительную ось принято называть осью рядов. Включаемые в документ диаграммы по указанию пользователя снабжаются заголовком, разметкой и названиями осей, координатными сетками и легендой.

Создание любой диаграммы в рабочий лист требует от пользователя выполнения одной и той же последовательности действий, которые обеспечиваются подсказками **Мастера построения диаграмм**.

Рекомендуется следующий порядок построения диаграмм:

1) выделить строки или столбцы таблицы, по которым должна быть построена диаграмма, желательно включение в выделенный фрагмент строки с названиями категорий и столбца с названиями рядов или наоборот, строки с названиями рядов и столбца с названиями категорий;

2) щелкнуть по кнопке **Мастер диаграмм (Диаграмма)** панели инструментов **Стандартная** или выбрать из меню команду **Вставка > Диаграмма...**;

3) в диалоговых окнах мастера выбрать понравившийся образец диаграммы и с помощью находящихся на вкладках окон элементов управления оформить ее по своему усмотрению.

Для изменения первоначально выбранных геометрических размеров диаграммы нужно щелкнуть по любому ее участку – диаграмма окажется окруженной рамкой с черными (зелеными) квадратиками, расположенными по углам и серединам ее сторон. Эти квадратики являются маркерами изменения размеров. Выполняя

перетаскивание маркеров в каком-либо направлении, можно придать диаграмме нужные размеры. Чтобы выйти из режима изменения размеров, нужно щелкнуть по любой ячейке вне диаграммы.

2.4. Упражнения

Задание 1.

Табличный процессор Excel.

1) Дать определение:

- рабочая книга,
- рабочие листы.

2) Из чего состоит рабочее поле Excel?

3) Что такое:

- ячейка,
- адрес ячейки,
- активная ячейка.

4) Активизировать ячейку C3.

5) Освойте технологию переименования листов книги.

6) Что такое диапазон?

7) Выделить диапазон:

- B2:F2,
- C1:C10.

8) Ввести информацию:

Адрес ячейки	Вводимый текст
B1	Группа
A3	№
C3	Класс
A5	Оценка
E5	Подпись
E10	имя

Задание 2. Сформируйте структуру таблицы и заполните ее постоянными значениями.

Расписание уроков 7 «А» класса.

понедельник	1. Алгебра	четверг	1. Физика
	2. Физика		2. Физика
	3. Философия		3. Алгебра
	4. Русский язык		4. Алгебра
	5. Русский язык		5. Литература
	6. Литература		6. Литература
вторник	1.Технология	пятница	1. Ритмика
	2.Физкультура		2. Ритмика
	3. Литература		3. Литература
	4. Физика		4. Геометрия
	5. Алгебра		5. Геометрия
	6. Русский язык		6. История
среда	1.Технология	суббота	1. Ритмика
	2.Физкультура		2. Ритмика
	3. Литература		3. Литература
	4. Физика		4. Геометрия
	5. Алгебра		5. Геометрия
	6. Русский язык		6. История

Задание 3. Сформируйте структуру таблицы и заполните ее постоянными значениями.

Средний балл по группам				
Номер группы	Математические дисциплины		Гуманитарные дисциплины	
	Информатика	Высшая математика	История	Философия
Гр. 101	3,5	3,2	4,4	3
Гр. 102	4,5	3,9	4,5	4
Гр. 103	3,9	4,3	3,8	3
Гр. 104	4,9	3,5	4,2	4
Факультет	4,5	3,7	4,2	3,5

Задание 4.

1) Запомнить!

Все числа в ячейках по умолчанию представляются в общем формате и отображаются так, как они вводятся. Затем к введенному числу можно применить любой другой предусмотренный в программе Excel числовой формат.

Некоторые числовые форматы

Общий	Числовой	Денежный	Процентный	Экспотенциальный
1234,1	1 234,10	1 234,10 р.	123410,00 %	1,23E+03

Для изменения формата числового значения, находящегося в текущей ячейке, в окне Формат ячейки следует перейти на вкладку Число и в списке Числовые форматы выбрать строку с названием нужного формата.

2) Создать таблицу.

Задание 5. Создайте таблицу, содержащую данные о расходах студента за неделю. Подсчитайте расходы по отдельным статьям за неделю, а также ежедневные расходы.

Дата	26.фев	27.фев	28.фев	01.мар	02.мар	03.мар	04.мар	За неделю
Продукты	7,00р.	17,50р.	2,00р.	2,00р.		4,00р.	1,70р.	
Транспорт			25,00р.					
Книги				17,40р.				
Развлечения						10,00р.		
Прочие расходы					4,00р.			
Итого								

Задание 6. Составьте таблицу, содержащую цены на мониторы в различных фирмах. Заголовок таблицы отцентрируйте относительно всех столбцов. Измените шрифты следующим образом: наименования мониторов – Times New Roman, цвет коричневый; названия фирм – Courier New, цвет синий, курсив; заголовок таблицы – Arial, цвет красный, полужирный; заголовки столбцов, содержащих минимальную и максимальную цены на монитор данного вида – Arial, зеленый, цвет фона (ячейки) – коричневый. Цены могут быть набраны любым шрифтом, их начертание измените на курсив. Установите размеры шрифтов: для заголовка таблицы – 20 пт; в остальных ячейках – 15 пт.

Цены на мониторы

Модель	Previous	Солярис	Ellipse	Нейтрин	Вектор	Мин.	Макс.
Samsung 14"	205	212	203	207	204		
Samsung 15"	312	322	324	334	320		
Samsung 17"	586	598	567	579	593		
LG 14"	210	217	215	221	210		
LG 15"	316	327	320	312	318		
Daewoo 14"	180	196	188	192	179		

Задание 7. Создать два столбца: первый столбец не по алфавитному списку, а второй столбец по алфавитному списку.

Красноперов А.А.
 Сидоров С.С.
 Снигирев К.Н.
 Половайкина А.К.
 Абросимов Н.Г.
 Яковлев П.Г.
 Григорьев К.Ш.
 Виноградова П.Л.
 Быкова П.Р.
 Акумов М.Р.
 Груздев П.О.
 Дымко А.Р.
 Журавлев Р.О.

Задание 8. Создайте таблицу, содержащую ежемесячные затраты фирмы "Твистор" на проект "Зеленый остров". Найдите также максимальные и минимальные затраты за каждый месяц. Вычислите, каковы были затраты по каждой статье за пять месяцев.

Фирма "Твистор"
 Проект "Зеленый остров"
 Анализ издержек

Наименование	Январь	Февраль	Март	Апрель	Май	Итого
Контракты	3000.00р.	3600.00р.	2700.00р.	2880.00р.	5580.00р.	
Гонорары	1500.00р.	2052.00р.	2658.00р.	2280.00р.	4938.00р.	
Реклама	2880.00р.	2328.00р.	1650.00р.	2394.00р.	4044.00р.	
Фотографии	1043.00р.	585.96р.	338.58р.	562.02р.	90.60р.	
Приемы	531.00р.	453.90р.	589.32р.	298.50р.	887.82р.	
Поездки	1074.00р.	2074.92р.	1550.04р.	1907.94р.	3457.98р.	
Поддержка	2378.00р.	2531.40р.	2705.94р.	2928.00р.	5633.00р.	
Дискеты	1133.94р.	1404.00р.	2052.12р.	1426.00р.	3479.04р.	
Проспекты	2873.70р.	2110.98р.	3465.30р.	3960.00р.	7425.30р.	
Максимально						
Минимально						

Измените шрифты следующим образом: наименования месяцев – Times New Roman, цвет красный, курсив; название фирмы – Arial, цвет зеленый, полужирный, размер шрифта 20 пт.

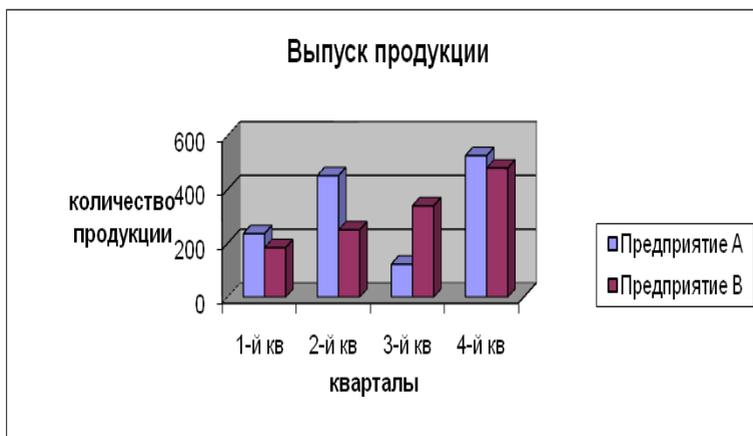
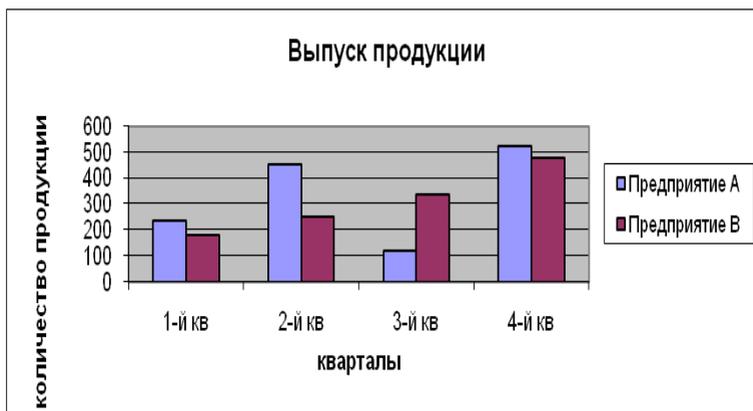
Задание 9. Создать таблицу и произвести расчеты каждого изделия по месяцам и всего за все месяцы. Переименовать Лист1 на "Соловей Ко".

Компания "Соловей Ко" – Данные о реализации

Код изделия	Янв.	Фев.	Мар.	Апр.
*ВН 09	80234.00р.	54900.00р.	50900.00р.	90000.00р.
*ВН 10	70300.00р.	56478.00р.	20000.00р.	80000.00р.
*ВН 11	60600.00р.	68900.00р.	56000.00р.	70000.00р.
*ВН 12	50700.00р.	45000.00р.	45000.00р.	60000.00р.
*ВН 13	30900.00р.	50900.00р.	34000.00р.	50000.00р.
Итого				

Задание 10. Создать таблицу и по данным таблицы создать плоскую и объемную диаграммы.

Выпуск продукции в 1996 году				
Название предприятия	1-й кв	2-й кв	3-й кв	4-й кв
Предприятие А	236	452	123	526
Предприятие В	184	250	340	480



Задание 11. Создайте таблицу и по данным таблицы получите график зависимости расходов от месяца года для каждой из статей расходов (контракты, гонорары и т. д.). Вычислите каковы были затраты на каждую из статей за пять месяцев и постройте по этим данным круговую диаграмму.

Фирма "Твистор"
 Проект "зеленый остров"
 Анализ издержек

Наименования статей расходов	январь	февраль	март	апрель	май	итого
Контракты	3000.00р.	2500.00р.	1900.00р.	1700.00р.	3000.00р.	4567.00р.
Гонорары	2000.00р.	3500.00р.	1800.00р.	1800.00р.	2500.00р.	45689.00р.
Реклама	1000.00р.	1000.00р.	2000.00р.	1900.00р.	1500.00р.	2356.00р.
Фотографии	1500.00р.	1700.00р.	2500.00р.	2000.00р.	1900.00р.	5467.00р.
Приемы	1600.00р.	1500.00р.	2300.00р.	3000.00р.	1600.00р.	3478.00р.
Поездки	1700.00р.	1800.00р.	3000.00р.	1500.00р.	1700.00р.	6789.00р.

Задание 12. Для таблицы постройте два вида диаграмм (любых).

Уровень успеваемости

	Информатика	Высшая математика
Гр.101	3.5	3.2
Гр.102	4.6	3.9
Гр.103	3.9	4.3
Гр.104	4.2	3.5
Факультет	4.05	3.725

Задание 13. По данным таблицы создать три вида диаграмм (круговую, график и объемную гистограмму).

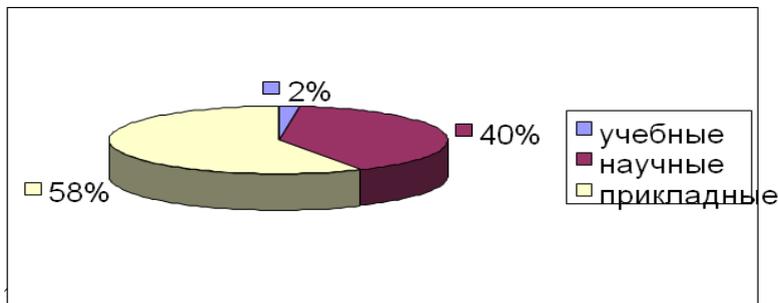
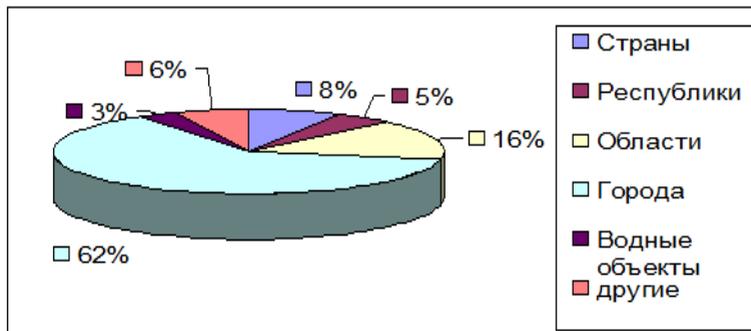
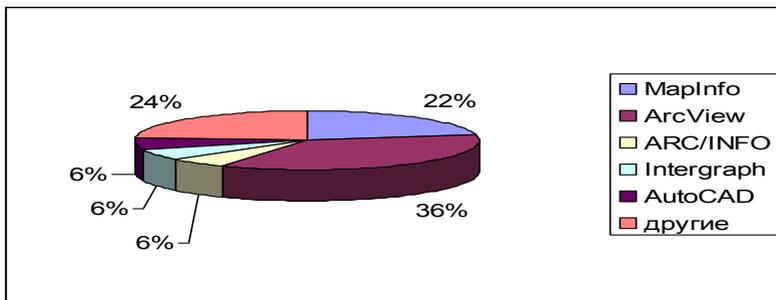
Поток наличности	
Год 1	410 000.00р.
Год 2	460 000.00р.
Год 3	480 000.00р.
Год 4	560 000.00р.

Задание 14. Создайте несколько (минимальное количество – три) диаграмм на основе данных о продажах наиболее ликвидных акций за первые три недели.

Итоги торгов: количество сделок за период

Эмитент	8-14 янв	15-21 янв	22-28 янв
РАО «ЕЭС России»	2211	1365	1995
НК «Лукойл»	1417	1005	1159
Сургутнефтегаз	447	248	228
Ростелеком	398	245	237
Норильский никель	72	28	28

Задание 15. Создать диаграммы, приведенные ниже.



2.5. Тестовые задания

1. Основное отличие электронных таблиц от реляционных баз данных:
 - 1) приспособленность к расчетам;
 - 2) структуризация данных;
 - 3) табличное представление данных;
 - 4) свойства перечисленные в 1, 2.
2. *SuperCalc, QuattroPro, Excel* – это:
 - 1) графические редакторы;
 - 2) СУБД;
 - 3) текстовые редакторы;
 - 4) электронные таблицы.
3. В ячейку электронной таблицы нельзя ввести:
 - 1) текст;
 - 2) формулу;
 - 3) иллюстрацию;
 - 4) число.
4. Операндами формулы в электронных таблицах могут быть:
 - 1) математические функции;
 - 2) константы;
 - 3) номера ячеек;
 - 4) все перечисленное в 1-3.
5. Пользователь может сортировать в электронной таблице:
 - 1) клетки;
 - 2) строки клеток;
 - 3) столбцы клеток;
 - 4) все перечисленное в 1-3.
6. Диапазон ячеек задается через знак:
 - 1) ":";
 - 2) "=";
 - 3) "\$";
 - 4) "/".
7. Абсолютный адрес ячейки создается с помощью знака:
 - 1) "/";
 - 2) "!";
 - 3) "\$";
 - 4) "=".
8. Рабочая страница в электронной таблице Excel содержит:
 - 1) программное окно;
 - 2) внутреннее окно;
 - 3) нижнее окно;
 - 4) правое окно.

9. Создание формулы в электронной таблице Excel начинается с ввода знака:

- 1) ":";
- 2) "=";
- 3) "/"
- 4) "\".

2.6. Контрольные вопросы

1. Что представляет собой электронная таблица?
2. Охарактеризуйте основные возможности табличного процессора.
3. Как осуществляется запуск и завершается работа табличного процессора?
4. Какую структуру имеет рабочая зона окна табличного процессора?
5. Какую структуру имеет строка формул?
6. Что понимают под рабочей книгой? Что такое рабочий лист?
7. Как формируется адрес ячейки?
8. Как выделяется текущая ячейка?
9. Что может находиться в ячейке табличного процессора?
10. Что называется диапазоном ячеек?
11. Опишите структуру окна Параметры (Параметры – *OpenOffice.org Calc*).
12. Как осуществить масштабирование содержимого рабочей зоны окна?
13. Что называется диаграммой?
14. Перечислите основные типы диаграмм в табличном процессоре.
15. Перечислите стандартные элементы, из которых состоят диаграммы.
16. Опишите порядок создания диаграмм в табличном процессоре.
17. Где можно поместить диаграмму?
18. Каким образом можно добавить новые данные к уже существующей диаграмме?
19. Как отредактировать заголовок диаграммы?
20. Как вставить (удалить) линии сетки?
21. Как переместить диаграмму на другое место в пределах одного рабочего листа?
22. Как удалить диаграмму?

3. ВВЕДЕНИЕ В ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ СИ

3.1. Программа на языке Си

Программа на языке Си представляет собой набор последовательно описанных функций (процедуры и подпрограммы в языке Си считаются частным случаем функций). Каждая функция – самостоятельная единица программы, предназначенная для решения определенной задачи (или подзадачи).

При описании она имеет следующий вид:

```
Тип_функции Имя (<список а р г у м е н т о в >)  
<о п и с а н и е а р г у м е н т о в >  
{  
<о п и с а н и я >  
<о п е р а т о р ы >  
}
```

Список аргументов может быть пустым (однако скобки после ее имени сохраняются).

Символы "{" и "}" используются для объединения описаний и операторов в составной оператор или блок, так что все конструкции, заключенные в фигурные скобки, оказываются синтаксически эквивалентными одному оператору. Точка с запятой никогда не ставится после первой фигурной скобки, которая завершает блок.

Существует одна главная функция (с именем *main*), с которой начинается выполнение программы. Функции могут обращаться к другим функциям посредством конструкций вызова. Вызов функции используется при вычислении значения выражения. В результате вызова функция возвращает вычисленное значение, которое и является значением вызова функции. Попутно функция может преобразовывать значения своих аргументов. Такой результат вызова функции называется побочным эффектом.

В модуле, вызывающем данную функцию, тип возвращаемого ею значения должен быть описан (даже если это неопределенное значение) вместе с описанием переменных.

Аргументы (которые бывают формальными и фактическими) передаются по значению путем копирования в соответствующие (по порядку) параметры, указанные в определении функции. Соответствие аргументов и параметров по количеству и типу не контролируется в языке Си.

Операторы предпроцессора могут появляться в любом месте программы, и их действие распространяется на весь исходный файл.

Часто используются следующие операторы предпроцессора:

```
#include  
#define
```

Важная возможность предпроцессора – включение в исходный текст содержимого других файлов. Эта возможность в основном используется для того, чтобы снабжать программы какими-то общими для всех данными определениями. Например, очень часто в начале программы на языке Си встречается предпроцессорная конструкция `#include <stdio.h>`

Когда исходный текст программы обрабатывается предпроцессором, на место этой инструкции ставится содержимое расположенного в некоем стандартном месте (каталоге INCLUDE) файла *stdio.h*, содержащего макроопределения и объявления данных, необходимых для работы функций из стандартной библиотеки ввода-вывода.

Директива *#define* позволяет дать в программе макроопределения (или задать макросы). Оператор макроопределения имеет вид:

```
#define <макроимя> <строка лексем>  
или  
#define <макроимя> (<список параметров>)  
<строка лексем>
```

Макроимя – идентификатор. Строка лексем – последовательность лексем от Макроимени до конца строки. Точка с запятой в конце макроопределения не ставится.

Предпроцессорная обработка макроопределения сводится к тому, что любое появление Макроимени (макровызов) в качестве отдельной лексемы в тексте программы, расположенном после макроопределения, ведет к замене этого Макроимени на указанную Строку лексем.

Оператор присваивания имеет общий вид:

```
<Имя переменной> = <Выражение>;
```

В языке Си разрешается включать присваивания в выражения, то есть присваивание может рассматриваться как операция с учетом старшинства и влияния скобок.

В языке Си имеются специальные операции (`+=` `-=` `*=` `/=` `%=`) для компактной записи операторов присваивания. Так, следующие две

записи на языке Си эквивалентны:

`i=i+2` и `i+=2`.

Операция присваивания сама по себе имеет значения (равное значению выражения, стоящего справа от знака "=") и может входить в выражения.

Оператор *if/else* имеет вид:

```
if (<выражение>
<оператор1>
else
<оператор2>;
```

Здесь часть *else <оператор2>* является необязательной, можно применять и одиночный оператор

```
if (<выражение>) <оператор1>;
```

Вначале вычисляется значение выражения. *Оператор1* выполняется, если значение выражения истинно. Если выражение ложно (его значение равно нулю) и если есть часть с *else*, то выполняется *оператор2*.

При программировании требуется аккуратно различать знаки = и ==, потому что в ряде случаев компилятор не сможет обнаружить ошибки, связанной с неправильным использованием знаков этих операций, что приведет к неверным результатам.

В языке Си имеется компактный способ записи оператора *if/else*. Он называется "условным выражением" или "тернарной операцией". Такое выражение выглядит так:

```
V1 ? V2 : V3
```

Сначала вычисляется значение выражения V1. Если оно отлично от нуля, то вычисляется значение выражения V2, которое и становится значением условного выражения. В противном случае вычисляется значение выражения V3, и оно становится значением условного выражения. Условное выражение удобно использовать в тех случаях, когда имеется некоторая переменная, которой можно присвоить одно из двух возможных значений. Типичным примером являются присваивание переменной значения большей из двух величин:

```
max=(a>b) ?a:b;
```

Оператор-переключатель *switch* удобен в тех случаях, когда в программе необходимо произвести выбор одного из нескольких вариантов. Его синтаксис:

```
switch (<выражение>)
```

```

{
case <константа1>: <список операторов1>;
case <константа2>: <список операторов2>;
. . .
case <константа N>: <список операторов N>
default : <список операторов>;
}

```

Оператор-переключатель выполняется следующим образом. Вычисляется значение выражения в скобках, приведенного после ключевого слова *switch*, затем программа просматривает список меток, указанных после слов *case*, до тех пор, пока не находит ту, которая соответствует данному значению. Далее программа переходит к выполнению оператора, расположенного в этой строке. Если подходящей метки не найдется и если существует строка с меткой *default.*, то будет выполняться оператор, помеченный этой меткой. В противном случае произойдет переход к оператору, расположенному за оператором *switch*.

Оператор цикла *for*

```

for (<оператор1>;<выражение>;<оператор2>)
<оператор3>;

```

позволяет организовать повторяющийся вычислительный процесс. Как правило, *оператор1* и *оператор2* являются операторами присваивания или обращения к функции, а *выражение* – условным выражением.

Цикл *for* удобно использовать в тех случаях, когда заранее известно число повторений тела цикла или имеется явно выраженная переменная, управляющая циклом. В этом случае *выражение* вычисляется один раз и задает инициализацию управляющей переменной. *Выражение* является условием завершения цикла, а *оператор2* задает приращение управляющей переменной.

Любой из операторов и выражений в цикле *for* может быть опущен, хотя точка с запятой при этом должна оставаться. Если отсутствует *оператор1* или *оператор2*, то он просто выпадает из вычислений. Если же отсутствует *выражение*, то считается, что оно всегда истинно.

Оператор цикла с предусловием *while* в общем виде записывается так:

```

while (<выражение>)
<оператор>;

```

Цикл *while* является "условным" циклом, использующим условие на входе. Если *выражение* "истинно", то *оператор*, входящий в цикл *while*, выполняется один раз, а затем *выражение* проверяется снова. Эта последовательность действий, состоящая из проверки и выполнения *оператора*, периодически повторяется до тех пор, пока *выражение* не станет ложным. После этого управление передается оператору, следующему за оператором цикла *while*.

При построении цикла *while* необходимо включить в него какие-то конструкции, изменяющие величину проверяемого выражения так, чтобы, в конце концов, оно стало ложным. В противном случае выполнение цикла никогда не завершится.

Оператор цикла с постусловием – в нем истинность условия проверяется после выполнения каждой итерации цикла. Этот подход реализуется с помощью цикла *do/while*.

Тело цикла *do/while* всегда выполняется, по крайней мере, один раз, поскольку проверка условия осуществляется только после его завершения.

Форма записи:

```
do <о п е р а т о р >  
while (<в ы р а ж е н и е >);
```

Оператор *break* дает возможность выйти из операторов цикла *for*, *while*, *do/while*, а также из переключателя *switch* без проверки условия. Оператор *break* приводит к немедленному выходу из самого внутреннего охватывающего его цикла или из переключателя.

Оператор *continue* вызывает преждевременное завершение выполнение тела цикла и переход к следующему шагу цикла. Оператор *continue* действует только на самый внутренний цикл, частью которого он является.

Оператор безусловного перехода *goto* предназначен для безусловной передачи управления в заданную точку программы. Его выполнение заключается в передаче управления оператору, помеченному заданной меткой.

В качестве метки используется идентификатор. Метка отделяется от оператора, к которому она относится, двоеточием.

Синтаксис оператора:

```
goto <м е т к а >;  
. . .
```

<метка>; <оператор>

Оператор *return* завершает выполнение данной функции и передает управление вызывающей функции. Оператор *return*, в главной функции *main*, вызывает завершение выполнения всей программы.

Оператор *return* может содержать любое выражение:
return (<выражение>;

Если *выражение* не пусто, то вычисляется его значение, которое и становится значением данного вызова функции. Достижение "конца" функции эквивалентно выполнению оператора *return* без возвращаемого значения (т.е. оператор *return* в конце функции может быть опущен). С помощью оператора *return* функции можно вернуть и несколько значений.

Средства ввода-вывода. Имеется ряд библиотечных функций Си, обеспечивающих стандартную систему ввода-вывода для программ на Си. Макроопределения, описания переменных и определения этих функций содержатся в файле стандартных заголовков *stdio.h*. Поэтому каждая пользовательская программа должна содержать в начале ссылку
#include <stdio.h>.

Наиболее часто используется функция форматного вывода *printf*. В общем случае обращение к этой функции имеет вид:

```
printf <формат>, <выражение1>, <выражение2>, .  
. . ., <выражениеN>;
```

где <выражение1>, <выражение2>, . . ., <выражениеN> – произвольные выражения, результаты которых надо вывести. Управляющая строка "формат" содержит объекты двух типов: обычные символы, которые просто копируются в выходной поток (печатаются), и спецификации преобразования значений из внутреннего машинного представления в текстовое для вывода на печатающем устройстве.

Функцией ввода, аналогичной функции вывода *printf()*, является *scanf()* – стандартная функция форматного ввода.

Обращение к этой функции имеет вид:

```
scanf (<формат>, &<имя1>, &<имя2>, . . ., &<имяN>);
```

где <имя1>, <имя2>, . . ., <имяN> – имена переменных, значения которых надо ввести. Наличие символа "&" перед каждым именем обязательно (кроме переменных строкового типа).

При обращении к функции *scanf* выполнение программы

приостанавливается, ожидается ввод значений указанных переменных, после чего работа программы продолжается.

В качестве спецификаций в формате можно использовать те же символы, что и в функции *printf()*. Спецификации формата должны соответствовать количеству и типу вводимых переменных.

Набор стандартных функций ввода и вывода включает большое число функций для работы с данными различного типа, различными устройствами, буферизованного и небуферизованного, форматного и бесформатного ввода и вывода.

Лексемами называют последовательности символов языка (идентификаторы, служебные слова, константы, строки, составные знаки операций, разделители). Лексемы разделяются пробелами и другими неграфическими символами языка.

Идентификатор – это последовательность латинских букв, цифр и символа "_", начинающаяся с буквы или символа "_".

Большие и маленькие латинские буквы считаются различными. Например, *u* и *U* – это разные имена. Рекомендуется в именах переменных использовать только строчные буквы.

Литералы – неизменяемые объекты языка (константы). Литерал может быть либо числовым, либо символьным, либо строковым.

Числовые литералы могут быть десятичными (целыми и вещественными, простыми и длинными), восьмеричными, шестнадцатеричными.

Последовательность символов, заключенных в двойные кавычки, называется **строковым литералом**.

Программа на процедурных языках, к которым относится Си, представляет собой описание операций над величинами различных типов. Тип определяет множество значений, которые может принимать величина, и множество операций, в которых она может участвовать. В языке Си типы связаны с именами (идентификаторами) величин, то есть с переменными. Все переменные должны быть описаны до их использования.

Описание состоит из спецификатора типа и следующего за ним списка переменных. Переменные в списке разделяются запятыми. В конце описания ставится точка с запятой.

Переменным могут быть присвоены начальные значения внутри их описаний. Если за именем переменной следует знак

равенства и константа, то эта константа служит в качестве инициализатора.

Основные типы в языке Си:

int – целый ("integer"); имеются служебные слова, которые можно использовать с типом *int*: *short int* ("short integer" – "короткое целое"), *unsigned int* ("unsigned integer" – "целое без знака"), *long int* ("длинное целое"), которые сокращают или, наоборот, расширяют диапазон представления чисел.

char – символьный ("character"); допустимое значение для этого типа – один символ (не путать с текстом!); символ записывается в апострофах.

float – вещественный (с плавающей точкой). Значения этого типа – числа, но в отличие от *char* и *int*, не обязательно целые.

double – вещественные числа двойной точности; этот тип аналогичен типу *float*, но имеет значительно больший диапазон значений.

В языке Си нет специального типа, который можно было бы использовать для описания строк. Вместо этого строки представляются в виде "массива" элементов типа *char*. Это означает, что символы в строке будут располагаться в соседних ячейках памяти.

Указатель – некоторое символическое представление адреса ячейки памяти, отведенной для переменной. Например, *&name* – указатель на переменную *name*. Здесь *&* – операция получения адреса. Фактический адрес – это число, а символическое представление адреса *&name* является константой типа "указатель".

Для **описания массива** (одномерного и многомерного) после идентификатора переменной при описании ее типа в квадратных скобках указывается максимальное значение индекса элементов массива.

В языке Си предусмотрена возможность **определения имен типов данных**. Любому типу данных с помощью определения *typedef* можно присвоить имя и использовать это имя в дальнейшем при описании объектов.

Формат:

```
typedef <старый тип> <новый тип>
```

Арифметические операции

Сложение	+
Вычитание (бинарное)	-
Умножение	*
Деление	/
Остаток от деления нацело	%
Вычитание (унарное)	-

В языке Си принято правило: если делимое и делитель имеют тип *int*, то деление производится нацело, то есть дробная часть результата отбрасывается.

Специальные операции

Увеличение	++
Уменьшение	--

Следующие записи на языке Си являются эквивалентными:

`i=i+1;` и `i++;`

`j=j-1;` и `j--;`

Символ "++" или "--" записывается после имени переменной или перед ним.

Операции отношения и логические операции

Больше или равно	<code>>=</code>
Больше	<code>></code>
Меньше или равно	<code><=</code>
Меньше	<code><</code>
Равно	<code>==</code>
Не равно	<code>!=</code>
Логическое "и"	<code>&&</code>
Логическое "или"	<code> </code>
Отрицание "не"	<code>!</code>

Логическое значение "ложь" представляется целым нулевым значением, а значение "истина" представляется любым ненулевым значением.

Выражения, связанные логическими операциями `&&` и `||`, вычисляются слева направо, причем вычисление значения выражения прекращается сразу же, как только становится ясно, будет ли результат истинен или ложен.

Старшинство операции `&&` выше, чем у операции `||`.

Битовые операции

Битовое "и"	&
Битовое "или"	
Битовое исключаяющее "или"	~
Сдвиг вправо	>>
Сдвиг влево	<<
Инверсия битов (унарная операция)	\~

Выражения – это конструкции, включающие константы (литералы), переменные, знаки операций, скобки для управления порядком выполнения операций, обращенные к функциям. Если в выражениях встречаются операнды различных типов, то они преобразуются к общему типу в соответствии с определенными правилами:

1) переменные типа *char* интерпретируются как целые без знака (*unsigned*);

2) переменные типа *short* автоматически преобразуются в *int*; если один из операндов имеет тип *unsigned*, то другой (другие) также преобразуются к типу *unsigned* и результат имеет тип *unsigned*;

3) если один из операндов имеет тип *int*, то другой (другие) также преобразуются к типу *int* и результат имеет тип *int*;

4) если один из операндов имеет тип *char*, то другой (другие) также преобразуются к типу *char* и результат имеет тип *char*;

5) во время операции присваивания значение правой части преобразуется к типу левой части, который и становится типом результата;

6) в процессе преобразования *int* в *char* лишние старшие 8 бит просто отбрасываются.

Кроме того, существует возможность точно указывать требуемый тип данных, к которому необходимо привести некоторую величину (в скобках перед этой величиной). Скобки и имя типа вместе образуют операцию, называемую **приведением типов**.

3.2. Задачи и упражнения

1. Написать программу нахождения периметра:

- прямоугольника (ширину и длину вводить с клавиатуры);
- треугольника (длины всех сторон вводить с клавиатуры);
- произвольного четырехугольника.

2. Написать программу для вычисления значения выражения:
 - а) $y=15x^2+8x-9$;
 - б) $a=(b+c)d-k$.
3. Написать программу нахождения наибольшего общего делителя (НОД) двух неотрицательных чисел.
4. Старинная задача. Сколько можно купить быков, коров и телят, если плата за быка 10 руб., за корову – 5 руб., за теленка – полтинник (0,5 руб.), если на 100 руб. надо купить 100 голов скота.
5. Найдите сумму цифр числа.
6. Припишите по **1** в начало и в конец записи числа **n**. Например, было **n=3456**, стало **n=134561**.
7. Поменяйте местами первую и последнюю цифры числа.
8. Поменяйте порядок цифр числа на обратный. Например, было 12345, стало 54321.
9. Найдите количество четных цифр целого положительного числа.
10. Найдите самую большую цифру целого числа.
11. Найдите сумму цифр целого числа, больших 5.
12. Сколько раз данная цифра встречается в целом числе?
13. Составьте программу, проверяющую, является ли последовательность из 10 целых чисел, вводимых с клавиатуры, возрастающей.
14. Найдите НОД трех чисел. Примечание. НОД (a,b,c)=НОД(НОД(a,b),c).
15. Проверьте, являются ли два данных числа взаимно простыми. Два числа называются взаимно простыми, если их наибольший общий делитель равен 1.
16. Найдите наименьшее общее кратное (НОК) чисел **n** и **m**. Примечание. НОК(n,m)=nm/НОД(n,m).
17. Определить, является ли треугольник, заданный длинами его сторон, остроугольным.
18. Вычислить периметр и площадь прямоугольного треугольника по длинам двух катетов.
19. По координатам трех вершин некоторого треугольника найти его площадь и периметр.
20. По длинам двух сторон некоторого треугольника и углу (в градусах) между ними найти длину третьей стороны и площадь этого треугольника.

3.3. Тестовые задания

1. В языке Си лексема – это:
 - 1) набор специальных символов и директив;
 - 2) множество строк, определяющих состояние программы;
 - 3) процедура, выполняющая определенные задания;
 - 4) последовательности символов языка, разделяющиеся пробелами и другими неграфическими символами.
2. В языке Си указатель – это:
 - 1) специальный значок, показывающий, что это динамическая переменная;
 - 2) символическое представление адреса ячейки памяти;
 - 3) символ, указывающий на что-либо;
 - 4) метка.
3. В языке Си литерал – это:
 - 1) переменная зарезервированного типа;
 - 2) неизменяемый объект языка;
 - 3) строка;
 - 4) буква.
4. Комментарии заключаются в скобки:
 - 1) { };
 - 2) /* */;
 - 3) [];
 - 4) /% %/.
5. Идентификатор – это:
 - 1) последовательность латинских букв, цифр и символа "_", начинающаяся с буквы или символа "_";
 - 2) неизменяемые объекты языка (константы);
 - 3) последовательность латинских и русских букв;
 - 4) способ кодирования, допустимые преобразования над значением данной переменной.
6. Фактический адрес в указателях – это:
 - 1) строка;
 - 2) указатель;
 - 3) число;
 - 4) буква.
7. Составной оператор – это:
 - 1) последовательность операторов, заключенная в фигурные скобки { };
 - 2) последовательность операторов, заключенная в квадратные скобки [];

- 3) последовательность операторов, заключенная в операторные скобки *begin...end*;
 - 4) последовательность операторов, заключенная в круглые скобки *()*.
8. Спецификация типа – это:
- 1) задание типа переменной;
 - 2) список переменных;
 - 3) перечисление всех переменных, которые использовались в программе;
 - 4) список типов переменных, которые использовались в программе.
9. Логическое "не равно" обозначается:
- 1) $< >$;
 - 2) \parallel ;
 - 3) $!$;
 - 4) $! =$;
10. Логическое "и" обозначается:
- 1) $=$;
 - 2) \parallel ;
 - 3) $\&$;
 - 4) $\&\&$.
11. Логическое "не" обозначается:
- 1) $!$;
 - 2) $!!$;
 - 3) \parallel ;
 - 4) *not*.
12. Текстовый поток – это:
- 1) логическое понятие, которое система может относить к чему угодно – от дисковых файлов до терминалов;
 - 2) последовательность символов, которая организуется в строки, завершающиеся символами новой строки;
 - 3) последовательность символов, которая организуется в списки слов, завершающиеся точкой с запятой;
 - 4) текст программы.
13. Выражения – это:
- 1) конструкции, включающие константы (литералы), переменные, знаки операций, скобки для управления порядком выполнения операций, обращения к функциям;
 - 2) основные строительные блоки программы; в языке Си указанием на наличие выражения служит символ "точка с запятой", стоящий в конце него;

- 3) набор символов и операций;
 - 4) операторы, выполняющие определенные действия с переменными.
14. Оператор переключатель – это:
- 1) оператор для выбора одного из нескольких вариантов (SWITCH);
 - 2) строка с меткой DEFAULT;
 - 3) CASE;
 - 4) BREAK.
15. Оператор цикла DO/WHILE является:
- 1) конструкцией цикла с предусловием;
 - 2) конструкцией цикла с постусловием;
 - 3) конструкцией цикла с выбором варианта;
 - 4) конструкцией цикла с перебором значений параметра.
16. Формальный аргумент – это:
- 1) конкретное значение, присвоенное этой переменной вызывающей программой;
 - 2) переменная в вызываемой программе;
 - 3) строка, которая пишется в скобках функции;
 - 4) строка, которая пишется в скобках процедуры.
17. Фактический аргумент – это:
- 1) конкретное значение, присвоенное этой переменной вызывающей программой;
 - 2) переменная в вызываемой программе;
 - 3) строка, которая пишется в скобках функции;
 - 4) строка, которая пишется в скобках процедуры.
18. Писать `#include <stdio.h>` нужно для:
- 1) подключения файла, содержащего макроопределения и объявления данных, необходимых для работы функций из стандартной библиотеки ввода-вывода;
 - 2) позволяет дать в программе макроопределения (или задать макросы);
 - 3) переопределения не только константы, но и целых программных конструкций;
 - 4) замены каждого параметра в строке лексем на соответствующий аргумент макровывова.
19. Точка с запятой является:
- 1) разделителем операторов;
 - 2) частью оператора;
 - 3) ключевым знаком языка Си;
 - 4) спецсимволом.

20. Какой тип данных отсутствует в Си в отличие от большинства других языков:
- 1) Real;
 - 2) Integer;
 - 3) String;
 - 4) Char.
21. Символ % сигнализирует программе:
- 1) о начале описания переменных;
 - 2) о начале описания функции;
 - 3) о присваивании переменной значения;
 - 4) начиная с этой позиции, необходимо вывести значение переменной.
22. В языке Си тело функции ограничено операторными скобками:
- 1) begin end;
 - 2) start finish;
 - 3) [];
 - 4) { }.
23. В языке Си программа начинает выполняться с функции:
- 1) Start;
 - 2) Main;
 - 3) Go;
 - 4) Do.

3.4. Контрольные вопросы

1. Охарактеризуйте этапы обработки программы на Си при получении исполняемого модуля.
2. Что такое директивы предпроцессора?
3. Какова структура программы на Си?
4. Какие типы данных используются в Си?
5. Перечислите виды присваиваний в Си?
6. Охарактеризуйте встроенные функции форматного ввода и вывода в Си.
7. Охарактеризуйте встроенные функции неформатного ввода и вывода в Си.
8. Как организуются модули программы Си – функции, определяемые программистом?

4. ОСНОВЫ РАБОТЫ В ИНТЕРНЕТЕ

Значение всемирной сети – Internet исключительно велико. Она заставляет взглянуть на компьютер по-новому, не как на средство переработки информации, а как на средство связи между людьми, находящихся в разных географических точках.

Интернет оказывает значительное влияние на работу специалистов в самых разных областях человеческой деятельности. И роль Интернета в профессиональной деятельности будет только возрастать. В связи с этим большое значение приобретают знание основных возможностей и практические навыки работы в Интернете.

4.1. Основные понятия и определения

ISOC. Координирующим органом сети является организация **ISOC** (от Internet SOCIety – общество Интернета), состоящая из специалистов и заинтересованных лиц разных стран мира. Эта организация выбирает авторитетный орган – IAB (от Internet Architecture Board – совет по архитектуре Интернета), который принимает решения о функционировании сети.

IP-адрес. Любой компьютер, включенный в Интернет по протоколу TCP/IP, имеет свой сетевой адрес, свой «личный номер», который образуется из четырех целых чисел, лежащих в диапазоне от 0 до 255 и отделенных друг от друга точкой. Например, таким адресом может быть 156.94.59.238. Такая система адресации компьютеров в сети позволяет подключить к ней максимум 4294967296 компьютеров. Адреса компьютеров в Интернете называют IP-номерами или IP-адресами.

Когда к Интернету подключается какая-либо крупная региональная или национальная сеть, ISOC выделяет для этой сети соответствующий ее возможным масштабам и потребностям диапазон IP-адресов. Далее руководство региональной сети уже самостоятельно выделяет входящим в нее городским и локальным сетям группы номеров из имеющегося в его распоряжении диапазона. Администраторы локальных сетей по тому же самому принципу выделяют подключаемым к сети компьютерам конкретные IP-адреса.

Существует два способа выделения IP-адреса. Первый – динамическое (временное) назначение, когда IP-адрес выделяется компьютеру только на время сеанса связи с Интернетом. Этот способ

обычно применяется в тех случаях, когда компьютер эпизодически подключается к сети. Второй способ – постоянное закрепление IP-адреса за компьютером. Такой способ применяется в случае постоянной работы компьютера в сети. Компьютер, имеющий как постоянный, так и динамический IP-адрес, является полноправным членом сети.

Компьютер, которому назначен постоянный или динамический IP-адрес, называется **головным, главным, ведущим** или **host-компьютером** (от host – ведущий). Каждый головной компьютер, во-первых, соединен линиями связи с другими головными компьютерами сети, а во-вторых, к каждому головному компьютеру могут быть присоединены компьютеры и терминалы, которые не имеют своих IP-адресов и, значит, не являются полноправными членами сети.

Головные компьютеры, находящиеся в распоряжении отдельных пользователей, подсоединяются к более мощным головным компьютерам локальной сети организации, которые, в свою очередь, соединены линиями связи высокой пропускной способности с головными компьютерами территориальных или национальных сетей. Последние связаны между собой по всему миру. Это обеспечивает связь и обмен информацией между любыми двумя компьютерами в Интернете.

Доменный адрес. IP-адреса используются сетевой аппаратурой для адресации и передачи информации между головными компьютерами сети. Этот способ адресации является самым «удобным» для сетевого оборудования. И в тоже время он не очень удобен для людей, работающих в сети. Чтобы сделать адресацию сети более приемлемой для людей, одновременно с IP-адресами в Интернете введена так называемая **доменная адресация** (от domain – домен, область, владение). В этой системе адресации вся совокупность компьютеров Интернета разбита на большие группы, которые называются доменами верхнего (первого) уровня. В эти группы включаются головные компьютеры сети по территориальному, национальному или какому-либо другому признаку. Каждый домен верхнего уровня имеет **собственное имя**. Эти имена регистрируются в ISOC и закрепляются за соответствующими сетями и организациями на постоянной основе. Примеры имен доменов верхнего уровня приведены в таблице 1. Из таблицы видно, например, что головные компьютеры России находятся в двух доменах *ru* и *su*, головные компьютеры Франции – в домене *fr*. К домену *net* относятся компьютеры сетевых служб

различных стран мира.

Таблица 1. Имена некоторых доменов верхнего уровня

Домен	au	be	ca	dk	it	jp	fr	net	ru	su	uk	us
страна	Австралия	Бельгия	Канада	Дания	Италия	Япония	Франция	Сетевые службы (по всему миру)	Россия	Россия, государства СНГ	Великобритания	США

Каждый домен верхнего уровня может содержать произвольное количество доменов следующего (второго) уровня. Обычно такие домены образуются из крупных территориальных, региональных или городских сетей. Например, в домене верхнего уровня *ru* (Россия) имеются домены: *samara* (Самара), *vologda* (Вологда), *izh* (Ижевск) и т.д. Эти домены, в свою очередь, могут содержать домены еще более низкого (третьего) уровня, соответствующего локальным компьютерным сетям. На самом нижнем уровне имена получают конкретные головные компьютеры. Доменная структура Интернета обычно содержит от двух до четырех уровней. Для задания адреса компьютера в Интернете необходимо последовательно указать имена всех доменов, в которые входит этот компьютер. Перечень имен принято начинать с сетевого имени самого компьютера, а заканчивать названием домена верхнего уровня. Имена в перечне отделяются друг от друга точкой. Такая структура называется **доменным адресом**. Например, адрес: <https://ru.wikipedia.org>

Узел Интернета. Организация, которая обеспечивает подключение отдельных пользователей к Интернету и имеет право выделять постоянные или динамические IP-адреса и/или доменные имена, называется **узлом** Интернета, **поставщиком услуг** Интернета или **провайдером** (от provider – поставщик). Доступ к сети конечным пользователям предоставляется из локальных сетей организаций или же по телефонным линиям. Узлы Интернета рассматриваемого уровня подключаются к более крупным узлам, которые обеспечивают работу группы узлов низшего уровня, разбросанных на определенной

территории. Крупные узлы, в свою очередь, могут присоединяться к национальным узлам, имеющим постоянную связь друг с другом по всему миру.

В состав узла Интернета обязательно входит сервер имен доменов, обеспечивающий соответствие доменных имен компьютеров и их IP-адресов для всех подсоединенных к данному узлу головных компьютеров. Кроме того, в каждом узле выделяется специальный компьютер с таблицами маршрутов, по которым для каждого адреса Интернета определяется следующий ближайший узел сети, на который необходимо переслать информацию, чтобы она, в конце концов, дошла до адресата.

Всемирная паутина. В настоящее время огромной популярностью пользуется гипертекстовая сетевая информационная система **World Wide Web** – Всемирная паутина. Всемирная паутина, которую для краткости называют WWW или Web, представляет собой гипертекстовую информационную систему, содержащую связанные между собой ссылками документы, которые могут создаваться в различных программных средах и находятся в любом из компьютеров сети Интернет.

Интернет и Всемирная паутина это не одно и то же. Интернет включает в себя: огромное множество линий связи, сетевую аппаратуру, компьютеры, данные, находящиеся в этих компьютерах, программное обеспечение и коллективы специалистов, обслуживающих все эти компоненты. Всемирная паутина – это только одна из возможностей Интернета, представляющая собой совокупность организованных в виде гипертекста и связанных между собой данных, расположенных на серверах сети.

Электронная почта. Электронная почта, или e-mail (от electronic mail – электронная почта), представляет собой один из способов связи между людьми. Любой пользователь локальной или глобальной сети, в которой предусмотрена соответствующая услуга, может послать «электронное письмо», то есть текстовое сообщение, которое по каналам связи передается от компьютера отправителя на компьютер адресата. Передача сообщения происходит в течение очень короткого времени. К дополнительным возможностям электронной почты можно отнести: передачу с помощью вложений в письма звуковых сообщений, документов, чертежей, фотографий, видеоматериалов, доступ к научным журналам, редким книгам, рекламу и продажу различных товаров, в том числе программного

обеспечения, поддержку документооборота в учреждениях и организациях, планирование коллективной работы.

Для своевременной и точной доставки сообщений пользователю достаточно правильно указать **электронный адрес** (e-mail) получателя. Правила формирования адреса в Интернете очень просты. Как и адрес в обычной почте, электронный адрес состоит из двух частей: «куда» и «кому». Эти элементы адреса отделяются друг от друга знаком @: кому@куда (например, адрес: askbill@microsoft.com). Каждое отправляемое по электронной почте сообщение принято сопровождать **темой сообщения**, которая представляет собой краткое объяснение его содержания.

Программы, обеспечивающие работу электронной почты в Интернете, построены по принципу клиент-сервер. Посланные пользователями сообщения попадают на специализированный компьютер узла Интернета, который содержит серверную часть программы электронной почты и в связи с этим называется **почтовым сервером**. Почтовый сервер обеспечивает отправку всех **исходящих** сообщений по наиболее подходящим маршрутам, а также прием и хранение до прочтения всех **входящих** сообщений обслуживаемых сервером пользователей.

Все получаемые пользователями **входящие** сообщения не сразу направляются по сетевым линиям связи своим адресатам. Вначале они попадают на почтовый сервер, и некоторое время хранятся там. А отправляемые пользователями **исходящие** сообщения вначале находятся на компьютере пользователя. Доставка почты (входящих и исходящих сообщений) происходит в короткие периоды подключения пользователя к почтовому серверу. Все остальное время связь между компьютером пользователя и почтовым сервером отсутствует. Поэтому говорят, что электронная почта в Интернете представляет собой ресурс, предоставляемый пользователям в **автономном** или **offline** режиме (от offline – вне линии связи).

4.2. Задачи и упражнения

I. Протоколы и настройки обозревателей рабочей станции для работы в Интернете.

1) Проверьте настройки сетевых протоколов вашего компьютера. Для этого, работая в среде *Windows*, откройте панель управления, приложение «Сеть». С помощью какого адаптера ваш компьютер подключен к Интернет? Каковы

настройки протокола IP – как установлен IP-адрес вашего компьютера, маска подсети, DNS-сервер?

2) Проверьте параметры обозревателя Интернет. Для этого откройте панель управления, свойства обозревателя. Какое используется соединение? Каковы настройки локальной сети? Каковы общие настройки обозревателя? Какие программы используются для работы в Интернете? Каков уровень безопасности?

II. Работа с электронной почтой.

1) Создайте тестовое сообщение электронной почты, пошлите его по известному вам адресу. Ответьте на тестовое сообщение, пришедшее на ваш адрес.

2) Сохраните пришедшее письмо в отдельной папке почтовой системы. Напечатайте его. Удалите его. Очистите папку для мусора.

III. Поиск информации в Интернете.

1) Выполните поиск в Интернете виртуальных электронных магазинов. Проверьте, имеется ли в них в продаже данная книга. Сделайте заказ. Оставьте запись в гостевой книге.

2) Соединитесь с поисковым сайтом **http://yandex.ru**. Сформируйте поисковый запрос для методических материалов по информатике. Выполните поиск. Ознакомьтесь с найденными страницами. Можно ли конкретизировать запрос? Выполните поиск с уточненным запросом.

4.3. Тестовые задания

1. В глобальных сетях существуют два режима информационного обмена – это:

- 1) пользовательский и сетевой;
- 2) диалоговый и пакетный;
- 3) информируемый и скрытый;
- 4) диалоговый и пользовательский.

2. Online – это:

- 1) информационная сеть;
- 2) утилита;
- 3) команда;
- 4) режим реального времени.

3. Offline – это:

- 1) операционная система;

- 2) телекоммуникационная сеть;
 - 3) команда;
 - 4) режим информационного пакетного обмена.
4. Мировая система телеконференций:
- 1) EUNET;
 - 2) Fidonet;
 - 3) Relcom;
 - 4) Usenet.
5. BBS – это:
- 1) компьютерная сеть;
 - 2) система телеконференций;
 - 3) электронная доска объявлений;
 - 4) режим работы.
6. BBS предназначена:
- 1) для определения маршрута информации;
 - 2) для обмена файлами между пользователями;
 - 3) для просмотра адресов;
 - 4) для управления информацией.
7. Хост-машина – это:
- 1) машина-хранилище информации;
 - 2) мультимедийный компьютер;
 - 3) компьютерные узлы связи;
 - 4) банк информации.
8. Модем – это:
- 1) хранилище информации;
 - 2) транспортная основа сети;
 - 3) устройство преобразования цифровых сигналов в аналоговые, и наоборот;
 - 4) устройство, которое управляет процессом передачи информации.
9. Функции модема:
- 1) соединяет компьютер с ближайшим узлом;
 - 2) служит сетевой платой для соединения компьютеров в локальную сеть;
 - 3) осуществляет протоколирование передаваемой информации;
 - 4) защищает информацию.
10. Транспортная основа глобальных сетей – это:
- 1) витая пара;
 - 2) коаксиальный кабель;
 - 3) телефонные линии и спутниковые каналы;
 - 4) телеграф.

11. Для связи компьютеров через модемы используются:
 - 1) только телефонные линии;
 - 2) только спутниковые каналы;
 - 3) только радиоволны;
 - 4) телефонные линии, оптоволокно, спутниковые каналы, радиоволны.
12. Технологическая структура e-mail – это:
 - 1) совокупность узловых станций, связывающихся друг с другом для обмена;
 - 2) совокупность компьютеров локальной сети;
 - 3) компьютеры, хранящие и кодирующие информацию;
 - 4) компьютеры, пересылающие информацию по запросам.
13. Типичная абонентская станция электронной почты состоит:
 - 1) из нескольких сетевых компьютеров;
 - 2) из компьютера, специальной программы и модема;
 - 3) из компьютера и почтового сервера;
 - 4) из хост-машин.
14. Типичная структура электронного письма:
 - 1) заголовок, тема сообщения, ФИО адресата;
 - 2) заголовок, тема сообщения, тип письма, адрес отправителя;
 - 3) дата отправления, адрес, обратный адрес, тема сообщения и текст;
 - 4) тема сообщения, адресная книга, текст и заголовок.
15. Домен – это:
 - 1) название файла в почтовом ящике;
 - 2) почтовый ящик узловой станции;
 - 3) код страны;
 - 4) короткое имя адресата.
16. Из чего состоит IP-адрес:
 - 1) адреса сети;
 - 2) последовательности адресов;
 - 3) протоколов;
 - 4) адреса сети и номера хоста.
17. Какой протокол поддерживает Internet:
 - 1) SCP/IP;
 - 2) SCP;
 - 3) TCP/IP;
 - 4) QCP/IP.
18. Для поддержки e-mail в Internet разработан протокол:
 - 1) STTP;
 - 2) SMTP;

- 3) SCTP;
- 4) SSTP.

4.4. Контрольные вопросы

1. Дайте характеристику глобальной сети Интернет?
2. Какова структура сети Интернет?
3. Как формируется и кто выделяет IP-адрес компьютеру?
4. Какие компьютеры Интернета считаются головными?
5. Для чего нужна доменная адресация?
6. Что представляет собой домен верхнего уровня? Приведите примеры.
7. Как формируется доменный адрес компьютера? Приведите примеры.
8. Какую роль в сети играет узел Интернета?
9. Опишите возможные схемы подсоединения пользователей к Интернету.
10. Что представляет собой Всемирная паутина?
11. Что представляет собой веб-страница, сайт, начальная страница, веб-сервер?
12. Опишите основные возможности электронной почты.
13. Что такое сообщение? Из чего оно может состоять?
14. Какую структуру имеют сообщения электронной почты Интернета?
15. Что такое почтовый сервер и какова его роль в работе электронной почты?
16. Почему говорят, что электронная почта работает в автономном режиме?
17. Как формируется адрес пользователя в системе электронной почты Интернета?
18. Для чего нужна тема сообщения?
19. Опишите используемые способы оповещения о прибытии почты.
20. Какие программные средства могут быть использованы для получения доступа к ресурсам Интернета?
21. Как осуществляется подключение к Интернету?
22. Как отключиться от Интернета?

СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ

1. *Немцова Т.И., Назарова Ю.В.* Практикум по информатике. Базовая компьютерная подготовка. - Форум-Инфра-М, 2011. - 320 с.
2. *Берлин А.Н.* Основы информационных технологий. - Интернет-Университет Информационных технологий, 2013. – 504 с.
3. *Кудинов Ю.И., Пащенко Ф.Ф., Келина А.Ю.* Практикум по основам современной информатики. – Лань, 2011. – 484 с.
4. *А.В.Могилев, Н.И.Пак, Е.К.Хеннер.* Практикум по информатике. - Академия, 2010. – 608 с.
5. *Сафронов И. К.* Задачник-практикум по информатике. - БХВ-Петербург, 2002. – 432 с.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1. РАБОТА С ТЕКСТОМ.....	5
1.1. ПРОГРАММЫ ДЛЯ РАБОТЫ С ТЕКСТОМ.....	5
1.2. ОСНОВНЫЕ ВОЗМОЖНОСТИ ТЕКСТОВЫХ РЕДАКТОРОВ	6
1.3. СОЗДАНИЕ ТАБЛИЦ	8
1.4. УПРАЖНЕНИЯ.....	11
1.5. ТЕСТОВЫЕ ЗАДАНИЯ.....	21
1.6. КОНТРОЛЬНЫЕ ВОПРОСЫ	23
2. РАБОТА С ЭЛЕКТРОННЫМИ ТАБЛИЦАМИ.....	25
2.1. ОБЩИЕ СВЕДЕНИЯ О ТАБЛИЧНЫХ ПРОЦЕССОРАХ.....	25
2.2. ИНТЕРФЕЙС ТАБЛИЧНОГО ПРОЦЕССОРА	26
2.3. ОБЩИЕ СВЕДЕНИЯ О ДИАГРАММАХ В ТАБЛИЧНОМ ПРОЦЕССОРЕ.....	28
2.4. УПРАЖНЕНИЯ.....	29
2.5. ТЕСТОВЫЕ ЗАДАНИЯ.....	37
2.6. КОНТРОЛЬНЫЕ ВОПРОСЫ	38
3. ВВЕДЕНИЕ В ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ СИ..	40
3.1. ПРОГРАММА НА ЯЗЫКЕ СИ	40
3.2. ЗАДАЧИ И УПРАЖНЕНИЯ.....	49
3.3. ТЕСТОВЫЕ ЗАДАНИЯ.....	51
3.4. КОНТРОЛЬНЫЕ ВОПРОСЫ	54
4. ОСНОВЫ РАБОТЫ В ИНТЕРНЕТЕ.....	55
4.1. ОСНОВНЫЕ ПОНЯТИЯ И ОПРЕДЕЛЕНИЯ.....	55
4.2. ЗАДАЧИ И УПРАЖНЕНИЯ.....	59
4.3. ТЕСТОВЫЕ ЗАДАНИЯ.....	60
4.4. КОНТРОЛЬНЫЕ ВОПРОСЫ	63
СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ	64

Учебное издание

Составитель
Любовь Александровна Габдуллина

ПРАКТИКУМ ПО ИНФОРМАТИКЕ

Учебно-методическое пособие

Напечатано в авторской редакции с оригинал-макета заказчика

Подписано в печать 17.11.2014. Формат 60x84 1/16. Усл. печ. л. 3,02
Печать офсетная. Уч.-изд. л. 3,08
Тираж 100 экз. Заказ № 1871.

РИО, Типография издательства “Удмуртский университет”
426034, Ижевск, ул. Университетская, 1.