

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Горшкова Надежда Кимовна

Государственное автономное профессиональное образовательное учреждение

Должность: Директор Чувашской Республики «Чебоксарский экономико-технологический колледж»

Дата подписания: 29.01.2026 13:56:44

Министерства образования Чувашской Республики

Уникальный программный ключ:

6e4febd30540ffff35fc4c6217bc0cf1c72a27f9

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ПРАКТИЧЕСКИХ ЗАНЯТИЙ

по учебному предмету

УПУУ.03 ИНФОРМАТИКА

для специальности
среднего профессионального образования

25.02.08 Эксплуатация беспилотных авиационных систем

Чебоксары

2024

Методические указания для студентов к практическим занятиям являются частью программы подготовки специалистов среднего профессионального образования Чебоксарского экономико-технологического колледжа Минобразования Чувашии и составлены на основе Федерального государственного образовательного стандарта среднего профессионального образования (далее – ФГОС СПО) по специальности 25.02.08 Эксплуатация беспилотных авиационных систем, в соответствии с рабочей программой учебного предмета УПУУ.03 Информатика.

Методические указания подготовлены с целью повышения эффективности профессионального образования и самообразования в ходе практических занятий по учебному предмету УПУУ.03 Информатика. Включенные в практические работы задачи стимулируют исследовательскую и творческую деятельность, развивают познавательные интересы, помогают не только глубже понять информатику , но и научиться применять полученные знания на практике.

Методические указания содержат задания к практическим работам, порядок их выполнения, рекомендации, перечень контрольных вопросов по каждой практической работе, требования к знаниям и умениям. Приведен список основной и дополнительной литературы для подготовки к практическим работам.

Методические указания к практическим занятиям предназначены для студентов очной формы обучения.

Организация-разработчик: Государственное автономное профессиональное образовательное учреждение Чувашской Республики «Чебоксарский экономико-технологический колледж» Министерства образования Чувашской Республики.

Рассмотрено и одобрено на заседании цикловой комиссии математических и естественнонаучных дисциплин.

Протокол № ____ от «____» _____ 2024 г.

СОДЕРЖАНИЕ

Пояснительная записка

Перечень лабораторных работ

Общие требования к практическим занятиям

Контроль выполнения практических занятий

Практические работы

Список литературы

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Методические указания по организации выполнения практической работы студентов составлены в соответствии с содержанием рабочей программы учебного предмета УПУУ.03 Информатика по специальности 25.02.08 Эксплуатация беспилотных авиационных систем.

Методические указания разработаны для организации самостоятельной работы студентов и рационального использования времени на овладение содержанием учебного предмета, закрепления теоретических знаний, полученных на аудиторных занятиях.

Практическая работа направлена на достижение студентами результатов освоения учебного предмета УПУУ.03 Информатика согласно требованиям рабочей программы.

В результате освоения учебной дисциплины обучающийся должен уметь:

Код и наименование формируемых компетенций	Планируемые результаты освоения предмета	
	Общие	Дисциплинарные
ОК 01. Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам	<p>В части трудового воспитания:</p> <ul style="list-style-type: none"> - готовность к активной деятельности технологической и социальной направленности, способность инициировать, планировать и самостоятельно выполнять такую деятельность; - интерес к различным сферам профессиональной деятельности, <p>Овладение универсальными учебными познавательными действиями:</p> <p>a) базовые логические действия:</p> <ul style="list-style-type: none"> - самостоятельно формулировать и актуализировать проблему, рассматривать ее всесторонне; - устанавливать существенный признак или основания для сравнения, классификации и обобщения; - определять цели деятельности, задавать параметры и критерии их достижения; - выявлять закономерности и противоречия в рассматриваемых явлениях; - вносить коррективы в деятельность, оценивать соответствие результатов целям, оценивать риски последствий деятельности; 	<ul style="list-style-type: none"> - понимать угрозу информационной безопасности, использовать методы и средства противодействия этим угрозам, соблюдать меры безопасности, предотвращающие незаконное распространение персональных данных; соблюдать требования техники безопасности и гигиены при работе с компьютерами и другими компонентами цифрового окружения; понимать правовые основы использования компьютерных программ, баз данных и работы в сети Интернет; - уметь организовывать личное информационное пространство с использованием различных средств цифровых технологий; понимание возможностей цифровых сервисов государственных услуг, цифровых образовательных сервисов; понимать возможности и ограничения технологий искусственного интеллекта в различных областях; иметь представление об использовании информационных технологий в различных профессиональных сферах

	<p>- развивать креативное мышление при решении жизненных проблем</p> <p>б) базовые исследовательские действия:</p> <ul style="list-style-type: none"> - владеть навыками учебно-исследовательской и проектной деятельности, навыками разрешения проблем; - выявлять причинно-следственные связи и актуализировать задачу, выдвигать гипотезу ее решения, находить аргументы для доказательства своих утверждений, задавать параметры и критерии решения; - анализировать полученные в ходе решения задачи результаты, критически оценивать их достоверность, прогнозировать изменение в новых условиях; - уметь переносить знания в познавательную и практическую области жизнедеятельности; - уметь интегрировать знания из разных предметных областей; - выдвигать новые идеи, предлагать оригинальные подходы и решения; - способность их использования в познавательной и социальной практике 	
ОК 02. Использовать современные средства поиска, анализа и интерпретации	<p>В области ценности научного познания:</p> <ul style="list-style-type: none"> - сформированность мировоззрения, соответствующего современному уровню 	<ul style="list-style-type: none"> - владеть представлениями о роли информации и связанных с ней процессов в природе, технике и обществе; понятиями «информация», «информационный процесс»,

информации и информационные технологии для выполнения задач профессиональной деятельности	<p>развития науки и общественной практики, основанного на диалоге культур, способствующего осознанию своего места в поликультурном мире;</p> <ul style="list-style-type: none"> - совершенствование языковой и читательской культуры как средства взаимодействия между людьми и познаниями мира; - осознание ценности научной деятельности, готовность осуществлять проектную и исследовательскую деятельность индивидуально и в группе; <p>Овладение универсальными учебными познавательными действиями:</p> <p>в) работа с информацией:</p> <ul style="list-style-type: none"> - владеть навыками получения информации из источников разных типов, самостоятельно осуществлять поиск, анализ, систематизацию и интерпретацию информации различных видов и форм представления; - создавать тексты в различных форматах с учетом назначения информации и целевой аудитории, выбирая оптимальную форму представления и визуализации; - оценивать достоверность, легитимность информации, ее соответствие правовым и морально-этическим нормам; - использовать средства информационных и коммуникационных технологий в решении когнитивных, 	<p>«система», «компоненты системы» «системный эффект», «информационная система», «система управления»; владение методами поиска информации в сети Интернет; уметь критически оценивать информацию, полученную из сети Интернет; характеризовать большие данные, приводить примеры источников их получения и направления использования;</p> <ul style="list-style-type: none"> - понимать основные принципы устройства и функционирования современных стационарных и мобильных компьютеров; тенденций развития компьютерных технологий; владеть навыками работы с операционными системами и основными видами программного обеспечения для решения учебных задач по выбранной специализации; - иметь представления о компьютерных сетях и их роли в современном мире; об общих принципах разработки и функционирования интернет-приложений; - понимать основные принципы дискретизации различных видов информации; умение определять информационный объем текстовых, графических и звуковых данных при заданных параметрах дискретизации; - уметь строить неравномерные коды, допускающие однозначное декодирование сообщений (префиксные коды); использовать простейшие коды, которые позволяют обнаруживать и исправлять ошибки при передаче данных;
---	---	--

	<p>коммуникативных и организационных задач с соблюдением требований эргономики, техники безопасности, гигиены, ресурсосбережения, правовых и этических норм, норм информационной безопасности;</p> <p>- владеть навыками распознавания и защиты информации, информационной безопасности личности</p>	<p>- владеть теоретическим аппаратом, позволяющим осуществлять представление заданного натурального числа в различных системах счисления; выполнять преобразования логических выражений, используя законы алгебры логики; определять кратчайший путь возвешенном графе и количество путей между вершинами ориентированного ациклического графа;</p> <p>- уметь читать и понимать программы, реализующие несложные алгоритмы обработки числовых и текстовых данных (в том числе массивов и символьных строк) на выбранном для изучения универсальном языке программирования высокого уровня (Паскаль, Python, Java, C++, C#); анализировать алгоритмы с использованием таблиц трассировки; определять без использования компьютера результаты выполнения несложных программ, включающих циклы, ветвления и подпрограммы, при заданных исходных данных;</p> <p>модифицировать готовые программы для решения новых задач, использовать их в своих программах в качестве подпрограмм (процедур, функций);</p> <p>- уметь реализовать этапы решения задач на компьютере; умение реализовывать на выбранном для изучения языке программирования высокого уровня (Паскаль, Python, Java, C++, C#) типовые алгоритмы обработки чисел, числовых</p>
--	--	--

		<p>последовательностей и массивов: представление числа в виде набора простых сомножителей; нахождение максимальной (минимальной) цифры натурального числа, записанного в системе счисления с основанием, не превышающим 10; вычисление обобщенных характеристик элементов массива или числовой последовательности (суммы, произведения среднего арифметического, минимального и максимального элементов, количества элементов, удовлетворяющих заданному условию); сортировку элементов массива;</p> <ul style="list-style-type: none"> - уметь создавать структурированные текстовые документы и демонстрационные материалы с использованием возможностей современных программных средств и облачных сервисов; умение использовать табличные (реляционные) базы данных, в частности, составлять запросы в базах данных (в том числе вычисляемые запросы), выполнять сортировку и поиск записей в базе данных; наполнять разработанную базу данных; умение использовать электронные таблицы для анализа, представления и обработки данных (включая вычисление суммы, среднего арифметического, наибольшего и наименьшего значений, решение уравнений); - уметь использовать компьютерно-математические модели для анализа объектов и процессов: формулировать цель моделирования, выполнять анализ
--	--	--

		результатов, полученных в ходе моделирования; оценивать адекватность модели моделируемому объекту или процессу; представлять результаты моделирования в наглядном виде
ПК 8.1 Разработка дизайна веб-приложений	<ul style="list-style-type: none"> - разрабатывать дизайн-концепции веб-приложений в соответствии с корпоративным стилем заказчика. - формировать требования к дизайну веб-приложений на основе анализа предметной области и целевой аудитории. - осуществлять разработку дизайна веб-приложения с учетом современных тенденций в области веб-разработки. 	<p>знать:</p> <p>нормы и правила выбора стилистических решений; современные методики разработки графического интерфейса; требования и нормы подготовки и использования изображений в информационно-телекоммуникационной сети "Интернет" (далее - сеть Интернет); государственные стандарты и требования к разработке дизайна веб-приложений.</p> <p>уметь:</p> <p>создавать, использовать и оптимизировать изображения для веб-приложений; выбирать наиболее подходящее для целевого рынка дизайнерское решение; создавать дизайн с применением промежуточных эскизов, требований к эргономике и технической эстетике; разрабатывать интерфейс пользователя для веб-приложений с использованием современных стандартов.</p> <p>иметь практический опыт в: разработке дизайна веб-приложений в соответствии со стандартами и требованиями заказчика; создании, использовании и оптимизации изображений для веб-приложений;</p>

		разработке интерфейса пользователя для веб-приложений с использованием современных стандартов.
--	--	---

ПЕРЕЧЕНЬ ПРАКТИЧЕСКИХ ЗАНЯТИЙ

Наименование тем	Количество часов
Тема 1.1. Информация и информационные процессы. Практическое занятие № 1. Сжатие данных с помощью алгоритма RLE, алгоритма Хаффмана. Сжатие данных с потерями (алгоритмы JPEG, MP3).	2
Тема 1.2. Компьютер – универсальное устройство обработки данных. Практическое занятие № 2. Компьютер и цифровое представление информации. Устройство компьютера.	2
Тема 1.3. Программное обеспечение. Практическое занятие № 3. Инсталляция и деинсталляция программ.	2
Тема 1.4. Компьютерные сети. Практическое занятие № 4. Сетевое администрирование.	2
Тема 1.5. Информационная безопасность. Практическое занятие № 5. Файл как единица хранения информации на компьютере.	2
Практическое занятие № 6. Создание архива данных. Извлечение данных из архива.	2
Практическое занятие № 7. Антивирусные программы. Шифрование данных.	2
Раздел 2. Теоретические основы информатики.	
Тема 2.1. Представление информации в компьютере.	
Практическое занятие № 8. Перевод чисел в разные системы счисления.	2
Практическое занятие № 9. Дискретизация графической, звуковой информации.	2
Тема 2.2. Основы алгебры логики.	
Практическое занятие № 10. Построение и анализ таблиц истинности	2
Тема 2.3. Компьютерная арифметика.	
Практическое занятие № 11. Изучение поразрядного машинного представления целых и вещественных чисел.	2
Тема 2.4. Моделирование.	
Практическое занятие № 12. Поиск выигрышной стратегии в игре с полной информацией.	2
Раздел 3. Алгоритмы и программирование.	
Тема 3.1. Введение в программирование.	
Практическое занятие № 13. Выделение и обработка цифр целого числа в различных системах счисления с использованием операций целочисленной арифметики.	2
Тема 3.2. Элементы теории алгоритмов.	
Практическое занятие № 14. Составление простой программы для машины Тьюринга.	2
Тема 3.3. Вспомогательные алгоритмы.	
Практическое занятие № 15. Использование подпрограмм стандартной библиотеки языка программирования.	2
Тема 3.4. Численные методы.	
Практическое занятие № 16. Численное решение уравнений. Поиск максимума (минимума) функции.	2
Тема 3.5. Алгоритмы обработки символьных данных	
Практическое занятие № 17. Обработка строк с использованием функций стандартной библиотеки языка программирования.	2
Тема 3.6. Алгоритмы обработки массивов.	

Практическое занятие № 18. Заполнение массива. Поиск минимального (максимального) элемента в числовом массиве. Линейный поиск заданного значения в массиве. Простые методы сортировки массива.	2
Тема 3.7. Алгоритмы и структуры данных.	
Практическое занятие № 19. Поиск простых чисел в заданном диапазоне. Анализ текста. Использование деревьев для вычисления арифметических выражений.	2
Тема 3.8. Основы объектно-ориентированного программирования.	
Практическое занятие № 20. Использование готовых классов в программе. Разработка простой программы с использованием классов.	2
Раздел 4. Информационные технологии.	
Тема 4.1. Обработка текстовых документов.	
Практическое занятие № 21. Правила редактирования текста. Создание документа.	2
Практическое занятие № 22. Ввод, редактирование и форматирование документа.	2
Практическое занятие № 23. Создание списков.	2
Практическое занятие № 24. Создание и форматирование таблиц	2
Практическое занятие № 25. Настройка внедренных объектов: формулы, рисунки, диаграммы.	2
Практическое занятие № 26. Специализированные средства редактирования математических текстов.	2
Тема 4.2. Анализ данных.	
Практическое занятие № 27. Встроенные функции и их использование.	2
Практическое занятие № 28. Математические функции и их использование.	2
Практическое занятие № 29. Статистические функции и их использование.	2
Практическое занятие № 30. Решение задач по специальности.	2
Тема 4.3. Компьютерно-математическое моделирование.	
Практическое занятие № 31. Математические модели в экономике.	2
Практическое занятие № 32. Моделирование систем.	2
Тема 4.4. Базы данных.	
Практическое занятие № 33. Работа с готовой базой данных.	2
Практическое занятие № 34. Разработка многотабличной базы данных.	2
Практическое занятие № 35. Запросы к многотабличной базе данных.	2
Практическое занятие № 36. Управление данными с помощью языка SQL.	2
Тема 4.5. Веб-сайты.	
Практическое занятие № 37. Создание текстовой веб-страницы.	2
Практическое занятие № 38. Создание веб-страницы, включающей мультимедийные объекты (рисунки, звуковые данные, видео).	2
Практическое занятие № 39. Оформление страницы с помощью каскадных таблиц стилей.	2
Тема 4.6. Компьютерная графика.	
Практическое занятие № 40. Обработка цифровых фотографий (кадрирование, исправление перспективы, коррекция уровней, коррекция цвета).	2
Практическое занятие № 41. Ретушь цифровых фотографий.	2
Практическое занятие № 42. Многослойные изображения.	2
Практическое занятие № 43. Анимированные изображения.	2
Практическое занятие № 44. Векторная график.	2
Тема 4.7. 3D-моделирование.	
Практическое занятие № 45. Создание простых трёхмерных моделей.	2
Итого	90

Ознакомление с заданием и предварительная подготовка к работе

Практические занятия проводят согласно учебному плану под руководством преподавателя.

1. Предварительная подготовка к выполнению практической работы состоит в следующем:

Преподаватель заранее объявляет о предстоящий практической работе, информирует о содержании и целях работы, порядке ее подготовки и выполнения.

Преподаватель предлагает обучающимся самостоятельное (внеаудиторное) выполнение задания по подготовке к практической работе.

Обучающиеся повторяют теоретический материал к заданной теме, изучают главы параграфов, указанных преподавателем, конспекты.

2. Подготовка и проведение практического занятия.

Преподаватель подробно инструктирует обучающихся о ходе предстоящей работы: называет тему, цели, требования к выполнению работы, особенности заданий, объяснение методов (способов, приемов) их выполнения, критерии оценки.

Преподаватель выдает бланки заданий обучающимся, обучающиеся приступают к выполнению работы: читают задание, задают вопросы, в тетрадь записывают решения, производят расчеты, оформляют ответы и т. д.

В течение практического занятия преподаватель контролирует правильность выполнения заданий, сопровождает дополнительными разъяснениями по ходу работы (при необходимости).

В конце практического занятия проводится подведение итогов, выставляются оценки результатов работы отдельных студентов, ответы на вопросы студентов, выдача рекомендаций по устранению пробелов в системе знаний и умений студентов, по улучшению результатов работы, задание на дом для закрепления пройденного материала и по подготовке к следующему практическому занятию.

3. Требования к выполнению заданий.

Задания необходимо выполнять с максимальной точностью.

Обучающийся должен стремиться к аккуратности, полноте записей. В зависимости от задания, решения должны содержать: расчеты, формулы, заполненные таблицы, графики пр.

КОНТРОЛЬ ВЫПОЛНЕНИЯ ПРАКТИЧЕСКИХ ЗАНЯТИЙ

Критерии оценки

Отметка «5» ставится, если: работа выполнена верно и полностью; в логических рассуждениях и обосновании решения нет пробелов и ошибок; в решении нет математических ошибок (возможна одна неточность, описка, не являющаяся следствием незнания или непонимания учебного материала).

Отметка «4» ставится, если: работа выполнена полностью, но обоснования шагов решения недостаточны (если умение обосновывать рассуждения не являлось специальным объектом проверки); допущена одна ошибка или два-три недочета в выкладках, рисунках, чертежах или графиках (если эти виды работы не являлись специальным объектом проверки); выполнено без недочетов не менее 3/4 заданий.

Отметка «3» ставится, если: допущены более одной ошибки или более трех недочетов в выкладках, чертежах или графиках, но обучающийся владеет обязательными умениями по проверяемой теме; без недочетов выполнено не менее половины работы.

Отметка «2» ставится, если: допущены существенные ошибки, показавшие, что учащийся не владеет обязательными умениями по данной теме в полной мере.

К категории существенных ошибок следует отнести ошибки, которые обнаруживают незнание обучающимися формул, правил, основных свойств, теорем и неумение их применять; незнание приемов решения задач, рассматриваемых в учебниках, а также вычислительные ошибки, если они не являются опиской.

К категории несущественных ошибок следует отнести погрешности, связанные с небрежным выполнением записей, рисунков, графиков, чертежей, а также погрешности и недочеты, которые не приводят к искажению смысла задания и его выполнения.

К недочетам относятся нерациональное решение, описки, недостаточность или отсутствие пояснений, обоснований в решениях.

При наличии существенной ошибки задание считается невыполненным.

Практическое занятие № 1. Сжатие данных с помощью алгоритма RLE, алгоритма Хаффмана. Сжатие данных с потерями (алгоритмы JPEG, MP3).

Цель: Рассмотреть сжатие данных с помощью алгоритма RLE, алгоритма Хаффмана. Сжатие данных с потерями (алгоритмы JPEG, MP3)

Теория

В общем смысле под сжатием данных понимают такое их преобразование, что его результат занимает меньший объем памяти. При этом (по сравнению с исходным представлением) экономится память для их хранения и сокращается время передачи сжатых данных по каналам связи. Синонимы термина “сжатие” – упаковка, компрессия, архивация. Обратный процесс (получение исходных данных по сжатым) называется распаковкой, декомпрессией, восстановлением.

Качество сжатия характеризуется коэффициентом сжатия, равным отношению объема сжатых данных к объему исходных данных.

В зависимости от возможной точности восстановления исходных данных, различают сжатие без потерь (данные восстанавливаются точно в исходном виде) и сжатие с потерями (восстановленные данные не идентичны исходным, но их различиями в том контексте, в котором эти данные используются, можно пренебречь). Сжатие с потерями применяется, например, для упаковки многоцветных фотографических изображений (алгоритм JPEG), звука (алгоритм MP3), видео (группа алгоритмов MPEG). При этом используются особенности человеческого восприятия: например, глаз человека не может различить два близких оттенка цвета, закодированных 24 битами, поэтому можно без видимых искажений уменьшить разрядность представления цвета.

Для многих разновидностей данных – текстов, исполняемых файлов и т.д.
– допустимо применение только алгоритмов сжатия без потерь.

Сжатие без потерь, в основном, базируется на двух группах методов: словарных и статистических. Словарные методы используют наличие повторяемых групп данных и, например, записывают первое вхождение повторяющегося участка непосредственно, а все последующие вхождения заменяют на ссылку на первое вхождение. Другие словарные методы отдельно хранят словарь в явной форме и заменяют все вхождения словарных терминов на их номер в словаре.

Статистические методы используют тот факт, что частота появления в данных различных байтов (или групп байтов) неодинакова, следовательно, часто встречающиеся байты можно закодировать более короткой битовой последовательностью, а редко встречающиеся – более длинной. Часто в одном алгоритме используют и словарные, и статистические методы.

Алгоритм RLE

Самый простой из словарных методов – RLE (Run Length Encoding, кодирование переменной длины) умеет сжимать данные, в которых есть последовательности повторяющихся байтов. Упакованные RLE данные состоят из управляющих байтов, за которыми следуют байты данных. Если старший бит управляющего байта равен 0, то следующие байты (в количестве, записанном в семи младших битах управляющего байта) при упаковке не изменялись. Если старший бит равен 1, то следующий байт нужно повторить столько раз, какое число записано в остальных разрядах управляющего байта.

Например, исходная последовательность 00000000 00000000 00000000 00000000 11001100 10111111 10111011

будет закодирована в следующем виде (выделены управляющие байты): **10000100 00000000 00000011 11001100 10111111 10111011**.

А, например, данные, состоящие из сорока нулевых байтов, будут закодированы всего двумя байтами: **1010 1000 00000000**.

Вопросы и задания:

1. За счёт чего удаётся сжать данные без потерь? Когда это сделать принципиально невозможно?
2. Какие типы файлов сжимаются хорошо, а какие — плохо? Почему?
3. Текстовый файл, записанный в однобайтной кодировке, содержит только 33 заглавные русские буквы, цифры и пробел. Ответьте на следующие вопросы:
 - какое минимальное число битов нужно выделить на символ при передаче, если каждый символ кодируется одинаковым числом битов?
 - сколько при этом будет занимать заголовок пакета данных?
 - при какой минимальной длине текста коэффициент сжатия будет больше 1?
4. На чём основан алгоритм сжатия RLE? Когда он работает хорошо? Когда нет смысла его использовать?
5. Что такое префиксный код?
6. В каких случаях допустимо сжатие с потерями?
7. Опишите простейшие методы сжатия рисунков с потерями. Приведите примеры.
8. На чём основан алгоритм JPEG? Почему это алгоритм сжатия с потерями?
9. Что такое артефакты?
10. Для каких типов изображений эффективно сжатие JPEG? Когда его не стоит применять?
11. На чём основано сжатие звука в алгоритме MP3?
12. Что такое битрейт? Как он связан с качеством звука?
13. Какое качество звука принимается за эталон качества на непрофессиональном уровне?
14. Какие методы используются для сжатия видео?

Практическое занятие № 2. Компьютер и цифровое представление информации. Устройство компьютера.

Цель: рассмотреть цифровое представление информации

Теория

Компьютер – цифровая машина, т. е. внутреннее представление информации в нём дискретно. Для обработки в компьютере любая информация должна быть преобразована в числовую форму (закодирована). Такое преобразование осуществляют специальные программы-компиляторы.

Современные ЭВМ кодируют вводимую информацию с помощью двух состояний:

- есть сигнал или нет его;
- намагничен материал или нет;
- тумблер включен или нет.

Поэтому вся информация в компьютере представлена наборами только из двух знаков (0 – нет сигнала, и 1 – есть сигнал). Отсюда название - *двоичная система счисления*. Например, русская буква "А" в компьютере представлена (кодируется) двоичным числом $11000000_{(2)} = 192_{(10)}$. Одна цифра двоичной системы называется *битом*. Таким образом, **бит** – это наименьшая единица информации.

8 бит составляют 1 *байт*.

На практике используют более крупные единицы информации:

1 Килобайт = 1024 байт

1 Мегабайт = 1024 Кбайт (\approx 400 стр. текста)

1 Гигабайт = 1024 Мбайт (\gg 400 тыс. стр. текста)

Практические задания

Задание №1. Используя таблицу символов, записать последовательность десятичных числовых кодов в кодировке Windows для своих ФИО, названия улицы, по которой проживаете. Таблица символов отображается в редакторе MS Word с помощью команды: вкладка Вставка→Символ→Другие символы

Задание № 2. Запустить БЛОКНОТ. С помощью дополнительной цифровой клавиатуры при нажатой клавише ALT ввести код, отпустить клавишу ALT. В документе появиться соответствующий символ. Используя стандартную программу БЛОКНОТ, определить, какая фраза в кодировке Windows задана последовательностью числовых кодов:

А) Написать расшифровку кода

0223		0243	0247	0243	0241	0252		0226		0192	0207	0210		0239	0238

0239	0240	0238	0244	0229	0241	0241	0232	0232							

0206	0239	0229	0240	0224	0242	0238	0240			0221	0194	0204			

Б) прописать коды букв согласно таблице кодов

П	Р	О	Г	Р	А	М	М	А

Практическое занятие № 3. Инсталляция и deinсталляция программ.

Цели: Закрепить знания о программном обеспечении; изучить методы работы с программным обеспечением. Научиться инсталлировать и deinсталлировать программы.

Теория

Инсталляция и deinсталляция программного обеспечения – это два важных процесса, связанных с установкой и удалением программ на компьютере. Инсталляция – это процесс установки программного обеспечения, который позволяет добавить новое приложение на компьютер. Дeinсталляция – это процесс удаления программного обеспечения с компьютера, который освобождает место на жестком диске и удаляет все связанные с ней файлы и настройки.

При инсталляции программного обеспечения пользователю обычно предлагается выбрать желаемый язык, тип установки (стандартная или настраиваемая) и путь установки. После этого происходит загрузка файлов и их распаковка на жесткий диск компьютера. После успешной завершении инсталляции программа появляется в списке установленных приложений, и пользователь может начать ее использование.

Дeinсталляция программного обеспечения выполняется обратным образом. Пользователь должен выбрать программу, которую он хочет удалить, и запустить процесс deinсталляции. В процессе deinсталляции производится удаление всех связанных с

программой файлов и настроек, и программа исчезает из списка установленных приложений. После успешной деинсталляции пользователь может использовать полученное свободное место на диске для других целей.

Практическое задание

1. Установите программу «FineReader 6.0.Тренажер» из папки «Учебные файлы» Рабочего стола на компьютер. Опишите все этапы установки.
2. Удалите программу «FineReader 6.0.Тренажер» через «Панель управления». Опишите все этапы

Практическое занятие № 4. Сетевое администрирование.

Цель: дать представление о принципах администрирования компьютерных сетей.

Теория

Сетевое администрирование — это область информационных технологий, которая занимается управлением и обслуживанием компьютерных сетей. Сети являются неотъемлемой частью современного мира, они связывают компьютеры и другие устройства, позволяют передавать данные и обмениваться информацией.

Роль сетевого администратора включает в себя различные задачи: настройку сетевого оборудования, установку и обновление программного обеспечения, создание и поддержку безопасности сети, а также решение проблем сетевого подключения и скорости передачи данных.

Сетевое администрирование важно для бизнеса и организаций, так как позволяет обеспечить стабильную работу сети, максимальную производительность и безопасность передачи данных. Без квалифицированных сетевых администраторов компании могут столкнуться со множеством проблем, начиная от нестабильной работы сети и заканчивая утечкой конфиденциальной информации.

Практическое задание

Задание: изучить приемы создания, изменения и удаления учетных записей и групп; изучить приемы задавания и изменения паролей.

Практическое занятие № 5. Файл как единица хранения информации на компьютере.

Цель: дать представление о файловой структуре операционной системы.

Теория

Файл (file) – это определённое количество информации (программа или данные), имеющее имя и хранящееся в долговременной (внешней) памяти. Файл – это наименьшая единица хранения информации, которая хранится, передаётся и обрабатывается как единое целое. В файлах на устройствах внешней памяти хранится всё программное обеспечение компьютера.

С каждым файлом связываются:

- составное имя файла;
- атрибуты файла;
- дата создания;
- время создания;
- длина файла.

Под файловой системой понимают функциональную часть операционной системы, которая обеспечивает выполнение операций над файлами. Файловая система предоставляет средства поиска не одного, а сразу группы файлов путём задания так называемых шаблонов. Шаблоном (образцом) является составное имя файла, в полях имени и/или расширении которого используются символы-заменители. С помощью шаблона обозначается группа файлов, составные имена которых имеют некоторые общие признаки, например файлы, имена которых начинаются с буквы «а» или все файлы с расширением «txt» и др. При этом в самом шаблоне используются символы-заменители – «?» (вопросительный знак) и «*» (звёздочка).

Практическое задание

1. Даны полные имена файлов, хранящихся на диске D: .

D:\COUNTRY\USA\INFO\culture.txt

D:\COUNTRY\USA\Washington.txt

D:\COUNTRY\RUSSIA\Moscow.txt

D:\COUNTRY\RUSSIA\INFO\Moscow.txt

D:\COUNTRY\RUSSIA\culture.txt

Изобразите соответствующую файловую структуру.

2. Пользователь работал с каталогом D:\Программы\Кодеки\xDiv. Сначала он поднялся на два уровня вверх, затем спустился в каталог Игры и после этого спустился в каталог Квесты. Запишите полный путь к каталогу, в котором оказался пользователь.

1) D:\Программы\Кодеки\Игры\Квесты

2) D:\Программы\Игры\Квесты

3) D:\Программы\Игры\Кодеки\Квесты

4) D:\Игры\Квесты

3. Пользователь работал с каталогом D:\Program. Сначала он спустился в каталог Pascal, затем спустился в каталог Array, после чего поднялся на один уровень и спустился в каталог String.

Запишите полный путь к каталогу, в котором оказался пользователь.

1) D:\Program\String

2) D:\Program\Pascal\String

3) D:\Program\Pascal\Array\String

4) D:\String

4. Для групповых операций с файлами используются маски имён файлов. Мaska представляет собой последовательность букв, цифр и прочих допустимых в именах файлов символов, в которых также могут встречаться символы «?» и «*», в том числе «*» может задавать и пустую последовательность. Определите, какое из указанных имён файлов не удовлетворяет маске ??pri*. ?*.

1) caprika.wow

2) weprik.epp

3) otopri.e

4) reprint.be

5. Каталог содержит файлы:

а) z1.pas;

б) z21.ppt;

в) z4.p;

г) z33.p;

д) zad.pas;

е) zom.pp.

При выделении файлов с использованием маски z???.p* список всех выделенных файлов

1) а, б, д

2) а, б, д, е

3) б, г, д, е

4) б, д, е

5) в, г, е

6. Создайте файловую структуру



Содержимое файла **Структура.txt**:

«Одноуровневая файловая структура – это простая последовательность файлов .»

Содержимое файла **Область.docx**:

«Файл – это область на дисках»

Содержимое файла **Организация.txt**:

«Многоуровневая файловая структура – древовидный способ организации файлов на диске»

Задание:

1. Файл из **МАСКА** скопируйте в **ДИСК**
2. Переименуйте копию в **Последовательность.docx**.
3. Скопируйте **Структура.txt** в **ЗАДАЧА**.
4. Переименуйте файл **Организация.txt** в **Древовидный.txt**.
5. Удалите папку **ИНФОРМАТИКА**.
6. Покажите результат преподавателю.
7. Удалите папку **ДИСК**

Практическое занятие № 6. Создание архива данных. Извлечение данных из архива.

Цель: дать представление об архивации данных.

Практические задания

1. Перечислите цели создания архива?

2. Продолжите предложение:

- a) Процесс помещения файлов в архив называется _____
 b) Процесс извлечения файлов из архива называется _____

3. Дайте определения:

Многотомный архив - _____

Самораспаковывающийся архив - _____

Непрерывный архив - _____

4. Перечислите основные характеристики программ-архиваторов:

5. Приведите примеры программ-архиваторов: _____

6. Перечислите форматы архивных файлов

7. Перечислите основные функции архиваторов:

8. От чего зависит степень сжатия файлов? Перечислите все возможные причины.

9. Установите соответствие между утверждением и понятием:

1.	Процесс преобразования информации, хранящейся в файле, с целью уменьшения размера файла и объема памяти, необходимого для ее хранения -	A.	метод Максимальный
2.	Сжатие файла или группы файлов с целью уменьшить место, занимаемое ими на диске -	B.	сжатие информации
3.	Программа, производящая процесс сжатия информации по некоторому алгоритму -	C.	архивирование
4.	Файл, получаемый после работы программы-архиватора -	D.	многотомные архивы
5.	Извлечение файлов из архива -	E.	архиватор
6.	Архивы, состоящие из нескольких частей -	F.	разархивация
7.	Для получения наилучшего сжатия выбирают ...	G.	архивный файл

Ответ: _____

10. Какие параметры архивации на ваш взгляд обеспечат его наибольшую защищенность? _____

11. Перечислите способы извлечения файлов из архива.

12. Проанализируйте работу различных архиваторов при работе с файлами различных форматов

Наименование	Достоинства	Недостатки
WinRAR		
7-Zip		

Практическое занятие № 7. Антивирусные программы. Шифрование данных.

Цель: дать представление о компьютерных вирусах и антивирусных программах.

Практические задания

1. Дайте определение

Компьютерный вирус – это _____

2. Отличительные особенности компьютерных вирусов

3. Вставьте пропущенное слово

a) Программа, внутри которой находится вирус, называется _____

b) _____ вирусы после своего запуска остаются в оперативной памяти и постоянно заражают файлы компьютера до тех пор, пока он не будет выключен или перезагружен

c) Деятельность, направленная на обеспечение конфиденциальности, сохранности и доступности информации называется _____

4. Заполните таблицу классификации компьютерных вирусов

По среде обитания вируса	
По способу заражения	
По деструктивным возможностям	
По особенностям алгоритма вируса	

5. Заполните таблицу

	Файловые	Загрузочные	Макро-вирусы	Сетевые
Куда внедряются				
Начало активизации				
Прекращение активизации				
Профилактика				

6. Поставьте соответствие

1. Компьютерный код, который распространяется без взаимодействия с пользователем по сети	A.	Стелс-вирусы
2. Вредоносная программа, которая скрывается внутри других программ и чаще всего используются для первоначального распространения вирусов, для получения удаленного доступа к компьютеру через Интернет, кражи данных или их уничтожения	B.	Логическая бомба
3. Особенность вируса в том, что при каждом новом заражении они изменяют свои коды	C.	Макровирусы
4. Программа может быть установлена на компьютере без ведома пользователя и может изменять конфигурацию компьютера или собирать публичные и конфиденциальные сведения.	D.	Полиморфные (мутанты)
5. Позволяют полностью или частично скрыть свое присутствие. Наиболее распространенным алгоритмом является перехват запросов ОС на чтение/запись зараженных объектов	E.	Троянский конь
6. Вирусы не способные размножаться из-за существующих в них ошибок.	F.	Вирус-червь
7. Заражают файлы форматов Word, Excel и других приложений пакета Microsoft Office, содержащие Макросы.	G.	Intended-вирусам

8. Проявляет себя только при стечении определенных обстоятельств, распространяется за счет вирусов, вирусов-червей и загружаемого ПО	H.	Программа-шпион
--	----	-----------------

Ответ:

7. Заполните таблицу анализа различных антивирусных приложений.

	ПОЛИФАГИ (сканеры)	РЕВИЗОРЫ (CRC-сканеры)	БЛОКИРОВЩИКИ	ИММУНИЗАТОРЫ
Принцип работы				
Достоинства				
Недостатки				
Примеры программ				

Практическое занятие № 8. Перевод чисел в разные системы счисления.

Цель: Рассмотреть краткую и развернутую форму записи смешанных чисел в позиционных системах счисления. Перевод смешанного числа в позиционную систему счисления с заданным основанием.

Примеры выполнения заданий

- Переведите целые числа из предложенной системы счисления в десятичную

Решение:

$$240_5 = 2 \cdot 5^2 + 4 \cdot 5^1 + 0 \cdot 5^0 = 50 + 20 + 0 = 70_{10}$$

$$37_8 = 3 \cdot 8^1 + 7 \cdot 8^0 = 24 + 56 = 80_{10}$$

251₂ - неправильная запись числа

$$1011_2 = 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 8 + 0 + 2 + 1 = 11_{10}$$

$$7AB_{16} = 7 \cdot 16^2 + 10 \cdot 16^1 + 11 \cdot 16^0 = 1792 + 160 + 11 = 1963_{10}$$

- Переведите целые числа из десятичной системы счисления в предложенную

Решение:

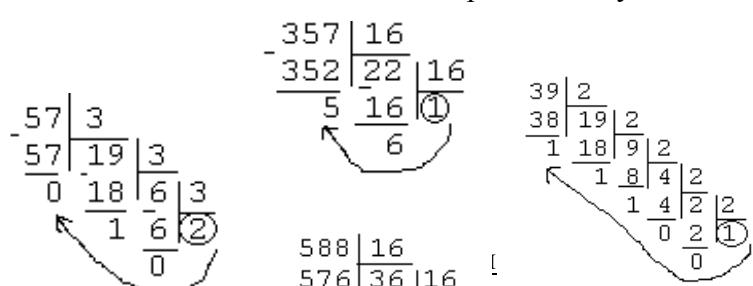
57₁₀ перевести в 3-сс = 2010₃

357₁₀ перевести в 16-сс = 165₁₆

39₁₀ перевести в 2-сс = 100111₂

588₁₀ перевести в 16-сс = 24C₁₆

- Переведите дробные числа:

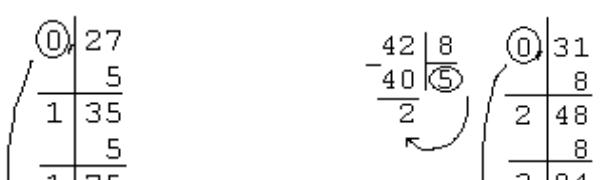


Решение:

$$31,24_5 = 3 \cdot 5^1 + 1 \cdot 5^0 + 2 \cdot 5^{-1} + 4 \cdot 5^{-2} = 15 + 1 + \frac{2}{5^1} + \frac{4}{5^2}$$

$$F3E,2A_{16} = 15 \cdot 16^2 + 3 \cdot 16^1 + 14 \cdot 16^0 + 2 \cdot 16^{-1} + 10 \cdot 16^{-2} = 15,256 + 48 + \frac{2}{16} + \frac{10}{256} = 3902164.._{10}$$

- Переведите дробные числа из десятичной системы счисления в предложенную



Решение:

$$0,27_{10} \text{ перевести в 5-сс} = 0,11(3)_5$$

$$42,31_{10} \text{ перевести в 8-сс} = 52,2365_8$$

Теория:

Таблица сложения в двоичной системе счисления:

$$0 + 0 = 0$$

$$1 + 0 = 1$$

$$0 + 1 = 1$$

$$1 + 1 = 10$$

Т.к. в двоичной системе счисления в записи чисел используются только 2 цифры – 0 и 1, значит при сложении 1 + 1 в младшем разряде записывается 0, а 1 переходит в старший разряд.

По аналогии с 10-СС: 9 + 1 (цифры десять нет в записи чисел), записывается 0 и 1 в старшем разряде, получается 10.

5. Выполните арифметические операции сложения для следующих двоичных чисел:

$$\begin{array}{r} 101101 \\ + 11011 \\ \hline 1001000 \end{array} \quad \begin{array}{r} 101 \\ + 01 \\ \hline 110 \end{array} \quad \begin{array}{r} 101 \\ + 11 \\ \hline 1000 \end{array}$$

6. Выполните арифметические операции вычитания для следующих двоичных чисел:

$$\begin{array}{r} 1011 \\ - 10 \\ \hline 1001 \end{array} \quad \begin{array}{r} 1001 \\ - 10 \\ \hline 111 \end{array}$$

Алгоритм процедуры вычитания в двоичной системе:

$$11101001 - 1001101$$

(обязательно числа нужно представить целым количеством байт)

$$11101001 - 01001101$$

1. Вычитаемое сначала инвертируется: 01001101 превращается в 10110010

2. К нему прибавляется единица: 10110010 + 1 = 10110011 – получается обратный код

3. Выполняется операция сложения 11101001 + 10110011 = 1'10011100

единица из восьмого разряда отбрасывается и остается ответ:

$$10011100$$

Теория:

Операция умножения сводится к поочередному сдвигу чисел и нахождению их суммы.

7. Выполните арифметические операции умножения для следующих двоичных чисел:

$$\begin{array}{r} 101 \\ * 11 \\ \hline 101 \\ + 101 \\ \hline 1111 \end{array} \quad \begin{array}{r} 101 \\ * 101 \\ \hline 101 \\ + 000 \\ \hline 11001 \end{array}$$

8. Выполните арифметические операции деления для следующих двоичных чисел:

$$\begin{array}{r} 10011 \\ \hline 10 \end{array} \left| \begin{array}{r} 10 \\ 1001 \\ -011 \\ \hline 10 \\ -1 \end{array} \right.$$

9. По виду числа определить СС:

- a) 241 – может быть записано в СС, начиная с 5-ричной, т.к. в ней используются цифры 0 1 2 3 4
- b) 73 - может быть записано в СС, начиная с 8-ричной, т.к. в ней используются цифры 0 1 2 3 4 5 6 7

10. Какое число ошибочно записано в:

- a. Троичной СС – 79, 212, 531

Решение: в троичной СС для записи чисел используются цифры 0 1 2 , значит цифры 79 и 531 записаны неверно

- b. Девятеричной СС – 419, 832, 4A

Решение: в девятеричной СС для записи чисел используются цифры 0 1 2 3 4 5 6 7 8, значит цифры 419 и 4A записаны неверно

11. Записать число в виде многочлена:

- a. $143,7_{10}$
- b. $246,5_8$

Решение:

$$143,7_{10} = 1 * 10^2 + 4 * 10^1 + 3 * 10^0 + 7 * 10^{-1}$$

$$246,5_8 = 2 * 8^2 + 4 * 8^1 + 6 * 8^0 + 5 * 8^{-1}$$

12. Когда $5 * 3 = 21$?

Решение:

Перефразируем задание: в какой СС число 15 (ведь $5 * 3 = 15$), записывается как 21?
Попробуем переводить число 15 в различные СС, можно сразу определить в какую.
Задаем вопрос: на что надо разделить число 15, чтобы частное было равно 2.

$$\begin{array}{r} 15 \\ \hline 14 \end{array} \left| \begin{array}{r} 7 \\ 2 \\ \hline 1 \end{array} \right.$$

Значит, в 7-ричной СС число 15 записывается как 21.

13. У меня 100 братьев. Младшему 1000 лет, а старшему 1111 лет. Старший учится в 1001 классе. Может ли такое быть?

Решение:

Такое может быть, если попробовать перевести данные в какую-либо СС. Начнем с двоичной. Представим, что все эти числа записаны в двоичной СС. Переведем их в десятичную.

100 – это число 4

1000 – это число 8

1111 – это число 15

1001 – это число 9

У меня 4 брата. Младшему 8 лет, а старшему 15 лет. Старший учится в 9 классе.

Такое может быть.

14. Расставить знаки арифметических операций так, чтобы были верны следующие равенства в двоичной СС:

$$\underline{1100 ? 11 ? 100 = 100000}$$

Решение:

Перепишем неравенство в десятичной СС и расставим знаки:

$$12 * 3 - 4 = 32$$

Практические задания:

Часть – А

№1. Выполните перевод чисел в соответствующую систему счисления:

$$35_{10} \rightarrow ?_2$$

$$102_{10} \rightarrow ?_8$$

$$55_{10} \rightarrow ?_{16}$$

$$1010_2 \rightarrow ?_{10}$$

$$24_8 \rightarrow ?_{10}$$

$$33_{16} \rightarrow ?_{10}$$

№2. Выполните расчеты

$$11101_2 + 1010_2$$

$$11001_2 + 10110_2$$

$$1100101_2 + 11010_2$$

$$11101_2 - 101_2$$

$$101101_2 - 1010_2$$

$$110101_2 - 10101_2$$

Часть – В

№1. Выполните перевод чисел в соответствующую систему счисления:

2-я СС	8-я СС	10-я СС	16-я СС
	23		
		15,5	

№2. Выполните расчеты

$$11111_2 + 1010_2$$

$$11011_2 + 100110_2$$

$$1100101_2 + 10011_2$$

$$11101_2 - 111_2$$

$$101111_2 - 1010_2$$

$$110111_2 - 1101_2$$

Часть – С

№1. Выполните перевод чисел в соответствующую систему счисления:

2-я СС	8-я СС	10-я СС	16-я СС
1101,1			
		10,25	

№2. Выполните расчеты

$$10111_2 + 111101_2$$

$$10001101_2 + 110110_2$$

$$1100101_2 + 11011_2$$

$$1101000101_2 - 111111000_2$$

$$10110010_2 - 1010001_2$$

$$110111100_2 - 100100010_2$$

Практическое занятие № 10. Построение и анализ таблиц истинности

Цель: рассмотреть операции «импликация», «эквиваленция». Логические функции. Законы алгебры логики. Эквивалентные преобразования логических выражений. Логические уравнения. Построение логического выражения с данной таблицей истинности. Дизъюнктивная нормальная форма. Конъюнктивная нормальная форма. Логические элементы компьютеров. Построение схем из базовых логических элементов. Дискретные игры двух игроков с полной информацией. Выигрышные стратегии.

Практические задания:

1. Отметьте, какие из следующих предложений являются логическими высказываниями

3+5=9	
Луна есть спутник Земли.	
Сегодня идет дождь.	
Если два угла равны между собой, то они вертикальные.	
Студент первого курса.	
Чебоксары – столица Чувашии.	
Все целые числа положительные.	
У нее карие глаза.	

2. Отметьте истинные высказывания

23 – простое число.	
- 6 > 0	
Информатика – это дисциплина.	
Технические средства в английском языке обозначаются словом SoftWare.	
1 Пбайт = 1024 Тбайт	
Информационные технологии – это совокупность методов и устройств, используемых людьми для обработки информации.	
Принтер – печатающее устройство.	
МЭСМ – отечественная машина первого поколения	

3. Приведите примеры

истинные высказывания	
ложные высказывания	

4. Установите соответствие

1.	логическое умножение	A.	дизъюнкция
2.	логическое сложение	B.	инверсия
3.	логическое отрицание	C.	конъюнкция

Ответ:

5. Установите соответствие

1.	высказывание (суждение)	A.	цепочка высказываний или утверждений, определенным образом связанных друг с другом
2.	умозаключение	B.	суждение, которое требуется доказать или опровергнуть
3.	утверждение	C.	некоторое предложение, которое может быть истинно (верно) или ложно
4.	рассуждение	D.	логическая операция, в результате которой из одного или нескольких данных суждений получается (выводится) новое суждение

Ответ:

6. Вставьте пропущенное слово

- a) _____ - это часть электронной логической схемы, которая реализует элементарную логическую функцию.

b) _____ - это табличное представление логической схемы, в котором перечислены все возможные сочетания значений истинности входных сигналов вместе со значением истинности выходного сигнала для каждого из сочетаний.

c) _____ - связывает два простых логических выражения, из которых первое является условием (A), а второе (B) – следствием из этого условия.

7. Составьте таблицу истинности и схему логических элементов

$$\bar{a} \vee \bar{b}$$

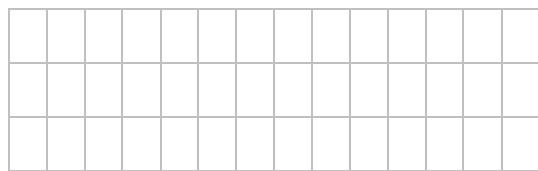
a	b			
0	0			
0	1			
1	0			
1	1			

$$\overline{A} \vee B \wedge \overline{C}$$

A	B	C				
0	0	0				
0	0	1				
0	1	0				
0	1	1				
1	0	0				

A large grid consisting of 40 empty squares arranged in a 5x8 pattern. The grid is defined by thin black lines on a white background.

1	0	1				
1	1	0				
1	1	1				



Практическое занятие № 11. Изучение поразрядного машинного представления целых и вещественных чисел.

Цель: Изучить поразрядное машинное представление целых и вещественных чисел.

Практические задания

1. Определите диапазон целых чисел, хранящихся в 1 байте памяти в двух вариантах: со знаком и без знака.
- 2.. Представьте число 63_{10} в беззнаковом 8-разрядном формате.
3. Найдите десятичные эквиваленты чисел по их прямым кодам, записанным в 8-разрядном формате со знаком:
 - a) 01001100;
 - б) 00010101.
4. Какие из чисел 443_8 , 101010_2 , 256_{10} можно сохранить в 8-разрядном формате?
5. Запишите следующие числа в естественной форме:
 - a) $0,3800456 \cdot 10^2$;
 - б) $0,245 \cdot 10^{-3}$;
 - в) $1,256900E+5$;
 - г) $9,569120E-3$.
6. Запишите число $2010,0102_{10}$ пятью различными способами в экспоненциальной форме.
7. Запишите следующие числа в экспоненциальной форме с нормализованной мантиссой — правильной дробью, имеющей после запятой цифру, отличную от нуля:
 - а) $217,934_{10}$;
 - б) 75321_{10} ;
 - в) $0,00101_{10}$.

Практическое занятие № 12. Поиск выигрышной стратегии в игре с полной информацией.

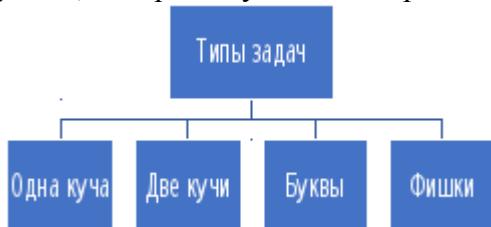
Цель: Рассмотреть поиск выигрышной стратегии в игре с полной информацией.

Теория

Стратегия игрока определяет его действие в любой момент игры и для каждого возможного течения игры.

Игрок имеет *выигрышную стратегию*, если он может выиграть при любых ходах противника.

Описать стратегию игрока – значит описать, какой ход он должен сделать в любой ситуации, которая ему может встретиться при различной игре противника.



Практические задания

1. Попробуйте установить соответствие:

Ход игрока зависит от хода соперника или Ход игрока не зависит от хода соперника

Шахматы

Морской бой

Крестики-нолики

Игры с кубиком и фишками Шашки

2. Двое по очереди ломают шоколадку 5×7 . За ход можно разломать любой кусок по прямой линии между дольками. Проигрывает тот, кто не может сделать ход. Кто выиграет при правильной игре?

Определить:

А) Сколько долек в итоге должно получиться после деления шоколада?

Б) Сколько всего ходов должно быть выполнено?

В) Кто выиграет: первый или второй игрок?

3. Если речь идёт о шоколадке $m \times n$.

При чётном количестве долек - выигрывает _____ игрок

При нечётном количестве долек – выигрывает _____ игрок

4. Стратегия-симметрия

Попробуйте определить: кто и как выиграет при одинаковом количестве предметов в обеих кучках и разном количестве?

5. В ромашке **14** лепестков. Два игрока по очереди отрывают лепестки. Можно отрывать **1** или **2**, но только соседние лепестки. *Проигрывает тот, кто не может сделать ход.*

Кто выигрывает при «правильной игре» тот, кто начинает или тот, кто ходит вторым?

При каком количестве лепестков выигрывает второй игрок(при «правильной» игре)

6. Определить вид задачи:

№1. На столе лежат **2** кучки спичек, по **14** штук в каждой. Два игрока по очереди берут спички. За один ход разрешается взять столько спичек, сколько захочешь, но только из одной кучки. Выигрывает тот, кто берёт со стола последние спички. Как вы думаете: кто выигрывает при «правильной игре» тот, кто начинает или тот, кто ходит вторым?

№2. На столе лежат **12** конфет. Два игрока по очереди берут **1** или **2** конфеты. Выигрывает тот, кто возьмёт последнюю конфету. Как вы думаете: кто выигрывает при «правильной игре» тот, кто начинает или тот, кто ходит вторым?

№3. Имеется три кучки камней: в первой - 10, во второй - 15, в третьей - 20. За ход можно разбить любую кучку на две меньшие. Проигрывает тот, кто не может сделать ход. Кто выиграет?

Практическое занятие № 13. Выделение и обработка цифр целого числа в различных системах счисления с использованием операций целочисленной арифметики.

Цель: Научиться выделять и обрабатывать цифры целого числа в различных системах счисления с использованием операций целочисленной арифметики.

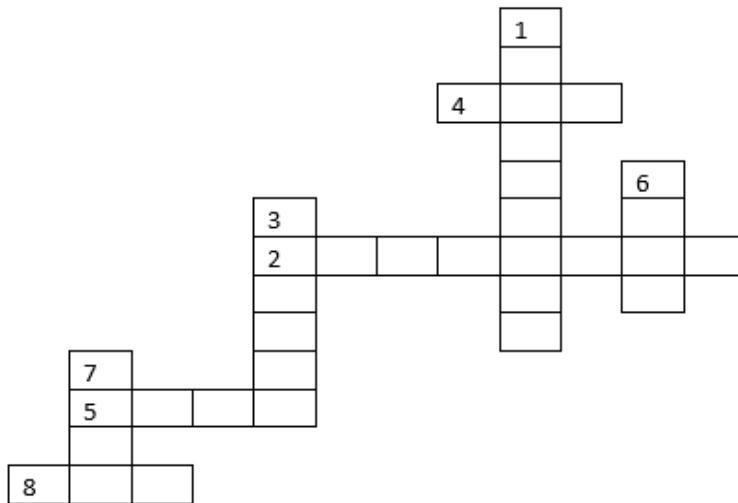
Практическое задание

1. Найти количество единиц в двоичной записи числа, являющегося результатом десятичного выражения

$$2^{4000} + 4^{2016} + 2^{2018} - 8^{600} + 6$$

2. Найти количество цифр в восьмеричной записи числа, являющегося результатом десятичного выражения: $2^{299} + 2^{298} + 2^{297} + 2^{296}$.

3. Решить кросснамбер



По вертикали:

1. Найди сумму и запиши в двоичной системе счисления $153_8 + F9_{16}$
3. Найди произведение и запиши в двоичной системе счисления $122_3 * 11_2$
6. Выполните операцию деления $10010000_2 / 1100_2$
7. Реши пример, ответ запиши в десятичной системе счисления $(564_8 + 234_8) * C_{16}$

По горизонтали:

2. Разность двоичных чисел $11001100 - 11111$
4. Найти разность $167_8 - 56_8$
5. Выполнить операцию деления $41612_8 / 12_8$
8. Найти разность $12E_{16} - 79_{16}$ ответ запиши в десятичной системе счисления

Практическое занятие № 14. Составление простой программы для машины Тьюринга.

Цель: научиться строить машины Тьюринга на специальном тренажере для изучения универсального исполнителя.

Теория

Один из важнейших вопросов современной информатики — существует ли формальный исполнитель, с помощью которого можно имитировать любого формального исполнителя. Ответ на этот вопрос был получен почти одновременно двумя выдающимися учеными — А. Тьюрингом и Э. Постом. Предложенные ими исполнители отличались друг от друга, но оказалось, что они могут имитировать друг друга, а главное — имитировать работу любого формального исполнителя.

Что такое формальный исполнитель? Что значит — один формальный исполнитель имитирует работу другого формального исполнителя? Если Вы играли в компьютерные игры — на экране объекты беспрекословно подчиняются командам играющего. Каждый объект обладает набором допустимых команд. В то же время компьютер сам является исполнителем, причем не виртуальным, а реальным. Вот и получается, что один формальный исполнитель имитирует работу другого формального исполнителя.

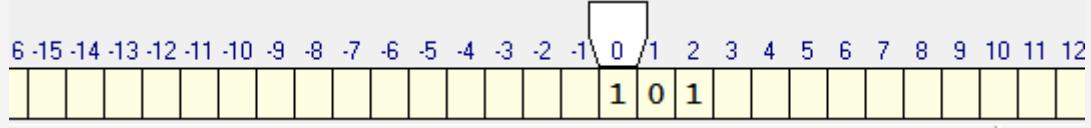
Рассмотрим работу Машины Тьюринга.

Машина Тьюринга представляет собой бесконечную ленту, поделенную на ячейки, и каретку (считывающее-печатывающее устройство), которая движется вдоль ленты.

Таким образом Машина Тьюринга формально описывается набором двух алфавитов:

$A = \{a_1, a_2, a_3, \dots, a_n\}$ — внешний алфавит, служит для записи исходных данных

$Q = \{q_1, q_2, q_3, \dots, q_m\}$ — внутренний алфавит, описывает набор состояний считающее-печатного устройства.



Машина Тьюринга

Каждая ячейка ленты может содержать символ из внешнего алфавита $A = \{a_0, a_1, \dots, a_n\}$ (В нашем случае $A = \{0, 1\}$)

Допустимые действия Машины Тьюринга таковы:

- 1) записать какой-либо символ внешнего алфавита в ячейку ленты (символ, бывший там до того, затирается)
- 2) сместиться в соседнюю ячейку
- 3) сменить состояние на одно из обозначенных символом внутреннего алфавита Q

Машина Тьюринга — это автомат, который управляет табличей.

Строки в таблице соответствуют символам выбранного алфавита A , а столбцы — состояниям автомата $Q = \{q_0, q_1, \dots, q_m\}$. В начале работы машина Тьюринга находится в состоянии q_1 .

Состояние q_0 — это конечное состояние, попав в него, автомат заканчивает работу.

В каждой клетке таблицы, соответствующей некоторому символу a_i и некоторому состоянию q_j , находится команда, состоящая из трех частей

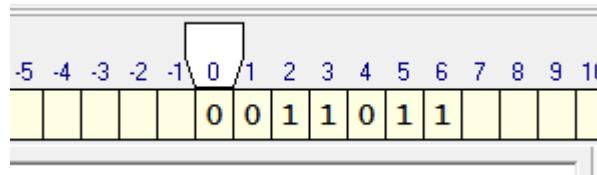
- символ из алфавита A
- направление перемещения: «>» (вправо), «<» (влево) или «.» (на месте)
- новое состояние автомата

	Q ₁	Q ₂	Q ₃	Q ₄
0				
1				
_				

В приведенной выше таблице алфавит $A = \{0, 1, _\}$ (содержит 3 символа), а внутренний алфавит $Q = \{q_1, q_2, q_3, q_4, q_0\}$, q_0 — состояние, заставляющее каретку остановиться.

Практические задания

Задача 1. Пусть $A = \{0, 1, _\}$. На ленте в ячейках находятся символы из алфавита в следующем порядке 0011011. каретка находится над первым символом. Необходимо составить программу, которая заменит 0 на 1, 1 на 0 и вернет каретку в первоначальное положение.



Задача 2. Дано: конечная последовательность 0 и 1 (001101011101). Необходимо выписать их после данной последовательности, через пустую ячейку, а в данной последовательности заменить их на 0. Например:

Из 001101011101 получим 000000000000 1111111.

Практическое занятие № 15. Использование подпрограмм стандартной библиотеки языка программирования.

Цель: Дать представление о подпрограммах и возможностях их использования.

Теория

При создании программы для решения сложной задачи программисты выполняют разделение этой задачи на подзадачи, подзадачи - на еще меньшие подзадачи и так далее, до легко программируемых элементарных задач. Со временем у каждого программиста через некоторое время появляется большой набор собственных заготовок, неординарных решений и т. д., которые он хотел бы использовать во всех своих творениях.

Языки программирования позволяют разделять программу на отдельные части, которые называются подпрограммами. Сам термин подпрограмма говорит о том, что она подобна и подчинена основной программе. Подпрограммы решают три важные задачи, значительно облегчающие программирование:

- избавляют от необходимости многократно повторять в тексте программы аналогичные фрагменты, т. е. сократить объем программы;
- улучшают структуру программы, облегчая понимание при разборе;
- уменьшают вероятность появления ошибок, повышают устойчивость к ошибкам программирования и непредвиденным по следствиям при модификации.

Таким образом, **подпрограмма** - это повторяющаяся группа операторов, оформленная в виде самостоятельной программной единицы. Она записывается однократно, а в соответствующих местах программы обеспечивается лишь обращение к ней по имени.

Практические задания

1. Составить программу сложения 2-х чисел, значения вводятся с клавиатуры.
2. Составить программу нахождения максимального из четырёх чисел, вводимых с клавиатуры.
3. Треугольник задан координатами вершин, вычислить периметр, площадь.
Вычисление длин сторон оформить с помощью подпрограммы.

Практическое занятие № 16. Численное решение уравнений. Поиск максимума (минимума) функции.

Цель: Рассмотреть алгоритмы определения максимального и минимального значений на множестве данных.

Теория

Рассмотрим алгоритм в общем виде:

1. описать для каждого максимума и минимума по одной переменной того же типа, что анализируемые данные;
2. до цикла максимуму присваивается либо *заранее известное* для анализируемых данных значение, либо первый элемент данных; минимуму присваивается либо *заранее известное* для анализируемых данных значение, либо первый элемент данных;
3. в теле цикла каждый подходящий для поиска элемент данных *t* обрабатывается операторами вида:

```
if t>max then max:=t; -- для максимума;  
if t<min then min:=t; -- для минимума,  
где max и min -- переменные, введенные для величин максимума и минимума  
соответственно.
```

Практические задания

Найти минимум функции $y=2x^3 + 10x^2 + 6x - 20$ в интервале $[a,b]$

Практическое занятие № 17. Обработка строк с использованием функций стандартной библиотеки языка программирования.

Цель: рассмотреть алгоритмы обработки строк с использованием функций стандартной библиотеки языка программирования.

Теория

Все библиотечные функции, предназначенные для работы со строками, можно разделить на три группы:

1. ввод и вывод строк;
2. преобразование строк;
3. обработка строк.

Практические задания

1. Определить, сколько раз подстрока встречается в строке.
2. Ввести строку, добавить к ней вторую строку. Определить длину полученной строки.

Практическое занятие № 18. Заполнение массива. Поиск минимального (максимального) элемента в числовом массиве. Линейный поиск заданного значения в массиве. Простые методы сортировки

Цель: рассмотреть алгоритмы поиска минимального и максимального элемента в числовом массиве, простые методы сортировки

Практические задания

1. Вычислить максимальный элемент массива.
2. Вычислить произведение минимального и максимального элементов массива.
3. Получить новый массив путем умножения всех элементов на наименьший элемент.
4. Отсортировать массив по убыванию.

Практическое занятие № 19. Поиск простых чисел в заданном диапазоне. Анализ текста. Использование деревьев для вычисления арифметических выражений.

Цель: рассмотреть алгоритмы поиска простых чисел в заданном диапазоне. Рассмотрение алгоритма выполнения текстового анализа.

Практические задания

1. Напишите программу, которая ищет среди целых чисел, принадлежащих числовому отрезку [0; 400] простые числа.
2. Ввести произвольный текст. Подсчитать сколько раз встречается заданное слово в тексте.

Практическое занятие № 20. Использование готовых классов в программе. Разработка простой программы с использованием классов.

Цель: научиться создавать простейшие объектно-ориентированные программы с классами

Практические задания

Ответить на вопросы

1. Что такое класс?
2. Как объявляется класс?
3. Как называются данные класса?
4. Как называются функции класса?
5. Что называется методами класса?
6. Что такое конструктор класса?
7. Какое имя должно быть у конструктора класса?
8. Что необходимо указывать, если метод определяется вне объявления класса?
9. Что такое закрытые члены класса?
- 10.** Что такое открытые члены класса?

Практическое занятие № 21. Правила редактирования текста. Создание документа.

Цель: изучить интерфейс текстового процессора MS Word, правила редактирования документов.

Теоретическая часть

Основные возможности Текстового редактора MicrosoftWord

MicrosoftWord - текстовый редактор, предназначенный для выполнения всех процессов обработки текста: набора и верстки, проверки орфографии, вставки в текст графики, печати текста. В документах наряду с текстом могут встречаться рисунки, таблицы, формулы.

К основным возможностям программы относятся следующие операции:

- набор и редактирование текста;
- исправление орфографических и грамматических ошибок;
- оформление внешнего вида документа;
- создание таблиц, графиков и рисунков;
- оформление шаблонов деловых писем, визитных карточек и других документов;
- расчёт простейших формул в таблицах;
- слияние документов;
- защита документа паролем;
- вывод документа на печать;
- подготовка pdf-документа;
- совместная работа с документом и т.п.

Документы MicrosoftWord имеют расширения docx.

Набор и корректировка текста

Принято различать в документе MicrosoftWord такие структурные элементы:

1. *символ* - это буква, цифра, знаки препинания, специальные знаки (типа табуляции);
2. *абзац* - это набор символов между двумя признаками конца абзацев;
3. *раздел* - структурная единица документа, имеющая определённые параметры страницы.

Клавиши навигации по тексту

В процессе чтения и корректировки текста необходимо свободно перемещаться по тексту, используя соответствующие клавиши.

Стрелка влево / стрелка вправо - переход по символам.

Ctrl+стрелка влево / стрелка вправо - переход по словам.

Home / End - переход в начало / конец строки.

Ctrl+Home / End - переход в начало / конец документа.

Стрелка вверх / вниз - переход по строкам.

Ctrl+стрелка вверх / стрелка вниз - переход к началу предыдущего / следующего абзаца.

PgUp / PgDn - переход по экранным страницам.

Ctrl+PgUp / PgDn - переход по реальным печатным страницам документа.

Alt+стрелка вверх / стрелка вниз - переход по предложениям.

F5 или Ctrl+G - перейти на страницу по номеру, введя номер страницы, нажав Enter, затем нажав клавишу Esc.

Shift+F5 - при открытии существующего документа вернуться к месту последних сделанных изменений

Практические задания

1. Создайте новый документ и именем *Редактирование* в своей сетевой паке WORD.
Ведите текст.

Рабочее окно MicrosoftWord.

Окно программы состоит из строки заголовка, ленты вкладок, рабочего поля и строки состояния. В строке заголовка находятся название файла документа, имя программы, стандартные значки "Свернуть", "Развернуть", "Восстановить", "Закрыть". При запуске

программы создаётся новый пустой документ с названием Документ 1, расположенный в рабочем поле.

Под строкой заголовка находится лента, состоящая из нескольких вкладок. С помощью опций на ленте можно выполнять различные операции с текстом: изменять шрифт, размер, цвет, создавать таблицы, вставлять рисунки и многое другое. Для перехода на ленту нажмите клавишу Alt или F10. Под лентой вкладок находится рабочее поле, в котором осуществляется набор и форматирование текста. При открытии документа курсор находится в первой позиции первой строки.

Ниже рабочего поля располагается строка состояния, в которой отображается информация о документе: количество страниц и номер текущей страницы, язык ввода текста, статистика, состояние проверки правописания, кнопки режимов отображения документа.

2. На следующей странице ранее созданного документа введите текст:

- [1] По умолчанию установлен режим просмотра документа "Разметка страницы".
- [2] Web - документ - применяется для публикации документа в виде Web-страницы.
- [3] Существуют следующие режимы просмотра документов:
- [4] Режим чтения - используется для чтения документа.
- [5] Черновик - используется для быстрого редактирования документа.
- [6] Структура - документ отображается в виде структуры, содержащей главы, параграфы, пункты.
- [7] Разметка страницы - видно, как документ выглядит на печатной странице с колонтитулами и номерами страниц.

Разместите пронумерованные фрагменты текста в следующем порядке: первый абзац – [3]; второй – [7], третий – [2], четвертый – [4], пятый – [6], шестой – [5], седьмой – [1]

Контрольные вопросы

1. Что такое зона выделения?
2. Каковы наиболее часто используемые способы выделения текста?
3. Как выполняется форматирование абзаца? Каковы его основные возможности?

Практическое занятие № 22. Ввод, редактирование и форматирование документа.

Цель: изучить правила редактирования и форматирования документов.

Теоретическая часть

Работа с фрагментами текста

С фрагментами текста можно выполнять следующие операции:

1. копировать и перемещать, как в рамках одного документа, так и переносить их в другие документы;
2. удалять;
3. заменять вновь набираемым текстом;
4. изменять параметры форматирования.

Для совершения операций с фрагментом текста необходимо вначале его выделить.

Практические задания

Задание 1. Наберите текст, примените видоизменения по образцу

ВИДОИЗМЕНЕНИЕ

Зачеркнутый

Двойное зачеркивание

Надстрочный

Подстрочный

МАЛЫЕ ПРОПИСНЫЕ

ВСЕ ПРОПИСНЫЕ

Задание 2. Наберите текст, примените соответствующий вид регистра

РЕГИСТР

как в предложениях

Все Строчные

все прописные

начинать с прописных

изменить регистр

Задание 3. Наберите текст, примените соответствующий вид текстового эффекта

ТЕКСТОВЫЕ ЭФФЕКТЫ

Сплошная заливка текста красным цветом

Контур текста сплошная желтая линия

Тень – заготовки - снаружи

Тень – заготовки - внутри

Перспектива

Отражение – заготовки - Полное отражение - касание

Свечение и сглаживание – заготовки

Задание 4. Наберите текст. Для заголовка используйте шрифт Cambria, 15, Ж. Для основного текста используйте шрифт Таahoma, 14, К. Подчеркните названия атрибутов символов двойной синей линией.

Атрибуты формата отдельных символов:

шрифт — общий дизайн символов (вид шрифта);

размер — высота символов, измеряемая в пунктах (1 пункт = 1/72 дюйма);

начертание — внешний вид (обычный, полужирный, курсив и т. д.);

подчеркивание символа — одинарное, двойное, пунктирное или только слова (пробелы не подчеркиваются);

эффекты — выделение символов: зачеркивание, верхний индекс, нижний индекс, скрытый, малые прописные и все прописные;

цвет — цвет символов на экране монитора или на цветном принтере;

интервал — расстояние добавляемое или отнимаемое от межсимвольного интервала для получения растянутого или сжатого текста;

смещение — расстояние, на которое символы поднимаются или опускаются, при этом размер символов не уменьшается;

кернинг — символы в некоторых парах (например, A и W) сдвигаются ближе. Word автоматически регулирует кернинг пар символов, размер которых равен или превышает значение, указанное в поле «Пунктов и более».

Практическое занятие № 23. Создание списков.

Цель: получить представление о работе со списками в MS Word, изучить команду Формат-Список, назначение кнопок панели инструментов форматирование

Теория

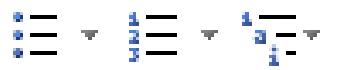
Правила оформления различных документов иногда требуют наличия в документах списков. Принято выделять три типа списков:

Маркированные списки(или списки-буллетени) используются при перечислении или выделении отдельных фрагментов текста.

Нумерованные списки полезны в тех случаях, когда нужно определить порядок изложения.

Многоуровневые (или иерархические) списки, т.е. имеющие несколько уровней. В таких списках допустимы как нумерованные элементы, так и символы маркера.

Для создания списков используются кнопки на панели инструментов Абзац



Практические задания:

Задание 1. Наберите названия 12 месяцев оформите их в виде маркированных и нумерованных списков по образцу:

➤ январь	☺ январь	⊕ январь
➤ февраль	☺ февраль	⊕ февраль
➤ март	☺ март	⊕ март
➤ апрель	☺ апрель	⊕ апрель
➤ май	☺ май	⊕ май
➤ июнь	☺ июнь	⊕ июнь
➤ июль	☺ июль	⊕ июль
➤ август	☺ август	⊕ август
➤ сентябрь	☺ сентябрь	⊕ сентябрь
➤ октябрь	☺ октябрь	⊕ октябрь
➤ ноябрь	☺ ноябрь	⊕ ноябрь
➤ декабрь	☺ декабрь	⊕ декабрь
1) январь	A. январь	Первый месяц. январь
2) февраль	B. февраль	Второй месяц. февраль
3) март	C. март	Третий месяц. март
4) апрель	D. апрель	Четвертый месяц. апрель
5) май	E. май	Пятый месяц. май
6) июнь	F. июнь	Шестой месяц. июнь
7) июль	G. июль	Седьмой месяц. июль
8) август	H. август	Восьмой месяц. август
9) сентябрь	I. сентябрь	Девятый месяц. сентябрь
10) октябрь	J. октябрь	Десятый месяц. октябрь
11) ноябрь	K. ноябрь	Одиннадцатый месяц. ноябрь
12) декабрь	L. декабрь	Двенадцатый месяц. декабрь

Задание 2. Оформите многоуровневый список

1) Столбец однотипных данных в Access называется:

- a) записью;
 b) бланком;
 c) полем;
 d) отчетом.
- 2) Стока, описывающая свойства элемента таблицы, называется:
 a) полем;
 b) бланком;
 c) записью;
 d) ключом.

Практическое занятие № 24. Создание и форматирование таблиц

Цель: получить представление о работе с таблицами в MS Word, изучить команды пункта меню Таблица, назначение кнопок панели инструментов таблицы и границы

Теория

Таблица Word состоит из строк и столбцов ячеек. Таблицы могут содержать цифры, текст и рисунки. Таблицы используются для упорядочения и представления данных.

Для создания новой таблицы Word используется панель инструментов **Таблицы** вкладки **Вставка**. Существует три способа создания таблиц:

1. Нарисовать
2. Вставить
3. Создание на основе существующих данных (текста, чисел)

Практические задания

Задание 1. Нарисуйте таблицу. Залейте каждую строчку разным цветом.

Задание 2. Вставьте таблицу. Оформите внешние границы двойной линией, шириной 1,5 mm.

Задание 3. Вставьте таблицу состоящую из 4 столбцов и 6 строк. Примените любой стиль таблицы.

Задание 4. Вставьте таблицу и оформите её в соответствии с образом

- Шрифт TimesNewRoman 12
- Ширина таблицы 13см.
- Раскрасьте ячейки таблице как показано ниже на рисунке 1

День недели	Время	Задания	Выполнено
Понедельник	8.00 – 15.20	Обучение Word	Сделано
Вторник	8.00 – 16.00	Обучение Excel	В процессе
Среда	8.00 – 15.20	Обучение Access	Ожидается
Четверг	8.00 – 16.00	Обучение Outlook	Изучается
Пятница	8.00 – 15.20	Обучение Power Point	Сделано
Суббота		Выходной	Сделано
Воскресенье		Выходной	Сделано

Рисунок 1 Пример таблицы

Практическое занятие № 25. Настройка внедренных объектов: формулы, рисунки, диаграммы.

Цель: получить представление о технологии OLE, внедрении графических объектов, изучить способы создания и редактирования графических изображений, редактор формул

Практические задания

- Наберите объявление



• разговорный язык и письмо
 • индивидуально и в группах
 • опыт работы
 • апробированная методика

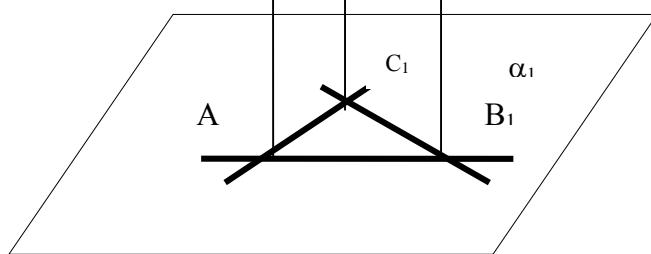
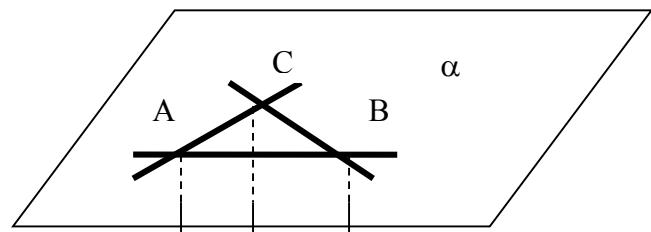
 123-45-67

123-45-67 англ.язык							
------------------------	------------------------	------------------------	------------------------	------------------------	------------------------	------------------------	------------------------

- Наберите формулировку теоремы и выполните к ней чертеж в соответствии с предложенным образцом

Теорема 16.1

Пересекающиеся прямые,
соответственно
параллельные
перпендикулярным
прямым, сами



Практическое занятие № 26. Специализированные средства редактирования математических текстов.

Цель: Рассмотреть средства создания и редактирования математических текстов. Работа с формулами. Вставка и редактирование формул.

Теория

Вставка формул.

Вставка формул в редакторе WORD осуществляется с помощью формульного редактора.

Вызов формульного редактора Equation Editor из Word можно осуществить следующей последовательностью действий:

- поместите курсор в то место, где должна быть вставлена формула;
- в меню "вставка" выберите команду "объект";
- выберите закладку "создание";
- В окне "тип объекта" выберите "Microsoft Equation";

Более удобный вызов редактора математических формул выполняется при помощи кнопки редактора формул, которая помещается на панель инструментов. Размещение кнопки на панели выполняется следующим образом: через меню Вид/Панели инструментов/Настройка на вкладке Команды отыскивается в списке категорий пункт Вставка, в котором выбирается кнопка Редактор формул и перетаскивается на любую панель инструментов.

В результате вызова Редактора формул на экране появляется панель инструментов, состоящая из двух рядов пиктограмм:

В первом ряду расположено 10 пиктограмм, за каждой из которых находится группа символов (математические операции, греческие символы и т. д.) Во втором ряду находятся пиктограммы для вызова шаблонов наиболее распространенных структурных формул (матрицы, суммы и т. д.). Кроме того главное меню Word заменяется на меню редактора математических формул.

Задание 1.

С помощью формульного редактора Equation Editor наберите формулу:

$$\chi^2 = \sum_{i=1}^n \left(\frac{X_i - \mu}{\sigma} \right)^2$$

Методические указания по выполнению задания 1.

Вызовите формульный редактор;

В палитре шаблонов выберите третий слева шаблон с индексами;

В открывшемся списке выберите левый в первом ряду;

Наберите греческое "Хи" (в палитре символов второе поле справа - греческие символы).

Обратите внимание на различный вид курсоров. Вводимый символ вставляется в позицию, определяемую вертикальной чертой курсора!

Подведите курсор в поле верхнего индекса и нажмите 2;

Ведите "=" после X;

— из палитры шаблонов выберите знак суммы с верхним и нижним индексами, и введите индексы;

Выберите из палитры шаблонов объект с верхним индексом (первый в четвёртом ряду);

Выберите шаблон со скобками;

Выберите шаблон для дроби;

— Выберите шаблон с нижним индексом, введите Хи, переведите курсор в следующую позицию (стрелкой -> или щелчком мыши), наберите "-", затем "m"

В знаменателе введите " σ "

В месте верхнего индекса наберите 2;

Выходите из редактора формул, щёлкнув левой кнопкой мыши вне поля редактирования.

Сохраните формулу в файле.

Задание 2.

Наберите систему линейных уравнений в матричной записи в виде:

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{pmatrix}$$

Задание 3. Наберите систему неравенств

$$\begin{cases} \frac{5 + \sqrt{25 - 4p}}{2p} < 0, \\ \frac{5 - \sqrt{25 - 4p}}{2p} > 0. \end{cases}$$

Задание 4. Наберите формулу вычисления корней квадратного уравнения

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Задание 5. Наберите формулу вычисления консолидированного платежа

$$S = \sum_{j=1}^m S_j (1 + pt_j / K) + \sum_{j=m+1}^n S_j (1 + pt_j / K)^{-1}.$$

Задание 6. Наберите текст решения уравнения

$$\left(\log_{1,5} \frac{12}{-3-x} = \log_{1,5} (1-x) \right) \Leftrightarrow \begin{cases} \frac{12}{-3-x} = 1-x, \\ -3-x > 0, \\ 1-x > 0, \end{cases} \Leftrightarrow$$

Практическое занятие № 27. Встроенные функции и их использование.

Цель: Изучить интерфейс приложения, технологию обработки числовой информации. Ввод и редактирование данных. Автозаполнение. Форматирование ячеек. Стандартные функции.

Теория

Excel – программа, предназначенная для обработки и анализа числовой информации, хранимой в электронных таблицах в среде Microsoft Windows.

Microsoft Excel позволяет:

- создавать широкий спектр документов табличного вида; ёмкость электронной таблицы достаточную для реализации большинства документов;
- группировать документы в рабочие книги;
- представлять табличную информацию в графической форме (в виде диаграмм различного типа);
- выполнять сортировки и фильтрации данных в таблицах;
- создавать связанные и сводные таблицы;
- проводить аналитические исследования с помощью:
 - формул и связей;
 - диаграмм;
 - программы «поиск решения»;
- проводить статистический анализ данных и т.д.

Список расширений файлов, которые будут формироваться при работе с Excel:

.xls – файл таблицы (электронная таблица)

.xlw – расширение для определенного пользователем объединения нескольких файлов, которые должны обслуживаться под одним общим именем (рабочая папка);

.xlt – файл шаблона (шаблоны – это файлы, содержащие заголовки для часто используемого бланка).

Итак. Из всех выше перечисленных возможностей Excel рассмотрим самую первую, или, другими словами, попробуем создать небольшой табличный документ.

Пример 1. Создание простейшего документа в Excel .

Основные изучаемые операции:

- запуск Microsoft Excel;

- создание рабочей книги;
- определение параметров страницы;
- выполнение простейших операций по вводу и редактированию данных;
- создание автоматически вычисляемых значений (итогов);
- использование стандартных функций;
- оформление заголовков документов.

Пусть требуется создать документ о результатах продаж соответствующих товаров по месяцам за полугодие:

месяц	январь	февраль	март	апрель	май	июнь	итого
продукция							
наименование продукции 1	1	1	1	1	1	1	6
наименование продукции 2	2	2	2	2	2	2	12
наименование продукции 3	3	3	3	3	3	3	18
Всего	6	6	6	6	6	6	36

Для того чтобы создать этот документ, необходимо загрузить Microsoft Excel.

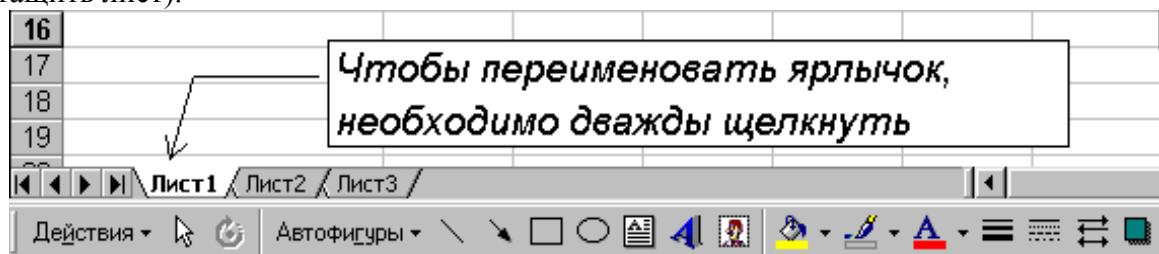
Запуск Excel:

Вариант 1. на рабочем столе Windows дважды щелкнуть по значку Microsoft Excel.

Вариант 2. через кнопку Пуск. Выбрав пункт меню Программы, найти Microsoft Excel.

При запуске Excel создает новую незаполненную рабочую книгу.

Рабочая книга – это обычный документ или тип файла в Microsoft Excel. Она представляет собой электронный эквивалент папки – скоросшивателя. Книга состоит из листов. Имя каждого листа выводится на ярлычок в нижней части рабочей книги. Листы можно перемещать и копировать из одной книги в другую, а также менять их порядок в пределах одной книги (нажав кнопкой мыши на наименование листа и не отжимая кнопку, перетащить лист).



Пусть первый активный лист будет называться «Общая сводка за полугодие».

Переименуйте его.



В Excel работают в основном с рабочими листами. Рабочий лист представляет собой сетку, содержащую 256 столбцов и 65536 строк. Пересечение столбца и строки образует ячейку – основной элемент электронной таблицы для хранения данных. Каждая ячейка имеет свой уникальный адрес или ссылку. Например, ячейка, находящаяся на пересечении столбца С и строки 4, имеет адрес C4. Эти адреса используются при записи формул или ссылках на ячейки. Обычно вначале выделяется ячейка или ячейки, с которыми будут работать, а затем вводятся данные или выбирается команда. Выбранные ячейки выделяются на экране

подсветкой. **Активной ячейкой** называется та, в которую в данный момент вводится информация или которая выделена толстой рамкой.

A	B	C	D	E
1				
2				
3				
4				
5				
6				
7				

Ввод данных в таблицу.

Прежде чем создавать таблицу, необходимо проанализировать характер вводимых данных.

В выполняемом документе можно выделить такие группы:

- заголовок документа;
- наименование строк и столбцов таблицы («шапка» таблицы);
- текстовые данные (наименование продукции);
- числовые данные (результат продаж за месяц – задаваемые значения);
- итоговые данные (суммы по строкам и столбцам – вычисляемые значения).

Каждая из выделенных групп данных требует определенных знаний и приемов при вводе и оформлении.

Заголовок таблицы.

Неизвестно какой ширины будет таблица, поэтому заголовок следует разместить в первой колонке и пока не выравнивать по центру страницы.

Определите, в какой строке будет располагаться заголовок, например в 3-й строке. Активизируйте ячейку A3 (щёлкнуть кнопкой мыши по ячейке A3). Наберите требуемый текст: «Общая сводка за полугодие»

A	B	C	D	E	F
1					
2					
3	Общая сводка за полугодие				
4					
5					

Для того чтобы отредактировать введенный текст, необходимо активизировать ячейку, к которой он привязан, и затем либо дважды щёлкнуть кнопкой мыши по тексту, либо перейти в строку ввода текста и формул.

Обратите внимание, что, хотя текст и занимает на экране несколько ячеек, он всегда привязан (адресуется) только к одной ячейке.

A	B	C	D	E	F	G	H
1							
2							
3	Общая сводка за полугодие						
4							
5							
6							

Заголовок привязан к ячейке A3

Создание наименований столбцов таблицы.

В формируемом документе наименование столбцов таблицы занимает две строки.

Заполните соответствующие ячейки названиями столбцов, начиная с 6-й строки столбца

A.

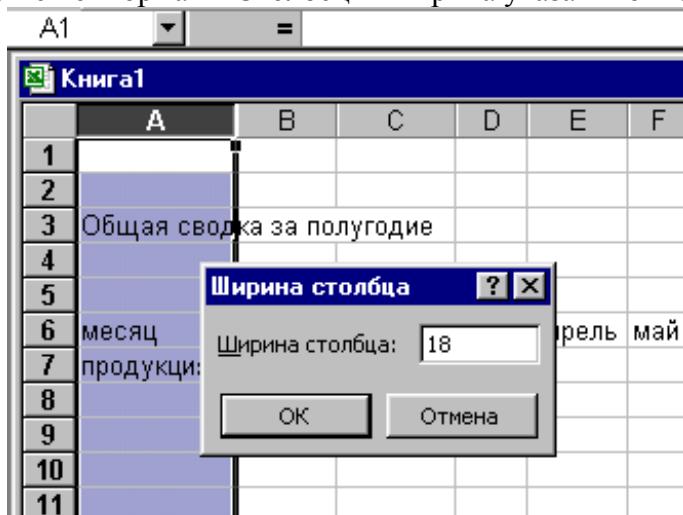
	A	B	C	D	E	F	G	H	I
1									
2									
3	Общая сводка за полугодие								
4									
5									
6	месяц	январь	февраль	март	апрель	май	июнь	итого	
7	продукция								
8									

По умолчанию ширина всех столбцов однааковая. Для создаваемого документа ширина первого столбца, содержащего наименование продукции, безусловно, должна быть больше, чем ширина остальных столбцов, которые заполняются числовыми данными.

Изменение ширины столбца

Вариант 1. Установить курсор на разделительную линию между столбцами А и В и, когда курсор мыши примет вид черной двунаправленной стрелки, не отжимая кнопку мыши, переместить границу на требуемое расстояние (ширина столбца указывается в окне подсказки).

Вариант 2. Выделить столбец А, нажав кнопкой мыши на наименование столбца, а затем через меню Формат – Столбец – Ширина указать точное значение ширины

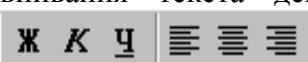


Форматирование ячеек.

Для выравнивания текста по правому краю следует активизировать ячейку А6 и нажать



на пиктограмму . Пиктограммы выравнивания текста действуют только для форматирования данных внутри ячеек таблицы.

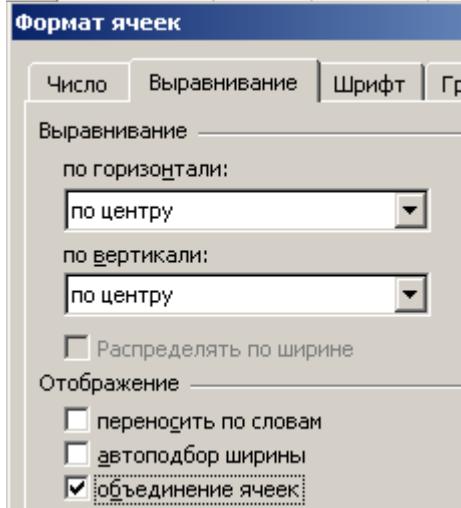


Объединение ячеек

Наименование столбцов с названиями месяцев требуется расположить в центре объединенных ячеек 6-й и 7-й строк.

Выделить ячейки В6-В7, через меню Формат – Ячейки перейти на закладку Выравнивание и отметить параметр Объединение ячеек или нажать на кнопку на панели инструментов Форматирование . Операции объединения выполнить для остальных заголовков столбцов.

3	Общая сводка за полугодие
4	
5	
6	месяц
7	январь
8	февраль
	март



Центрирование данных.

Выделить ячейки, в которых требуется отцентрировать данные. Через меню Формат – Ячейки перейти на закладку Выравнивание и установить параметры: по горизонтали – по центру, по вертикали – по центру.

5
6
7
8
месяц
январь
февраль
март
апрель
май
июнь
итого

Перенос текста внутри ячеек.

Ячейки с наименованиями продукции (A8 – A10) содержат текст, который необходимо расположить с переносом в несколько строк и отцентрировать.

Для переноса текста внутри ячеек необходимо:

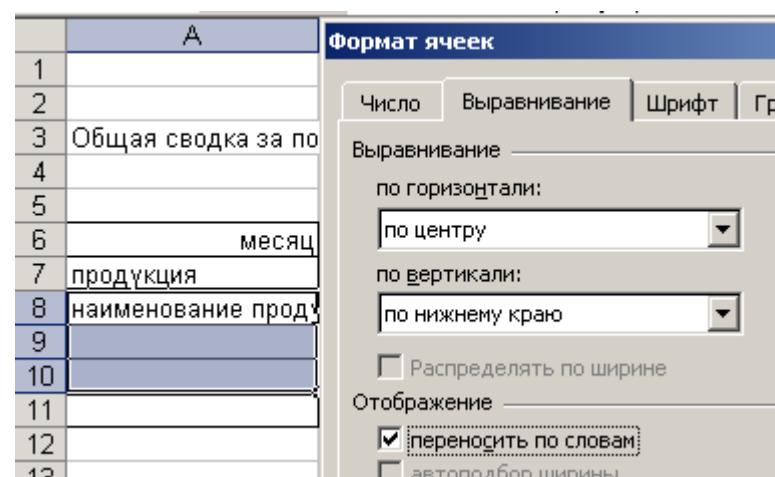
- выделить ячейки, в которых требуется перенос текста;
- через меню Формат – Ячейки перейти на закладку Выравнивание;
- отменить параметр переноса по словам;
- задать параметры Выравнивания: по горизонтали – по центру, по вертикали – по нижнему краю.

6	месяц	январь	ф
7	продукция		
8	наименование продукции 1		

Копирование данных.

Текст в ячейках А8 – А10 повторяется за исключением цифр, поэтому целесообразно скопировать в две нижележащие ячейки и затем отредактировать. Для выполнения копирования требуется:

- активизировать ячейку, текст будет копироваться;
- через меню Правка – Копировать сохранить выделенную информацию в буфере временной памяти;



- выделить ячейки A9 – A10 и, указав меню Правка – Вставить, заполнить ячейки текстом.

Выполните редактирование данных в ячейках A9 – A10, заменив в них номер продукции.

В ячейку A11 введите Всего.

Ввод и форматирование числовых данных.

В ячейки B8 – B10 введите соответственно числа 1-3.

Скопируйте значения из ячеек B8 – B10 в соответствующие ячейки по месяцам.

Отформатируйте значения в интервале ячеек B8 – G10: выравнивание - по центру ячеек, начертание – жирным.

	A	B	C	D	E	F	G	H
6	месяц	январь	февраль	март	апрель	май	июнь	
7	продукция							итого
	наименование							
8	продукции 1	1	1	1	1	1	1	
9	наименование							
	продукции 2	2	2	2	2	2	2	
10	наименование							
	продукции 3	3	3	3	3	3	3	
11	Всего							

Ввод формул.

Формула – это комбинация констант, вычислительных операторов и функций, которая вводится в ячейку для получения нового результирующего значения.

Вид формул может изменяться в зависимости от решаемых задач за счёт использования ссылок, функций, текста и имён.

Арифметические операторы :

- + сложение
- вычитание
- / деление
- * умножение
- % процент
- ^ возвведение в степень.

Операторы сравнения:

- = равно
- > больше
- < меньше
- >= больше или равно
- <= меньше или равно
- <> не равно

Оператор текста.

& соединяет или сцепляет два или более текстовых значения в единое текстовое значение.

Порядок вычисления операторов.

- % процент
- ^ возвведение в степень
- * и / умножение и деление
- + и – сложение и вычитание
- & присоединение текста
- =<> <=> = <> сравнения.

Круговые скобки изменяют порядок вычисления: в первую очередь выполняются действия, объединённые в формуле круглыми скобками.

В создаваемом документе требуется подсчитать сумму по строкам и столбцам графах “Итого” и “Всего”.

Для того чтобы подсчитать сумму за январь (сумма по столбцу), необходимо:

- установить курсор в ячейку (активизировать ячейку), где будет размещена сумма (ячейка B11);
- нажать на верхней панели на пиктограмму Σ автосуммирования;
- завершить формирование суммы нажатием Enter.

- скопируйте содержимое ячейки B11 в ячейки C11 – G11.

	A	B	C	D
6	месяц	январь	февраль	март
7	продукция			
8	наименование продукции 1	1	1	1
9	наименование продукции 2	2	2	2
10	наименование продукции 3	3	3	3
11	Всего	=СУММ(B8:B10)		

Обратите внимание, как изменились ссылки на суммируемые ячейки. Ссылки позволяют использовать в одной формуле данные из различных областей, а так же значения из одной ячейки в нескольких формулах.

Типы ссылок.

Относительные ссылки подобны определению навигации от текущего положения, т.е. такая ссылка определяет, как найти требуемую ячейку, стартуя из ячейки, содержащей формулу. Например, A6, F17.

Абсолютные ссылки определяют, как найти ячейку, отсчитывая ее местоположение от начала электронной таблицы. В абсолютных ссылках перед именем столбца и номером строки ставится знак \$ (доллар). Например, A\$6\$, F\$17\$.

Смешанные ссылки содержат комбинации относительных и абсолютных ссылок. Например, A6\$, F\$17\$.

Различие между относительными и абсолютными ссылками проявляется, когда формулы копируются или пересыпаются из одной ячейки в другую.

Типы операторов ссылок

Оператор интервала (двоеточие) определяет одну ссылку на все ячейки, расположенные между двумя ссылками включая последние (например: A2:C2. Результат: выделены ячейки A2, B2, C2).

Оператор объединения (точка с запятой) определяет одну ссылку, объединяя ячейки, на которые указывают две исходные ссылки (например: A2; C2. Результат: выделены ячейки A2 и C2).

Оператор пересечения (пробел) определяет одну ссылку на ячейки, общие для двух ссылок (например: A5:D5 C4:C9. Результат: выделена ячейка C5).

Изменение типов ссылки

Чтобы поменять тип ссылок, который используется в формулах, необходимо:

f _x	=СУММ(\$B\$8:\$B\$10)
B	C D E

- Выделить ячейку или интервал ячеек, содержащих формулы;
- В строке формул выделить всю формулу либо только ту ее часть, где необходимо поменять тип ссылки;
- Нажатием клавиши F4, циклически изменения типы ссылок, выбрать нужный тип;

- Нажатием на Enter завершить изменение ссылки.

Создание формул с использованием встроенных функций

Осталось ввести формулы для вычисления сумм по строкам в столбец «Итого» (ячейки H8-H11).

Для этого требуется:

- активизировать ячейку H8;
- перейти в строку ввода формул;
- ввести знак = (ввод любой формулы начинается со знака равно);
- выбрать через меню Вставка - Функция вычисление суммы;
- указать интервал ячеек, значения которых требуется просуммировать (переместить мышкой диалоговое окно функции и выделить требуемый интервал ячеек, который будет размещен в поле функции для суммирования);
- для завершения создания формулы нажать Enter.

Скопируйте созданную формулу в ячейки H9-H11. Отформатируйте значения столбца «Итого»: выравнивание - по центру ячеек, начертание - жирным.

СУММ X ✓ fx =СУММ(B8:G8)

	A	B	C	D	E	F	G	H	I	J	K
6	месяц	январь	февраль	март	апрель	май	июнь	итого			
7	продукция										
8	наименование продукции 1	1	1	1	1	1	1	B8:G8)			
9	наименование продукции 2										
10	наименование продукции 3										
11	Всего										
12											
13											

Центрирование заголовка

После того как таблица готова, можно оформить заголовок:

- выделить ячейки в строке заголовка (3-я строка) до границы листа;
- через меню Формат - Ячейки перейти на закладку Выравнивание и поставить опцию Объединение ячеек, затем установить параметры Выравнивание по горизонтали - по центру, Выравнивание по вертикали - по центру.

A	B	C	D	E	F	G	H
1							
2							
3	Общая сводка за полугодие						
4							

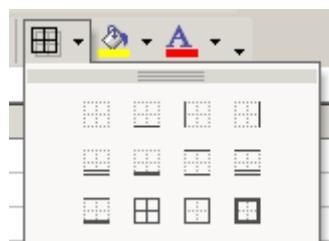
Документ готов. Нажмите на клавишу просмотра документа и оцените свои труды. Очевидно, что не хватает обрамления табличной части документа.

Обрамление таблиц

Для этого требуется:

- Выделить ячейки, входящие в табличную часть документа (A6-H11);
- Раскрыть пиктограмму обрамления и выбрать тип обрамления для создания границ таблицы.

Нажмите на клавишу для просмотра документа.



выбрать

Практическое занятие № 28. Математические функции и их использование.

Цель: Изучить различные типы адресации ячеек, математические функции и их использование.

Теория. Адресация ячеек.

В Excel существует два типа адресации ячеек (ссылки на ячейку). (**Ссылка** - это формализованное обращение к другой ячейке.) Относительный адрес показывает, как расположена та или иная ячейка относительно заданной. Например, формула содержащая адрес **A1** находится в ячейки **A3**.

ПРОИЗВЕД X ✓ = =A1*5

A	B	C
1 34		
2		
3 =A1*5	170	

Это значит, что речь в формуле идет о ячейки, которая находится через одну ячейку над ячейкой с формулой. И при копировании (например, в ячейку **B5**) или переносе адрес автоматически изменится таким образом, чтобы в новом положении он показывал на ячейку, которая расположена точно так же.

	B5	=	=B3*5
1	34		
2			
3	170		
4			
5	0		
6			

адрес показывает на ячейку, которая расположена так же через одну ячейку над ячейкой с формулой

Абсолютный адрес всегда указывает на точный адрес ячейки, вне зависимости от того, где располагается формула, использующая эту ячейку. **Внешним признаком** абсолютного адреса является наличие знака \$ перед значением координаты в адресе ячейки. Знак \$ может стоять в адресе, как перед обозначением столбца, так и перед номером строки. Например, адрес **\$A\$1** - абсолютный адрес, при любом копировании и перемещении он останется неизменным и всегда будет указывать на ячейку, находящуюся на пересечении столбца A и строки 1.

ПРОИЗВЕД			=	=\$A\$1*5
	A	B	C	
1	34			
2				
3	=\$A\$1*5			абсолютный адрес на ячейку A1

	B5	=	=\$A\$1*5
	A	B	C
1		34	
2			
3		170	
4			
5		170	
6			

При копировании формулы
абсолютный адрес на ячейку A1
сохраняется

В адресах: **\$F3** и **B\$2** сочетаются абсолютная и относительная адресация (смешанная адресация). В первом случае (**\$F3**) абсолютная адресация используется для задания столбца, а относительная - для строки, во втором случае (**B\$2**) - наоборот.

Обычно в формулах используется относительная адресация, так как в этом случае при копировании и перемещении формул полностью сохраняются взаимосвязи между ячейками таблицы. Абсолютная адресация используется при включении в формулу неизменных величин, которые всегда расположены в одних и тех же ячейках, независимо от положения ссылающихся на них формул.

Пример 1. Расчеты в таблицах

Рассчитать величину начисленных денег и величину выдаваемой премии для каждого сотрудника отдела по формулам:

Начислено = Отработано часов*Тарифная ставка.

Премия = Начислено *Процент премии.

Величина Процент премии – 10% - хранится в отдельной ячейке E1.

Формулы, по которым рассчитываются соответствующие таблицы, приведены ниже

	A	B	C	D	E
1				премия	0,1
2					0,2
3	ФИО	Тарифная ставка (руб.)	Отработано часов	Начислено	Премия
4	Петрова О.Д.	100	25	=B4*C4	=D4*\$E\$1
5	Кириллова А.О.	80	10	=B5*C5	=D5*\$E\$1
6	Никитина Т.Л.	80	15	=B6*C6	=D6*\$E\$1
7	Федотова Р.З.	110	20	=B7*C7	=D7*\$E\$1
8					

Результаты вычисления по формулам:

	A	B	C	D	E
1				премия	0,1
2					0,2
3	ФИО	Тарифная ставка (руб.)	Отработано часов	Начислено	Премия
4	Петрова О.Д.	100	25	2500	250
5	Кириллова А.О.	80	10	800	80
6	Никитина Т.Л.	80	15	1200	120
7	Федотова Р.З.	110	20	2200	220
8					

Пример 2. Абсолютные ссылки

Задание. Составьте таблицу распределения дивидендов (выплаты доходов акционерам по итогам работы) в акционерном обществе в соответствии с количеством акций каждого акционера.

- Сначала нужно определить исходные данные задачи: величину распределяемой суммы дохода (число), фамилии работников (текст), количество акций у каждого акционера (целое число). В качестве рассчитываемых данных в таблице будет суммарное значение числа всех акций (число) и сумма выплаты каждому акционеру (число).
- Запустите Excel, переименуйте Лист1 в Дивиденды, опишите на нем структуру таблицы и введите исходные данные следующим образом
- В ячейку B8 введите формулу суммы диапазона B4:B7; для этого, выделив диапазон B4:B7, щелкните кнопку Автосумма на панели инструментов Стандартная.

	A	B	C
1	Распределение дивидендов в акционерном обществе		
2	Сумма доходов		
3	Фамилия	Количество акций	Выплата
4	Иванов	5	
5	Петров	6	
6	Сидоров	8	
7	Костиц	4	
8	Итого:		

- Ведите в ячейку B2 величину распределяемого дохода, например, 10000.
- Ведите в ячейку C4 формулу расчета выплаты $=$B$2/$B$8*B4$. (Выплата каждому акционеру равна частному от всей суммы доходов акционерного общества и количества всех акций, умноженному на количество акций данного акционера.)

Обратите внимание, что адреса ячеек B2 и B8 записаны в абсолютной форме, т.е. они не изменяются при копировании формулы из ячейки C4 в C5:C7.

Примечание. Для создания абсолютной ссылки на ячейки B2, B8 нужно при вводе формулы щелкнуть на данной ссылке клавишей **F4**.

6. Задайте финансовый формат для значений данных в ячейках C4:C7, для чего, выделив этот диапазон, выберите в меню **Формат** опцию ячейки, а затем на вкладке **Число** окна **Формат ячеек** выберите вариант числового формата: **Финансовый**, после чего укажите число десятичных знаков 2, а в поле **Обозначение** выберите **p.** и щелкните кнопку «**ОК**».
7. Измените ширину столбца С, чтобы в нем правильно отображались выплаты в финансовом формате.
8. Проверьте действие таблицы, изменяя данные в ячейках B2, B4:B7.
9. Сохраните таблицу с диаграммой под именем **Дивиденды** и закройте окно программы Excel.

Упражнения:

Уровень 1

1. Создайте таблицу:

- При заполнении используйте: автозаполнение, копирование содержимого ячеек;
- Заголовки отцентрируйте, установите размер шрифта 14, полужирный;
- Задайте денежный формат для столбцов: Цена и Сумма;
- Используйте автосуммирование для нахождения итоговых значений по столбцам (ячейки строки Итого);
- Найдите значения в ячейках столбца Сумма по следующей формуле:
$$\text{Сумма} = \text{Цена} * \text{Кол-во за год}$$

№	Наименование	Ед.изм.	Цена, р.	1 кв	2 кв	3 кв	4 кв	Кол. за год	Сумма
1	Тетрадь	шт.	25	10	15	8	14		
2	Ручка	шт.	28	9	12	11	10		
3	Карандаш	шт.	15	15	10	12	16		
4	Блокнот	шт.	60	4	7	9	6		
5	Маркер	шт.	40	7	5	8	6		
Итого									

2. Постройте и заполните таблицу начисления зарплаты, если размер премии составляет 20% оклада.

№	ФИО	Оклад	Премия	Зарплата
1	Арсентьев В.А.	30000	?	?
2	Петров М.Н.	25 000	?	?
3	Романов К.Н.	20 000	?	?
4	Семенова Л.П.	15000	?	?
5	Яковлева С.Т.	10000	?	?
Итого:				?

Практическое занятие № 29. Статистические функции и их использование.

Цель: Использование стандартных функций. Мастер функций. Использование статистических функций в решении задач.

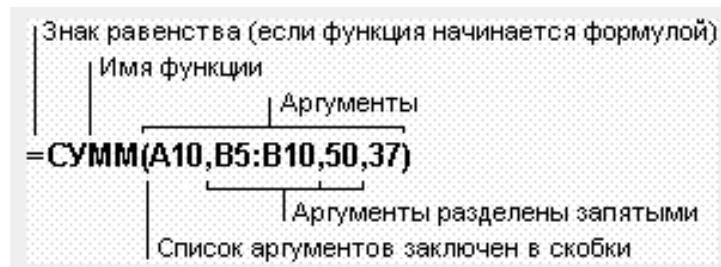
Теория Функции

Функции — заранее определенные формулы, которые выполняют вычисления по заданным величинам, называемым аргументами, и в указанном порядке. Эти функции позволяют выполнять как простые, так и сложные вычисления.

В Microsoft Excel используется более 100 функций, объединенных по категориям, например:

- **Статистические функции** позволяют выполнять статистический анализ диапазонов данных.

Функции задаются с помощью формул, которые выполняют вычисления по заданным величинам, называемым аргументами, и в указанном порядке, называемом синтаксисом. Список аргументов может состоять из чисел, текста, логических величин (ИСТИНА или ЛОЖЬ), массивов, значений ошибок (#Н/Д) или ссылок. Необходимо следить за соответствием типов аргументов. Кроме того, аргументы могут быть как константами, так и формулами. Эти формулы, в свою очередь, могут содержать другие функции.



Написание функции начинается с указания имени функции, затем вводится открывающая скобка, указываются аргументы, отделяющиеся запятыми, а затем — закрывающая скобка. Вставлять функцию в формулу рекомендуется с помощью *панели формул*. В панели формул отображается имя функции и ее аргументы, описание функции и аргументов, а также возвращаемое функцией и формулой значение.

Рис. 2. Мастер функций

При появлении сообщения **Ошибка в формуле**:

- Проверьте, одинаково ли количество открывающих и закрывающих скобок.
- Проверьте правильность использования оператора диапазона при ссылке на группу ячеек.
- Проверьте, все ли необходимые аргументы введены для функций.
- Если первый символ в имени книги или листа не является буквой, необходимо заключить имя в одинарные кавычки.
- Проверьте, в каждой ли внешней ссылке указано имя книги и полный путь к ней.
- Не изменяйте формат чисел, введенных в формулы. Например, даже если в формулу необходимо ввести 1000 р., то введите число 1000.

Ошибка #####. Ошибка появляется, когда вводимое числовое значение или результат выполнения формулы не умещается в ячейке. В этом случае можно увеличить ширину столбца путем перемещения границы, расположенной между заголовками столбцов. Кроме того, можно изменить формат числа ячейки.

Ошибка #ДЕЛ/0!. Ошибка появляется, когда в формуле делается попытка деления на ноль. Например, в качестве делителя используется ссылка на ячейку, содержащую нулевое или пустое значение (если operand является пустой ячейкой, то ее содержимое интерпретируется как ноль), или в формуле содержится явное деление на ноль.

Ошибка #Н/Д. Значение ошибки #Н/Д является сокращением термина “Неопределенные Данные”. Это значение помогает предотвратить использование ссылки на пустую ячейку. Введите в ячейки листа значение #Н/Д, если они должны содержать данные, но в настоящий момент эти данные отсутствуют. Формулы, ссылающиеся на эти ячейки, тоже будут возвращать значение #Н/Д вместо того, чтобы пытаться производить вычисления. Ошибка

может возникнуть, если не заданы один или несколько аргументов стандартной или пользовательской функции, а также задан недопустимый аргумент.

Ошибка #ИМЯ?. Ошибка #ИМЯ? появляется, когда Excel не может распознать имя, используемое в формуле. Возможная причина:

- Используемое имя было удалено или не было определено.
- Имеется ошибка в написании имени.
- Имеется ошибка в написании имени функции.
- В формулу введен текст, не заключенный в двойные кавычки.
- В ссылке на диапазон ячеек пропущен знак двоеточия (:).

Ошибка #ПУСТО!. Ошибка #ПУСТО! появляется, когда задано пересечение двух областей, которые в действительности не имеют общих ячеек.

Ошибка #ССЫЛКА!. Ошибка #ССЫЛКА! появляется, когда используется недопустимая ссылка на ячейку. Возможно, ячейки, на которые ссылаются формулы, были удалены или в эти ячейки было помещено содержимое других скопированных ячеек.

Ошибка #ЗНАЧ!. Ошибка #ЗНАЧ! появляется, когда используется недопустимый тип аргумента или операнда. Например, вместо числового или логического значения введен текст, и Microsoft Excel не может преобразовать его к нужному типу данных.

Пример: Рассчитать коэффициент усвоения задания (К).

Коэффициент усвоения задания (К) вычисляется по формуле:

$$K = \frac{\text{Средний балл за данное задание}}{\text{максимальный балл за данное задание}}$$

	A	B	C	D
1	Коэффициент усвоения задания			
2	Фамилия Имя	Задание 1	Задание 2	Задание 3
3	Петров Виталий	8	9	8
4	Семенов Алексей	10	7	8
5	Талызина Венера	9	8	9
6	Якимова Светлана	7	7	9
7	Средний балл	=СРЗНАЧ(B3:B6)	=СРЗНАЧ(C3:C6)	=СРЗНАЧ(D3:D6)
8	Максимальный балл	=МАКС(B3:B6)	=МАКС(C3:C6)	=МАКС(D3:D6)
9	Коэффициент усвоения	=B7/B8	=C7/C8	=D7/D8

Упражнения:

Уровень 1

1. Создайте таблицу. Выполните необходимые вычисления.

Месяц	2021	2022	2023	За три года
Январь	37,2	34,5	8	?
Февраль	11,4	51,3	1,2	?
Март	16,5	20,5	3,8	?
Апрель	19,5	26,9	11,9	?
Май	11,7	45,5	66,3	?
Июнь	129,1	71,5	60	?

Июль	57,1	152,9	50,6	?
Август	43,8	96,6	145,2	?
Сентябрь	85,7	74,8	79,9	?
Октябрь	86	14,5	74,9	?
Ноябрь	12,5	21	56,6	?
Декабрь	21,2	22,3	9,4	?
Суммарно	?	?	?	?
Максимум	?	?	?	?
Минимум	?	?	?	?
Среднемесячное	?	?	?	?

Уровень 2

2. Постройте таблицу: «Крупные водохранилища России», по которой будет видно: суммарная площадь водохранилищ, средний объем водохранилищ, максимальную глубину и минимальный напор водохранилищ.

Средняя глубина Камского водохранилища - 6,5 м. Площадь Горьковского водохранилища – 1400 кв.км. Объем Рыбинского водохранилища – 25 куб.км. Напор Цимлянского водохранилища – 26 м. Площадь Братского водохранилища – 5300 кв.км. Средняя глубина Куйбышевского водохранилища - 10,4 м. Объем Цимлянского водохранилища – 24 куб.км. Площадь Рыбинского водохранилища – 180 куб.км. Площадь Камского водохранилища – 1 700 кв.км. напор Куйбышевского водохранилища – 28 м. Средняя глубина Цимлянского водохранилища - 9,2 м. Напор Камского водохранилища - 21 м. Площадь Куйбышевского водохранилища – 5 000 кв.км. Напор Рыбинского водохранилища – 25 м. Средняя глубина братского водохранилища – 34 м. Объем Куйбышевского водохранилища – 52 куб.км. Напор Горьковского водохранилища – 18 м. Средняя глубина Рыбинского водохранилища – 5,5 м. Объем Камского водохранилища - 11 куб.км. Напор Братского водохранилища – 104 м. Площадь Цимлянского водохранилища – 2600 кв.км.

3. Постройте таблицу «Крупнейшие промышленные корпорации», по которой будет видно: суммарный и средний оборот всех компаний, максимальное и минимальное количество работников.

Компания «Дженирал Моторс» находится в США. Она занимается производством автомобилей. Компания в начале 90-х гг. имела оборот в 102млрд. долларов и 811000 работников. Компания «Тойота мотор» находится в Японии. Она занимается производством автомобилей. Компания в начале 90-х гг. имела оборот в 42 млрд. долларов и 84207 работников. Компания «Ройял Датч-Шелл» занимается производством нефтепродуктов. Компания в начале 90-х гг. имела оборот в 78 млрд. долларов и 133 000 работников. Компания «Тексако» находится в США. Она занимается производством нефтепродуктов. Компания в начале 90-х гг. имела 54481 работника. Компания «Эксон» находится в США. Она занимается производством нефтепродуктов. Компания в начале 90-х гг. имела оборот в 76млрд. долларов и 146000 работников. Компания «Форд Мотор» находится в США. Она занимается производством автомобилей. Компания в начале 90-х гг. имела оборот в 72млрд. долларов и 369300 работников. Компания «Интернационал бизнес мэшинс» находится в США. Она занимается производством вычислительной техники. Компания в начале 90-х гг. имела оборот в 54млрд. долларов и 403508 работников. Компания «Мобил» находится в США. Она занимается производством нефтепродуктов. Компания в начале 90-х гг. имела оборот в 52млрд. долларов. Компания «Бритиш петролиум» находится в Великобритании. Она занимается производством нефтепродуктов. Компания в начале 90-х гг. имела оборот в 45 млрд. долларов и 126020 работников. Компания «ИРИ» находится в Италии. Она занимается производством металлов. Компания в начале 90-х гг. имела оборот в 41 млрд. долларов и 422000 работников.

Практическое занятие № 30. Решение задач по специальности.

Цель: Математические функции. Способы представления данных. Виды диаграмм. Решение задач аналитической геометрии средствами Excel. Возможности визуализации различных уравнений. Построение кривых на плоскости.

Теория

Функция ОКРУГЛ

Функция ОКРУГЛ округляет число, задаваемое ее аргументом, до указанного количества десятичных разрядов и имеет следующий синтаксис: =ОКРУГЛ(число; количество_цифр)

Аргумент число может быть числом, ссылкой на ячейку, в которой содержится число, или формулой, возвращающей числовое значение. Аргумент количество цифр, который может быть любым положительным или отрицательным целым числом, определяет, сколько цифр будет округляться. Задание отрицательного аргумента количество цифр округляет до указанного количества разрядов слева от десятичной запятой, а задание аргумента количество цифр равным 0 округляет до ближайшего целого числа. Excel округляет цифры, которые меньше 5, с недостатком (вниз), а цифры, которые больше или равны 5, с избытком (вверх).

Не смешивайте функцию ОКРУГЛ с фиксированными форматами, например 0 и 0,00, которые имеются на вкладке Число окна диалога Формат ячеек. При использовании этих форматов содержимое в ячейке округляется до заданного числа десятичных разрядов, но при этом изменяется только отображение числа в ячейке, а не само значение. При выполнении вычислений Excel всегда использует хранимое, а не отображаемое значение.

Функция ПРОИЗВЕД

Функция ПРОИЗВЕД перемножает все числа, задаваемые ее аргументами, и имеет следующий синтаксис: =ПРОИЗВЕД(число1; число2;...). Функция ПРОИЗВЕД может иметь до 30 аргументов. Excel игнорирует любые пустые ячейки, текстовые и логические значения.

Функция КОРЕНЬ

Функция КОРЕНЬ возвращает положительный квадратный корень числа и имеет следующий синтаксис: =КОРЕНЬ(число)

Аргумент число должен быть положительным числом. Например, следующая функция возвращает значение 2: =КОРЕНЬ(4)

Если число отрицательное, КОРЕНЬ возвращает ошибочное значение: #ЧИСЛО!.

Функция LOG

Функция LOG возвращает логарифм положительного числа по заданному основанию. Эта функция имеет следующий синтаксис: =LOG(число; основание)

Например, следующая формула возвращает значение 2,321928095, то есть логарифм 5 по основанию 2: =LOG(5;2) Если вы не укажете аргумент основание, Excel примет его равным 10.

Функция LN

Функция LN возвращает натуральный (по основанию e) логарифм положительного числа, указанного в качестве аргумента. Эта функция имеет следующий синтаксис: =LN(число). Например, следующая формула возвращает значение 0,693147181: =LN(2)

Функция ПИ

Функция ПИ возвращает значение константы (с точностью до 14 десятичных знаков: 3,14159265358979. Эта функция имеет следующий синтаксис: =ПИ(). Функция ПИ не имеет аргументов, но мы все равно должны вводить после имени функции круглые скобки.

Обычно функция ПИ используется в других функциях или формулах.

Функции РАДИАНЫ и ГРАДУСЫ

Тригонометрические функции используют углы, выраженные в радианах, а не в градусах. Excel предоставляет две функции, РАДИАНЫ и ГРАДУСЫ, чтобы облегчить работу с тригонометрическими функциями. Вы можете преобразовать радианы в градусы, используя функцию ГРАДУСЫ, которая имеет следующий синтаксис: =ГРАДУСЫ(угол). Здесь угол — это число, представляющее собой угол, измеренный в радианах. Для преобразования градусов в радианы используется функция РАДИАНЫ которая имеет следующий синтаксис:

=РАДИАНЫ(угол). Здесь угол — это число, представляющее собой угол, измеренный в градусах. Например, следующая формула возвращает значение 180: =ГРАДУСЫ(3,1415927). В то же время следующая формула возвращает значение 3,1415927: =РАДИАНЫ(180)

Функция SIN

Функция SIN возвращает синус угла и имеет следующий синтаксис: =SIN(число) Здесь число — это угол в радианах. Например, следующая формула возвращает значение 0,997494987: =SIN(1,5)

Функция COS

Функция COS вычисляет косинус угла и имеет следующий синтаксис: =COS(число)

Здесь число — это угол в радианах. Например, следующая формула возвращает значение 0,070737202: =COS(1,5)

Функция TAN

Функция TAN вычисляет тангенс угла и имеет следующий синтаксис: =TAN(число). Здесь число — это угол в радианах. Например, следующая формула возвращает тангенс угла 1,5 радиана, равный 14,10141995: =TAN(1,5)

Пример. Использование математических функций

Задание. Создать таблицу с использованием математических функций, которая рассчитывает значения функции $y=\sin(2x/3)*\cos(x/2)$ на интервале значений x от $-\pi$ до $+2$ с шагом 0,1, вычисляет максимальное и минимальное значения функции на данном интервале области определения, а также строит график данной функции.

1. Запустите программу Excel и для получения подсказки о синтаксисе тригонометрических функций в окне справки Excel на вкладке **Содержание** выберите раздел *Справка по функциям*, тема *Математические функции*. Для просмотра информации щелкните по ссылкам SIN, COS и ПИ. Для возврата к предыдущему окну справки щелкните кнопку «Назад».

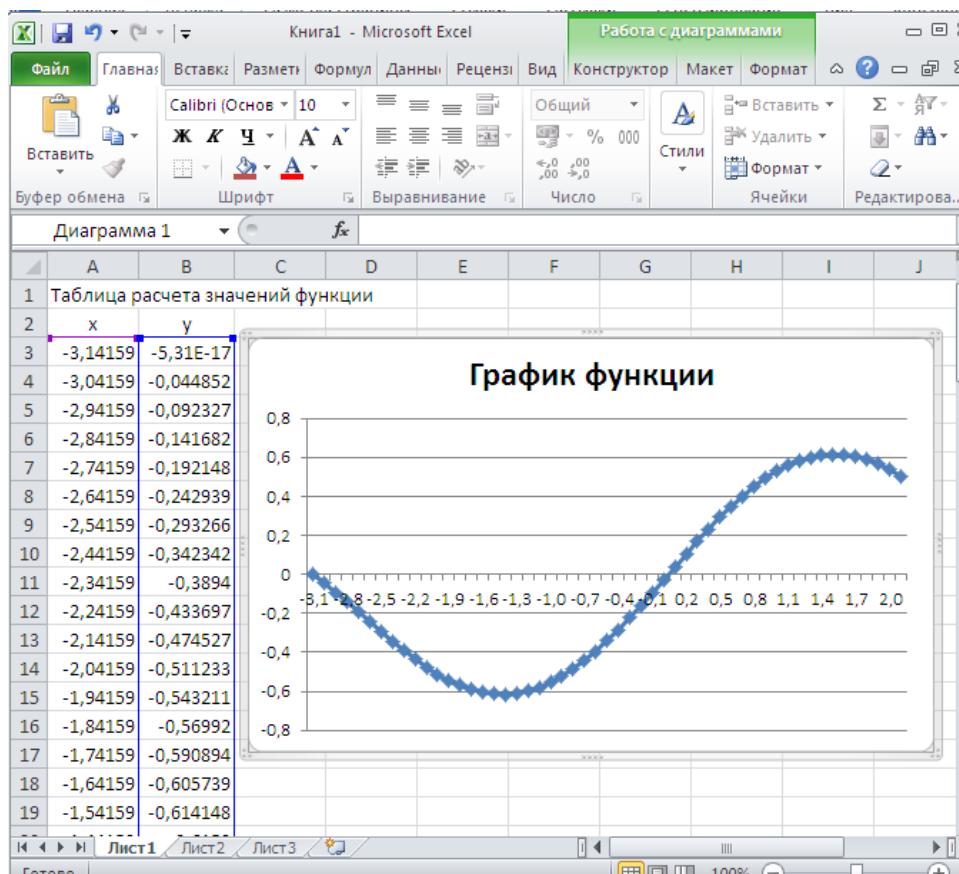


Рис. 3. Таблица расчета значений функции $y=\sin(2x/3)*\cos(x/2)$

- После просмотра справки закройте окно справки любым из стандартных способов.
2. В ячейку A1 введите заголовок таблицы «Таблица значений функции $y=\sin(2x/3)*\cos(x/2)$. В ячейку A2 введите x , в ячейку B2 введите y .
 3. Задайте в ячейке A3 формулу =-ПИ() для ввода начального значения x . В ячейке A4 задайте формулу =A3+0,1 для вычисления следующего значения x , изменяющегося с шагом 0,1. Скопируйте формулу из A4 в диапазон (A4:A97).
 3. В ячейку B3 введите формулу расчета значения функции =SIN(2*A3/3)*COS(A3/2). Затем скопируйте формулу из ячейки B3 в диапазон (B3:B97).
 4. В ячейку C2 введите формулу определения минимума функции =МИН(B2:B97), а в ячейку C3 - формулу определения максимума функции =МАКС(B2:B97). В результате получится таблица, фрагмент которой показан на рис. 1.
 5. Для построения графика функции выделите диапазон ячеек (B2:B97) и, щелкнув кнопку , Выберите тип диаграмм **График**, определите параметры диаграммы: заголовки, подписи данных, положение легенды, линии сетки и т.д.
 6. Сохраните полученную таблицу.

Практические задания:

1. В сельскохозяйственном кооперативе работают 5 сезонных рабочих. Норма сбора овощей составляет N кг. Оплата труда производится по количеству собранных овощей: k рублей за 1 кг. Составить таблицу, содержащую сведения о количестве собранных овощей каждым рабочим и об оплате труда каждого рабочего. Известно, что 1-й рабочий собрал в 3 раза больше нормы; 2-й на 50 кг меньше первого; 3-й - в 1,5 раза больше нормы; 4-й на 75 кг больше третьего; 5-й - на 10 кг больше первого.
Постройте круговую диаграмму, демонстрирующую распределение оплаты труда каждому работнику.
2. Оформите решение квадратного уравнения $ax^2+bx+c=0$, используя ЭТ. Напишите заголовок. Выполните решение для следующих наборов коэффициентов:
 - a) $a = 2, b = -3, c = -2;$
 - b) $a = 2, b = -4, c = -6;$
 - c) $a = -3, b = 1, c = 1.$
3. Найдите значение выражения для

a) $x = 1, y = 1;$	b) $x = 1, y = 0;$	c) $x = 2, y = 1.$
--------------------	--------------------	--------------------

$$1) \frac{x^2 - \frac{7}{12}y}{18y + x^2}; \quad 2) \frac{40y^3 + \frac{4}{9}x}{6x^2 - 18xy}; \quad 3) \frac{30\left(x^3 - \frac{5}{6}y\right)}{5(x^3 - 6y)};$$

Практическое занятие № 31. Математические модели в экономике.

Цель: Создание документа, содержащего элементы графического анализа в виде диаграмм и графиков, содержит операции:

- ✓ Изменение параметров (ширины) строк и столбцов;
- ✓ Работа со списками;
- ✓ Использование абсолютных ссылок;
- ✓ Построение диаграмм;
- ✓ Редактирование диаграмм;
- ✓ Построение смешанных диаграмм и графиков;
- ✓ Создание комментариев.

Например, требуется создать документ о росте доходов фирмы за год.

	A	B	C	D	
1					
2	Рост доходов за год				
3					
4	Месяц	Доход	Прирост за месяц	Рост доходов	
5	январь	800р.	0%	0%	
6	февраль	830р.	4%	4%	
7	март	820р.	-1%	3%	
8	апрель	900р.	10%	13%	
9	май	950р.	6%	19%	
10	июнь	980р.	3%	23%	
11	июль	1 000р.	2%	25%	
12	август	920р.	-8%	15%	
13	сентябрь	990р.	8%	24%	
14	октябрь	1 200р.	21%	50%	
15	ноябрь	1 240р.	3%	55%	
16	декабрь	1 300р.	5%	63%	
17					

Перечислим действия, которые требуется выполнить самостоятельно:

1. Создать новый файл в Excel с именем Доход.
2. Переименовать Лист 1 на Рост доходов.
3. Определить границы листа формата А4.
4. Во второй строке написать заголовок «Рост доходов за год».
5. В четвертой строке сформировать «шапку» документа из четырех столбцов: разместить надписи столбцов с переносом в несколько строк и отцентрировать их по высоте и ширине ячеек

Заполнение столбца доходы.

Выделить ячейки B5 – B16 и назначить им финансовый рублевый формат целого типа.

Заполнить эти ячейки соответствующими значениями из документа..

Заполнение столбца прирост за месяц

Прирост за месяц – это процент прироста дохода по сравнению с предыдущим месяцем.

За январь прирост будет составлять 0%.

Активизируйте ячейку C6 и составьте формулу для вычисления прироста за месяц в процентном соотношении.

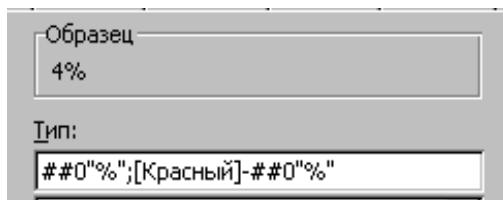
Скопируйте эту формулу для всех последующих месяцев.

= =(B6-B5)/B5*100	
B	C
Доход	Прирост за месяц
800р.	0%
830р.	3,75

Требование

Значения округлить до целых, написать знак процента (%) и отрицательные значения отобразить красным цветом.

Реализация указанного требования выполняется составлением формата ячеек, например, следующим образом:



типа: ##0%"';[Красный]-##0%"'

Назначьте этот формат для всех ячеек интервала C5-C16

Абсолютная адресация

Рост доходов – это процент прироста дохода по сравнению с началом года (январем). За январь рост доходов будет составлять 0%.

Особенность при составлении формулы для столбца Рост доходов заключается в том, что процент роста вычисляется во всех ячейках относительно одного значения - января. Поэтому, чтобы при копировании адрес ячейки B5 (январь) не смешался, необходимо использовать абсолютную адресацию.

Для составления формулы с абсолютной адресацией требуется:

- установить курсор в ячейку D6 и составить формулу вычисления роста доходов относительно начала года (января) в процентах;
- скопировать формулу на остальные ячейки этого столбца;
- назначить такой же формат ячеек, как и для столбца Прирост, за месяц.

D6			
A	B	C	D
Месяц	Доход	Прирост за месяц	Рост доходов
январь	800р.	0%	0%
февраль	830р.	4%	3,75

В результате выполнения всех действий получится документ, изображенный в начале статьи.

Построение диаграммы

Диаграмма - графическое представление данных.

Значение из ячеек рабочего листа, или точки данных, отображаются на экране в виде строк, линий, столбцов, секторов или других компонентов диаграммы.

Точки данных группируются в ряды данных, которые отличаются друг от друга цветом, формой, заполнением.

Построение диаграммы «Изменение величины дохода по месяцам»

Для изменения размеров области диаграммы установить курсор мыши на требуемую засечку и, не отжимая левой кнопки мыши, потянуть в нужном направлении.

Редактирование диаграммы.

Автоматически сформированная диаграмма явно требует редактирования.

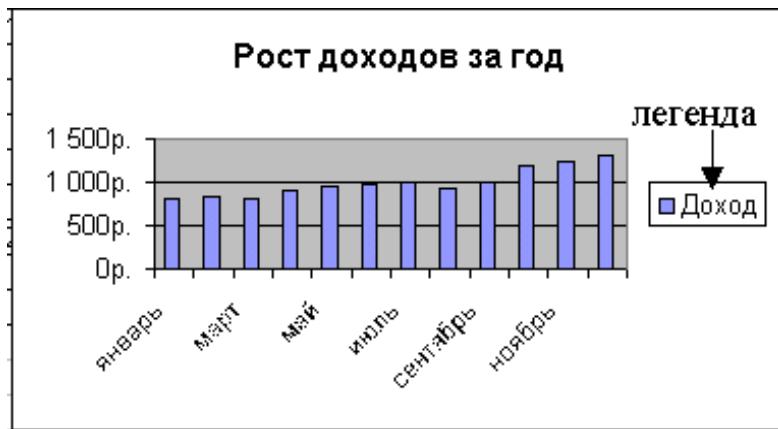
Правило: чтобы отредактировать элемент диаграммы, достаточно дважды щелкнуть по нему мышкой.

Удаление элемента диаграммы.

На диаграмме, состоящей из одного ряда данных, область легенды является излишней.

Для удаления области легенды необходимо:

- выделить Легенду, нажав мышкой один раз (Окно Легенды должно быть выделено засечками);
- нажать клавишу Del.



Надписи осей.

Для формирования надписи для Y требуется:

- установить курсор на область Диаграммы и нажать правую кнопку мыши;
- через меню Параметры диаграммы открыть закладку Заголовки и для оси Y задать надпись «Доход».

Редактирование надписи оси.

Чтобы все наименования месяцев разместились в области диаграммы, необходимо развернуть их вертикально. Для изменения надписи оси надо:

- установить курсор на любое наименование месяца на диаграмме (например, январь) и дважды щелкнуть левой кнопкой мыши;
- через меню Формат оси перейти на закладку Выравнивание и установить Ориентацию на 90 градусов;
- установить курсор на красную точку, которой заканчивается слово «надпись», и, не отжимая левой кнопки мыши, перетащить на требуемый угол поворота.

Изменение шага значений оси.

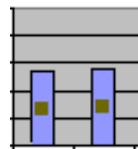
Шаг значений по оси Y равен 500. Требуется уменьшить шаг до 300. Для этого необходимо:

- установить курсор на любое из значений оси Y и дважды щелкнуть по нему левой кнопкой мыши;
- через меню Формат оси выбрать закладку Шкала и в опции Цена основных делений установить значение 300;
- перейти на закладку Шрифт и установить размер цифр – 10.

Изменение формата рядов данных.

Для того чтобы ряду данных (столбцам) придать объемный вид и сменить цветовую палитру, требуется:

- нажать курсором по центру одного из столбцов данных, чтобы во всех столбцах диаграммы в центре появились точки;
- через меню Формат выбрать Формат ряда данных или щелкнуть по выделенным столбцам правой кнопкой мыши;
- перейти на закладку Вид и установить параметры: Граница – с тенью, Заливка – голубая.



Подписи данных.

Диаграмма станет более наглядной, если каждый столбец будет содержать отображаемые числовые значения. Для этого необходимо:

- выделить весь ряд данных (во всех столбцах диаграммы должны быть точки);
- через меню Формат выбрать Формат ряда данных, затем перейти на закладку Подписи данных и отметить опцию Значение;
- для оформления подписей данных щелкнуть по ним левой кнопкой мыши, при этом обратить внимание, чтобы были выделены все подписи, и через меню Формат выбрать Формат подписей данных;

- перейти на закладку Шрифт и установить параметры: шрифт – Arial Суг, начертание – полужирный, высота символов – 8;
- перейти на закладку Выравнивание и установить параметры: Ориентация - на 90 градусов, Положение подписи – центр.

Изменение ширины столбцов диаграммы.

Чтобы подписи данных размещались в столбцах, необходимо увеличить ширину столбцов:

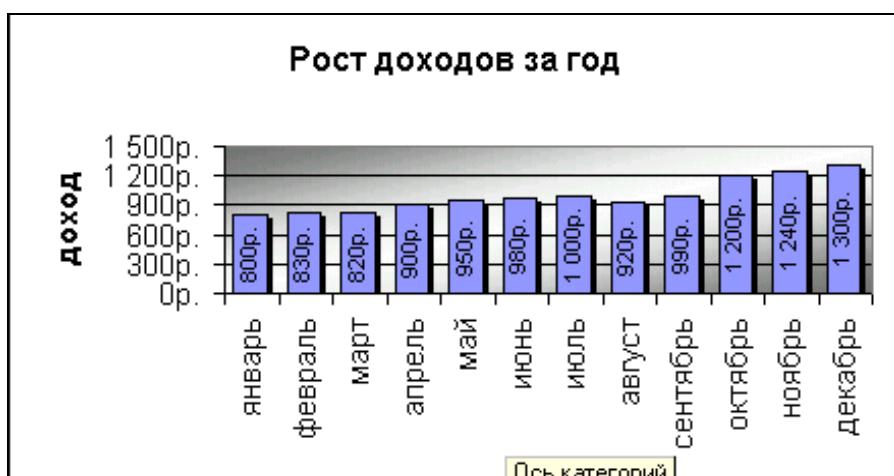
- выделить весь ряд данных (точки должны быть во всех столбцах диаграммы);
- через меню Формат – Выделенный ряд перейти на закладку Параметры и установить ширину зазора – 50.

Изменение фона построения диаграммы.

Следующая задача – изменить серый фон, на котором построена диаграмма. Для этого надо:

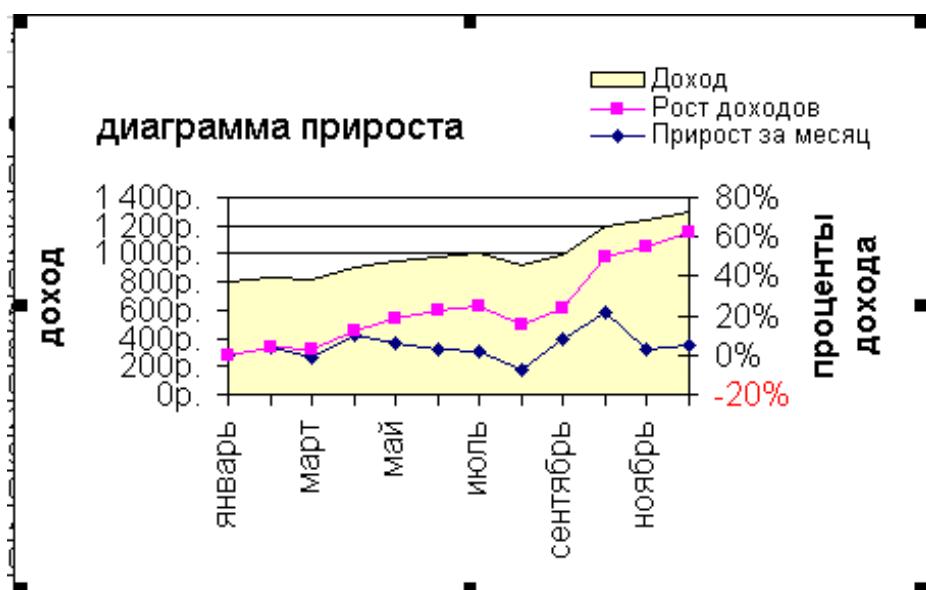
- выделить область построения диаграммы;
- через меню Формат – Выделенная область построения нажать на пункт Способ заливки, затем на закладке Градиентная указать: один цвет; цвет 1 – белый; тип штриховки – диагональная 1; тип – 4.

В результате диаграмма принимает следующий вид:



Практическое задание:

1. Построить диаграмму прироста



Практическое занятие № 32. Моделирование систем.

Цель: изучить математический аппарат имитационного моделирования

Практические задания

1. Определить относительную частоту выпадения орла или решки.
2. Бросание кубика. Найти статистическую вероятность для каждого значения выпадения очков 1, 2, 3, 4, 5, 6.
3. Из 25 экзаменационных билетов по информатике студент успел подготовить 11 первых и 8 последних билетов. Какова вероятность того, что на экзамене ему достанется билет, который он подготовил?

Практическое занятие № 33. Работа с готовой базой данных

Цель: Изучить Интерфейс. Понятие и назначение базы данных (далее – БД). Классификация БД. Системы управления БД (СУБД).

Теория. Общие сведения

Access - это система управления базами данных (СУБД). Под системой управления понимается комплекс программ, который позволяет не только хранить большие массивы данных в определенном формате, но и обрабатывать их, представляя в удобном для пользователей виде. Access дает возможность также автоматизировать часто выполняемые операции (например, расчет заработной платы, учет материальных ценностей и т.п.). С помощью Access можно не только разрабатывать удобные формы ввода и просмотра данных, но и составлять сложные отчеты.

Access является приложением Windows, а поскольку и Windows и Access разработаны одной фирмой (Microsoft), они очень хорошо взаимодействуют друг с другом. СУБД Access работает под управлением Windows; таким образом, все преимущества Windows доступны в Access, например, вы можете вырезать, копировать и вставлять данные из любого приложения Windows в приложение Access и наоборот.

Access - это реляционная СУБД. Это означает, что с ее помощью можно работать одновременно с несколькими таблицами базы данных. Применение реляционной СУБД помогает упростить структуру данных и таким образом облегчить выполнение работы. Таблицу Access можно связать с данными, хранящимися на другом компьютере или на сервере, а также использовать таблицу, созданную в СУБД Paradox или Dbase. Данные Access очень просто комбинировать с данными Excel.

Структура таблицы и типы данных

Все составляющие базы данных, такие, как таблицы, отчеты, запросы, формы и объекты, в Access хранятся в едином дисковом файле. Основным структурным компонентом базы данных является таблица. В таблицах хранятся вводимые нами данные. Внешне каждая таблица Access похожа на таблицы, с которыми мы привыкли работать на бумаге, - она состоит из столбцов, называемых *полями*, и строк, называемых *записями*. Каждая запись таблицы содержит всю необходимую информацию об отдельном элементе базы данных. Например, запись о преподавателе может содержать фамилию, имя, отчество, дату рождения, должность и т.п.

При разработке структуры таблицы, прежде всего, необходимо определить названия полей, из которых она должна состоять, типы полей и их размеры. Каждому полю таблицы присваивается уникальное имя, которое не может содержать более 64 символов. Имя желательно делать таким, чтобы функция поля узнавалась по его имени. Далее надо решить, данные какого типа будут содержаться в каждом поле. В Access можно выбирать любые из основных типов данных. Один из этих типов данных должен быть присвоен каждому полю. Значение типа поля может быть задано только в режиме конструктора. Ниже представлены типы данных Access и их описание.

Тип данных	Описание
Текстовый (Значение по умолчанию)	Текст или числа, не требующие проведения расчетов, например номера телефонов (до 255 знаков)
Числовой	Числовые данные различных форматов, используемые для проведения расчетов
Дата/время	Для хранения информации о дате и времени с 100 по 9999 год включительно
Денежный	Денежные значения и числовые данные, используемые в математических расчетах, проводящихся с точностью до 15 знаков в целой и до 4 знаков в дробной части
Поле МЕМО	Для хранения комментариев; до 65535 символов
Счетчик	Специальное числовое поле, в котором Access автоматически присваивает уникальный порядковый номер каждой записи. Значения полей типа счетчика обновлять нельзя
Логический	Может иметь только одно из двух возможных значений (True/False, Да/Нет)
Поле объекта OLE	Объект (например, электронная таблица Microsoft Excel, документ Microsoft Word, рисунок, звукозапись или другие данные в двоичном формате), связанный или внедренный в таблицу Access
Гиперссылка	Строка, состоящая из букв и цифр и представляющая адрес гиперссылки. Адрес гиперссылки может состоять максимум из трех частей: текст, выводимый в поле или в элементе управления; путь к файлу (в формате пути UNC) или к странице (адрес URL). Чтобы вставить адрес гиперссылки в поле или в элемент управления, выполните команду Вставка, Гиперссылка
Мастер подстановок	Создает поле, в котором предлагается выбор значений из списка или из поля со списком, содержащего набор постоянных значений или значений из другой таблицы. Это в действительности не тип поля, а способ хранения поля

В Access существует четыре способа создания пустой таблицы:

- использование мастера баз данных для создания всей базы данных, содержащей все требуемые отчеты, таблицы и формы, за одну операцию. Мастер баз данных создает новую базу данных, его нельзя использовать для добавления новых таблиц, форм, отчетов в уже существующую базу данных;
- мастер таблиц позволяет выбрать поля для данной таблицы из множества определенных ранее таблиц, таких, как деловые контакты, список личного имущества или рецепты;
- ввод данных непосредственно в пустую таблицу в режиме таблицы. При сохранении новой таблицы в Access данные анализируются, и каждому полю присваивается необходимый тип данных и формат;
- определение всех параметров макета таблицы в режиме конструктора.

Независимо от метода, примененного для создания таблицы, всегда имеется возможность использовать режим конструктора для дальнейшего изменения макета таблицы, например для добавления новых полей, установки значений по умолчанию или для создания масок ввода. Однако только четвертый метод позволяет сразу задать ту структуру таблицы, которая вам нужна, и поэтому далее рассмотрим именно этот метод.

Практическое задание

Задание 1. Создание базы данных.

1. Создайте новую базу данных.
2. Создайте таблицу базы данных.
3. Определите поля таблицы в соответствии с табл. 1.
4. Сохраните созданную таблицу.

Таблица 1. Таблица данных *Преподаватели*

Имя поля	Тип данных	Размер поля
Код преподавателя	Счетчик	
Фамилия	Текстовый	15
Имя	Текстовый	15
Отчество	Текстовый	15
Дата рождения	Дата/время	Краткий
Должность	Текстовый	9
Дисциплина	Текстовый	11
Телефон	Текстовый	9
Зарплата	Денежный	

ТЕХНОЛОГИЯ РАБОТЫ

Для создания новой базы данных:

- загрузите Access, в появившемся окне выберите пункт **Новая база данных**;
- в окне "Файл новой базы данных" задайте имя вашей базы (пункт **Имя Файла**) и выберите папку (пункт **Папка**), где ваша база данных будет находиться. По умолчанию Access предлагает вам имя базы db1, а тип файла - *Базы данные Access*. Имя задайте *Преподаватели*, а тип файла оставьте прежним, так как другие типы файлов нужны в специальных случаях;
- щелкните по кнопке <Создать>.

Для создания таблицы базы данных:

- в окне базы данных выберите вкладку *Таблицы*, а затем щелкните по кнопке <Создать>;
- в окне "Новая таблица" выберите пункт **Конструктор** и щелкните по кнопке <OK>. В результате проделанных операций открывается окно таблицы в режиме конструктора, в котором следует определить поля таблицы.

Для определения полей таблицы:

- введите в строку столбца Имя поля имя первого поля *Код преподавателя*;
- в строке столбца "Тип данных" щелкните по кнопке списка и выберите тип данных **Счетчик**. Поля вкладки *Общие* оставьте такими, как предлагает Access.

Примечание. Заполнение строки столбца "Описание" необязательно и обычно используется для внесения дополнительных сведений о поле.

Для определения всех остальных полей таблицы базы данных *Преподаватели* в соответствии с табл.1 выполните действия, аналогичные указанным выше.

Внимание! Обратите внимание на вкладку *Общие* в нижней части экрана. Советуем изменить данные в пункте **Размер поля**, а остальные пункты оставить по умолчанию (их функции рассмотрим далее). Например, для текстового типа данных Access предлагает по умолчанию длину 50 символов. Но вряд ли поле "Фамилия" будет содержать более 15 символов, хотя лучше точно подсчитать, сколько символов в самой длинной фамилии. Не бойтесь ошибиться - в дальнейшем можно скорректировать длину поля. Для числового типа Access предлагает *Длинное целое*, но ваши данные могут быть либо небольшие целые числа (в диапазоне от -32768 до 32767) - тогда надо выбрать *Целое*, либо дробные числа - тогда надо выбрать *С плавающей точкой..*. Для выбора необходимого параметра надо щелкнуть по полю, а затем нажать появившуюся кнопку списка и выбрать необходимые данные. В результате ваша таблица будет иметь более компактный вид, а объем базы данных уменьшится.

4. Для сохранения таблицы:

- выберите пункт меню **Файл, Сохранить**;

- в диалоговом окне "Сохранение" введите имя таблицы *Преподаватели*,
- щелкните по кнопке <OK>.

Примечание. В результате щелчка по кнопке <OK> Access предложит вам задать ключевое поле (поле первичного ключа), т.е. поле, однозначно идентифицирующее каждую запись. Для однотабличной базы данных это не столь актуально, как для многотабличной, поэтому щелкните по кнопке <Нет>.

ЗАДАНИЕ 2

Заполнение базы данных.

1. Введите ограничения на данные, вводимые в поле "Должность"; должны вводиться только слова *Профессор*, *Доцент* или *Ассистент*.
2. Задайте текст сообщения об ошибке, который будет появляться на экране при вводе неправильных данных в поле "Должность".
3. Задайте значение по умолчанию для поля "Должность" в виде слова *Доцент*.
4. Введите ограничения на данные в поле <Код>; эти данные не должны повторяться.
5. Заполните таблицу данными в соответствии с табл.2 и проверьте реакцию системы на ввод неправильных данных в поле "Должность".
6. Измените ширину каждого поля таблицы в соответствии с шириной данных.
7. Произведите поиск в таблице преподавателя Миронова.
8. Произведите сортировку данных в поле "Дата рождения" по убыванию,
9. Произведите фильтрацию данных по полям "Должность" и "Дисциплина".

Просмотрите созданную таблицу, как она будет выглядеть на листе бумаги при печати.

Таблица 2

Код	Фамилия	Имя	Отчество	Дата рожд.	Должность	Дисциплина	Телефон	Зарплата
1	Истомин	Ремир	Евгеньевич	23.10.74	Доцент	Информатика	10-44-68	22890р.
2	Миронов	Павел	Юрьевич	25.07.60	Профессор	Экономика	12-21-40	50200р.
3	Гришин	Евгений	Сергеевич	05.12.77	Доцент	Математика	60-23-65	25760р.
4	Сергеева	Ольга	Ивановна	12.02.82	Ассистент	Математика	34-85-69	15450р.
5	Емец	Татьяна	Ивановна	16.02.71	Доцент	Экономика	66-75-33	22890р.
6	Игнатьева	Татьяна	Павловна	30.05.76	Доцент	Информатика	10-36-98	23790р.
7	Миронов	Алексей	Николаевич	30.07.78	Доцент	Физика	66-75-33	23890р.

ТЕХНОЛОГИЯ РАБОТЫ

Для задания условия на значение для вводимых данных:

- войдите в режим *Конструктор* для проектируемой таблицы. Если вы находитесь в окне базы данных, то выберите вкладку *Таблицы* и щелкните по кнопке <Конструктор>. Если вы находитесь в режиме таблицы, то щелкните по кнопке на панели инструментов или выполните команду **Вид, Конструктор**;

- в верхней части окна щелкните по полю "Должность";
- в нижней части окна щелкните по строке параметра *Условие на значение*;
- щелкните по кнопке для определения условий на значение при помощи построителя выражений;
- в появившемся окне напишите слово *Профессор*, затем щелкните по кнопке (эта кнопка выполняет функцию ИЛИ), напишите *Доцент*, снова щелкните по этой же кнопке, напишите *Ассистент* и щелкните по кнопке <OK>. Таким образом, вы ввели условие, при котором в поле "Должность" могут вводиться только указанные значения.

2. В строке *Сообщение об ошибке* введите предложение "Такой должности нет, правильно введите данные".

3. В строке *Значение по умолчанию* введите слово "Доцент".

4. Введите ограничения на данные в поле "Код". Здесь ограничения надо вводить не совсем обычным способом. Дело в том, что коды преподавателей не должны повторяться, а также должна быть обеспечена возможность их изменения (из-за последнего условия в этом поле

нельзя использовать тип данных *Счетчик*, в котором данные не повторяются). Для выполнения второго условия пришлось задать в поле "Код" тип данных *Числовой*, а для выполнения первого условия сделайте следующее:

- щелкните по строке параметра *Индексированное поле*;

Примечание. Индекс - это средство Access, ускоряющее поиск и сортировку данных в таблице. Ключевое поле (поле первичного ключа) таблицы индексируется автоматически. Не допускается создание индексов для полей типа *MEMO* и *Гиперссылка vim* полей объектов *OLE*. Свойство *Индексированное поле* определяет индекс, создаваемый по одному полю. Индексированное поле может содержать как уникальные, так и повторяющиеся значения. Допускается создание произвольного количества индексов.

- выберите в списке пункт **Да (совпадения не допускаются)**;

перейдите в режим **Таблица**, щелкнув по кнопке на панели инструментов или выполнив команду **Вид, Режим таблицы**. На вопрос о сохранении таблицы щелкните по кнопке <Да>.

5. Введите данные в таблицу в соответствии с табл. 2. Попробуйте в поле <Должность> любой записи ввести слово *Лаборант*. Посмотрите, что получилось. На экране должно появиться сообщение: "Такой должности нет, правильно введите данные". Введите правильное слово.

6. Для изменения ширины каждого поля таблицы в соответствии с шириной данных:

- щелкните в любой строке поля "Код";
- выполните команду **Формат, Ширина столбца**;
- в появившемся окне щелкните по кнопке <По ширине данных>. Ширина поля изменится;
- проделайте эту операцию с остальными полями.

7. Для поиска в таблице преподавателя Миронова:

- переведите курсор в первую строку поля "Фамилия";
- выполните команду **Правка, Найти**;
- в появившейся строке параметра *Образец* введите *Миронов*;
- в строке параметра *Просмотр* должно быть слово *Все* (имеется в виду искать по всем записям);
 - в строке параметра *Совпадение* выберите из списка *С любой частью поля*;
 - в строке параметра *Только в текущем поле* установите флажок (должна стоять галочка);
 - щелкните по кнопке <Найти>. Курсор перейдет на вторую запись и выделит слово *Миронов*;
- щелкните по кнопке <Найти далее>. Курсор перейдет на седьмую запись и также выделит слово *Миронов*;
- щелкните по кнопке <Закрыть> для выхода из режима поиска.

8. Для сортировки данных в поле "Дата рождения" по убыванию:

- щелкните по любой записи поля "Дата рождения";
- щелкните по кнопке на панели управления или выполните команду **Записи, Сортировка, Сортировка по убыванию**. Все данные в таблице будут отсортированы в соответствии с убыванием значений в поле "Дата рождения".
 - Для фильтрации данных по полям "Должность" и "Дисциплина":
 - щелкните по записи *Доцент* поля "Должность";
 - щелкните по кнопке или выполните команду **Записи, Фильтр, Фильтр по выделенному**. В таблице останутся только записи о преподавателях - доцентах;
 - щелкните по записи *Информатика* поля "Дисциплина";
 - щелкните по кнопке или выполните команду **Записи, Фильтр, Фильтр по выделенному**. В таблице останутся только записи о преподавателях - доцентах кафедры информатики;
 - для отмены фильтрации щелкните по кнопке на панели инструментов или выполните команду **Записи, Удалить фильтр**. В таблице появятся все данные.

Для просмотра созданной таблицы:

- щелкните по кнопке или выполните команду **Файл, Предварительный, просмотр**. Вы увидите таблицу как бы на листе бумаги;
- закройте окно просмотра.

Примечание. Если вы захотите изменить поля или ориентацию таблицы на листе бумаги, выполните команду **Файл, Параметры страницы**. В открывшемся окне можете изменять указанные параметры.

Практическое занятие № 34 Разработка многотабличной базы данных

Цель: Проектирование многотабличной базы данных. Установка связей между таблицами.

Теория.

Организация данных

Слово "реляционная" происходит от английского *relation* - отношение. *Отношение* - тематическое понятие, но в терминологии моделей данных отношения удобно изображать в виде таблицы. При этом строки таблицы соответствуют кортежам отношения, а столбцы - атрибутам. Ключом называют любую функцию от атрибутов кортежа, которая может быть использована для идентификации кортежа. Такая функция может быть значением одного, из атрибутов (простой ключ), задаваться алгебраическим выражением, включающим значения нескольких атрибутов (составной ключ). Это означает, что данные в строках каждого из столбцов составного ключа могут повторяться, но комбинация данных каждой строки этих столбцов является уникальной. Например, в таблице Студенты есть столбцы *Фамилии* и *Год рождения*. В каждом из столбцов есть некоторые повторяющиеся данные, т.е. одинаковые фамилии и одинаковые годы рождения. Но если студенты, имеющие одинаковые фамилии, имеют разные годы рождения, то эти столбцы можно использовать в качестве составного ключа. Как правило, ключ является уникальным, т.е. каждый кортеж определяется значением ключа однозначно, но иногда используют и неуникальные ключи (ключи с повторениями). В локализованной (руссифицированной) версии Access вводится термин *ключевое поле*, которое можно трактовать как *первичный ключ*.

В Access можно выделить три типа ключевых полей: простой ключ, составной ключ и внешний ключ.

Одно из важнейших достоинств реляционных баз данных состоит в том, что вы можете хранить логически сгруппированные данные в разных таблицах и задавать связи между ними, объединяя их в единую базу. Для задания связи таблицы должны иметь поля с одинаковыми именами или хотя бы с одинаковыми форматами данных. Связь между таблицами устанавливает отношения между совпадающими значениями в этих полях. Такая организация данных позволяет уменьшить избыточность хранимых данных, упрощает их ввод и организацию запросов и отчетов. Поясним это на примере. Допустим, вам в базе надо хранить, данные о студентах (фамилия, изучаемая дисциплина) и преподавателях (фамилия, номер кафедры, ученая степень, преподаваемая дисциплина). Если хранить данные в одной таблице, то в строке с фамилией студента, изучающего конкретную дисциплину, будут храниться все атрибуты преподавателя, читающего эту дисциплину. Это же огромная избыточность данных. А если хранить данные о студенте в одной таблице, о преподавателе - в другой и установить связь между полями "Читаемая дисциплина" - "Изучаемая дисциплина" (фактически это одинаковые поля), то избыточность хранимых данных многократно уменьшится без ущерба для логической организации информации.

В Access можно задать три вида связей между таблицами; *Один-ко-многим*, *Многие-ко-многим* и *Один-к-одному*.

Связь *Один-ко-многим* - наиболее часто используемый тип связи между таблицами. В такой связи каждой записи в таблице А может соответствовать несколько записей в таблице В (поля с этими записями называют *внешними ключами*), а запись в таблице В не может иметь более одной соответствующей ей записи в таблице А.

При связи *Многие-ко-многим* одной записи в таблице А может соответствовать несколько записей в таблице В, а одной записи в таблице В - несколько записей в таблице А. Такая схема

реализуется только с помощью третьей (связующей) таблицы, ключ которой состоит по крайней мере из двух полей, одно из которых является общим с таблицей А, а другое - общим с таблицей В.

При связи *Один-к-одному* запись в таблице А может иметь не более одной связанной записи в таблице В и наоборот. Этот тип связи используют не очень часто, поскольку такие данные могут быть помещены в одну таблицу. Связь с отношением *Один-к-одному* применяют для разделения очень широких таблиц, для отделения части таблицы в целях ее защиты, а также для сохранения сведений, относящихся к подмножеству записей в главной таблице.

Тип создаваемой связи зависит от полей, для которых определяется связь:

- связь *Один-ко-многим* создается в том случае, когда только одно из полей является ключевым или имеет уникальный индекс, т.е. значения в нем не повторяются;
- связь *Один-к-одному* создается в том случае, когда оба связываемых поля являются ключевыми или имеют уникальные индексы;
- связь *Многие-ко-многим* фактически представляет две связи типа *один-ко-многим* через третью таблицу, ключ которой состоит, по крайней мере, из двух полей, общих для двух других таблиц.

Целостность данных

Целостность данных означает систему правил, используемых в СУБД Access для поддержания связей между записями в связанных таблицах (таблиц, объединенных с помощью связи), а также обеспечивает защиту от случайного удаления или изменения связанных данных. Контролировать целостность данных можно, если выполнены следующие условия:

- связанное поле (поле, посредством которого осуществляется связь) одной таблицы является ключевым полем или имеет уникальный индекс;
- связанные поля имеют один тип данных. Здесь существует исключение. Поле счетчика может быть связано с числовым полем, если оно имеет тип *Длинное целое*,
- обе таблицы принадлежат одной базе данных Access. Если таблицы являются связанными, то они должны быть таблицами Access. Для установки целостности данных база данных, в которой находятся таблицы, должна быть открыта. Для связанных таблиц из баз данных других форматов установить целостность данных невозможно.

ЗАДАНИЕ 1

Создание инфологической и логической моделей базы данных.

1. Разработайте информационно-логическую модель реляционной базы данных.

2. Разработайте логическую модель реляционной базы данных

ТЕХНОЛОГИЯ РАБОТЫ

1. Перед разработкой информационно-логической модели реляционной базы данных рассмотрим, из каких информационных объектов должна состоять эта база данных. Можно выделить три объекта, которые не будут обладать избыточностью, - *Студенты*, *Дисциплины* и *Преподаватели*. Представим состав реквизитов этих объектов в виде "название объекта (перечень реквизитов)": *Студенты* (код студента, фамилия, имя, отчество, номер группы, дата рождения, стипендия, оценки). *Дисциплины* (код дисциплины, название дисциплины), *Преподаватели* (код преподавателя, фамилия, имя, отчество, дата рождения, телефон, заработка плата).

Рассмотрим связь между объектами *Студенты* и *Дисциплины*. Студент изучает несколько дисциплин, что соответствует многозначной связи и отражено на рис. 3.9 двойной стрелкой. Понятно, что каждая дисциплина изучается множеством студентов. Это тоже многозначная связь, обозначаемая двойной стрелкой (связь "один" обозначена одинарной стрелкой). Таким образом, связь между объектами *Студенты* и *Дисциплины* - *Многие-ко-многим* ($M : N$).

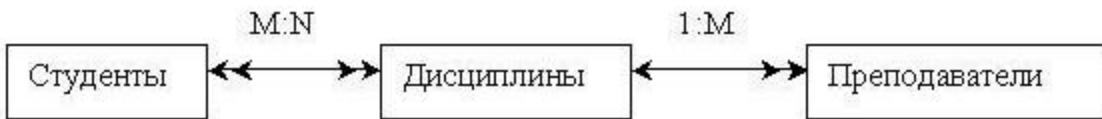


Рис Типы связей между объектами *Студенты*, *Дисциплины* и *Преподаватели*.

Множественные связи усложняют управление базой данных, например, в СУБД Access при множественных связях нельзя использовать механизм каскадного обновления. Поэтому использовать такие связи нежелательно и нужно строить реляционную модель, не содержащую связей типа *Многие-ко-многим*. В Access для контроля целостности данных с возможностью каскадного обновления и удаления данных необходимо создать вспомогательный объект связи, который состоит из ключевых реквизитов связываемых объектов и который может быть дополнен описательными реквизитами. В нашем случае таким новым объектом для связи служит объект *Оценки*, реквизитами которого являются код студента, код дисциплины и оценка. Каждый студент имеет оценки по нескольким дисциплинам, поэтому связь между объектами *Студенты* и *Оценки* будет *Один-ко-многим* (1:M). Каждую дисциплину сдает множество студентов, поэтому связь между объектами *Дисциплины* и *Оценки* также будет *Один-ко-многим* (1:M). В результате получаем информационно-логическую модель базы данных, приведенную на рисунке.

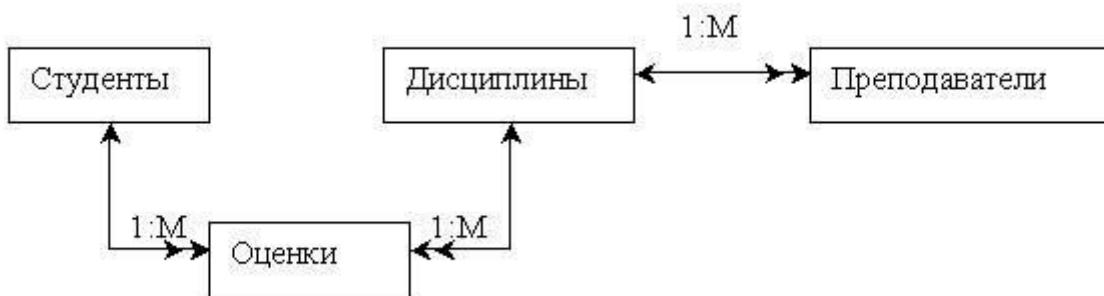


Рис. Информационно-логическая модель реляционной базы данных

2. В реляционной базе данных в качестве объектов рассматриваются отношения, которые можно представить в виде таблиц. Таблицы между собой связываются посредством общих полей, т.е. одинаковых по форматам и, как правило, по названию, имеющихся в обеих таблицах. Рассмотрим, какие общие поля надо ввести в таблицы для обеспечения связности данных. В таблицах *Студенты* и *Оценки* таким полем будет "Код студента", в таблицах *Дисциплины* и *Оценки* - "Код дисциплины", в таблицах *Преподаватели* и *Дисциплины* - "Код дисциплины". Выбор цифровых кодов вместо фамилий или названий дисциплин обусловлен меньшим объемом информации в таких полях: например, число "2". по количеству символов значительно меньше слова "математика". В соответствии с этим логическая модель базы данных представлена на рисунке где жирными буквами выделены ключевые поля.

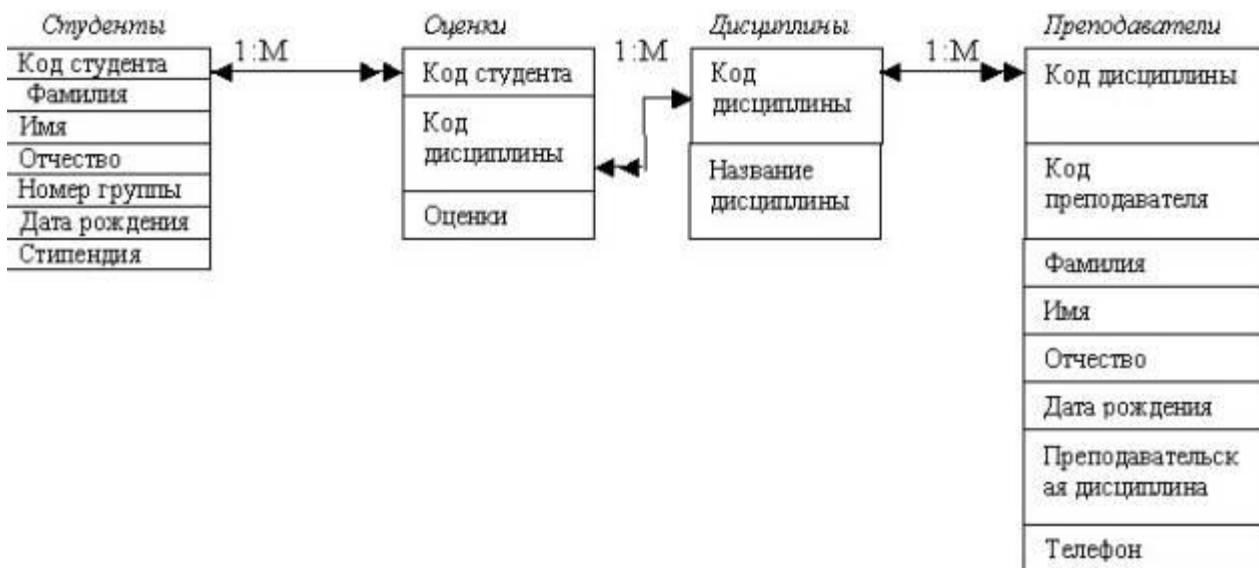


Рис. Логическая модель базы данных

ЗАДАНИЕ 2

Создание реляционной базы данных.

1. Создайте базу данных *Деканат*.
2. Создайте структуру таблицы *Студенты*.
3. Создайте структуру таблицы *Дисциплины*.
4. Измените структуру таблицы *Преподаватели*.
5. Создайте структуру таблицы *Оценки*.
6. Разработайте схему данных, т.е. создайте связи между таблицами.

ТЕХНОЛОГИЯ РАБОТЫ

1. Создайте базу данных *Деканат*, выполнив следующие действия:
 - загрузите Access, в появившемся окне выберите пункт *Новая база данных*, затем щелкните по кнопке <OK>;
 - в окне <Файл новой базы данных> задайте имя (пункт *Имя файла*) и выберите папку (пункт *Папка*), где ваша база будет находиться. По умолчанию Access предлагает имя базы *db1*, а тип файла - *Базы данных Access*. Имя задайте *Деканат*, а тип файла оставьте прежним, так как другие типы файлов нужны в специальных случаях;
 - щелкните по кнопке <Создать>
2. Создайте структуру таблицы *Студенты*. Для этого:
 - в окне базы данных выберите вкладку *Таблицы*, а затем щелкните по кнопке <Создать>;
 - в окне "Новая таблица" выберите пункт *Конструктор* и щелкните по кнопке <OK>. В результате проделанных операций открывается окно таблицы в режиме конструктора, в котором следует определить поля таблицы;
 - определите поля таблицы в соответствии с таблицей:

Имя поля	Тип данных	Размер поля
Код студента	Числовой	Целое
Фамилия	Текстовый	15
Имя	Текстовый	12
Отчество	Текстовый	15
Номер группы	Числовой	Целое
Телефон	Текстовый	9
Стипендия	Логический	Да/Нет

- в качестве ключевого поля задайте "Код студента". Для этого щелкните по полю "Код студента" и по кнопке на панели инструментов или выполните команду **Правка, Ключевое поле**;

- закройте таблицу, задав ей имя *Студенты*.

Примечание. Заполнять таблицу данными пока не требуется, это будет сделано в режиме формы.

3. Создайте структуру таблицы *Дисциплины* аналогично п. 2 в соответствии с таблицей.

Имя поля	Тип данных	Размер поля
Код дисциплины	Числовой	Целое
Название дисциплины	Текстовый	30

В качестве ключевого поля задайте "Код дисциплины". Заполняться эта таблица будет также в режиме формы.

4. Структура таблицы *Преподаватели* уже создана в работе 1 и заполнена данными, этому для работы используйте эту таблицу с одним лишь изменением - в структуру таблицы надо добавить поле "Код дисциплины" и заполнить его в соответствии с данными табл.

5. Создайте структуру таблицы *Оценки* аналогично п. 2 в соответствии с табл.

Имя поля	Тип данных	Размер поля
Код студента	Числовой	Целое
Код дисциплины	Числовой	Целое
Оценки	Числовой	Байт

В этой таблице задавать ключевое поле не надо, так как данные во всех полях могут повторяться. Эта таблица, аналогично предыдущим, будет заполняться в режиме формы.

6. Разработайте схему данных, т.е. создайте связи между таблицами. Для этого:

- щелкните по кнопке на панели инструментов или выполните команду **Сервис, Схема данных**. На экране появится окно "Схема данных";
- щелкните по кнопке на панели инструментов или выполните команду **Связи, Добавить таблицу**;
- в появившемся окне будет выделено название одной таблицы. Щелкните по кнопке <Добавить>;
- переведите выделение на имя следующей таблицы и щелкните по кнопке <Добавить>.

Аналогично добавьте оставшиеся две таблицы;

- закройте окно, щелкнув по кнопке <Закрыть>;
- создайте связь между таблицами *Дисциплины* и *Оценки*. Для этого подведите курсор мыши к полю "Код дисциплины" в таблице *Дисциплины* щелкните левой кнопкой мыши и, не отпуская ее, перетащите курсор на поле "Код дисциплины" в таблицу *Оценки*, а затем отпустите кнопку мыши. На экране откроется окно "Связи";
- установите флажок ("галочку") в свойстве *Обеспечение целостности данных*, щелкнув по нему;
- установите флажок в свойстве *Каскадное обновление связанных полей* и *Каскадное удаление связанных записей*;

Примечание. Задание каскадного обновления связанных полей и каскадного удаления связанных записей позволит вам отредактировать записи только в таблице *Дисциплины*, а в таблице *Оценки* эти действия будут со связанными записями выполняться автоматически. Например, если вы удалите из таблицы *Дисциплины* один предмет, то в таблице *Оценки* удаляются все строки, связанные с этим предметом.

- щелкните по кнопке <Создать>. Связь будет создана;
- аналогично создайте связи между полем "Код дисциплины" в таблице *Дисциплины* и полем "Код дисциплины" в таблице *Преподаватели*, а также между полем "Код студента" в таблице *Студенты* и полем "Код студента" в таблице *Оценки*. Результат представлен на рис. 12;
- закройте окно схемы данных, ответив *ДА* на вопрос о сохранении макета.

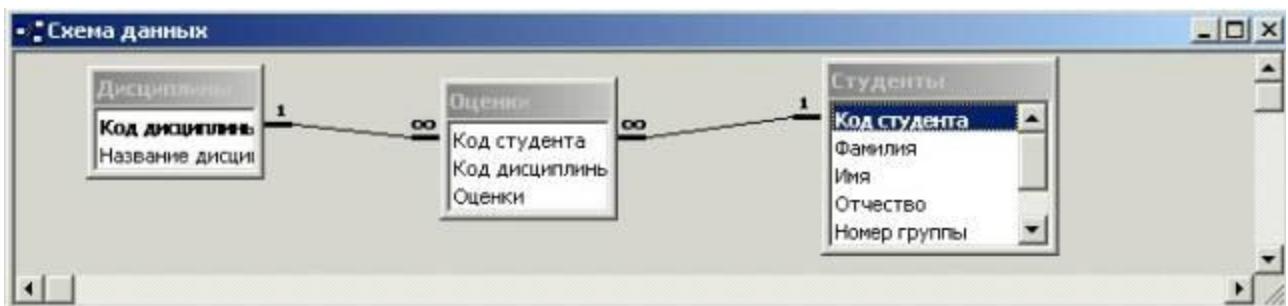


Рис. Структура таблицы Студенты

Практическое занятие № 35. Запросы к многотабличной базе данных.

Цель: Изучить способы выбора данных из таблиц с помощью запросов, создавать запросы на выборку, на обновление и удаление,

Теория

Формирование запросов на выборку

Запросы являются мощным средством обработки данных, хранимых в таблицах Access . С помощью запросов можно просматривать, анализировать и изменять данные из нескольких таблиц. Они также используются в качестве источника данных для форм и отчетов. Запросы позволяют вычислять итоговые значения и выводить их в компактном формате, подобном формату электронной таблицы, а также выполнять, вычисления над группами записей.

Запросы можно создавать самостоятельно и с помощью мастеров. Мастера запросов автоматически выполняют основные действия в зависимости от ответов пользователя на поставленные вопросы. Самостоятельно разработать запросы можно в режиме конструктора.

В Access можно создавать следующие типы запросов:

- запрос на выборку;
- запрос с параметрами;
- перекрестный запрос;
- запрос на изменение (запрос на удаление, обновление и добавление записей на создание таблицы);
- запросы SQL (запросы на объединение, запросы к серверу, управляющие запросы, подчиненные запросы)

Запрос на выборку используется наиболее часто. При его выполнении данные, удовлетворяющие условиям отбора, выбираются из одной или из нескольких таблиц и выводятся в определенном порядке. Например, можно вывести на экран данные о фамилиях доцентов, стаж которых более 15 лет (на основе таблицы Преподаватели).

Примечание. Простые запросы на выборку практически не отличаются от фильтров. Более того, фильтры можно сохранять как запросы.

Можно также использовать запрос на выборку, чтобы сгруппировать записи для вычисления сумм, средних значений, пересчета и других действий. Например, используя запрос на выборку, можно получить данные о среднем стаже доцентов и профессоров (на основе таблицы Преподаватели).

Запрос с параметрами - это запрос, при выполнении которого в его диалоговом окне пользователю выдается приглашение ввести данные, на основе которых будет выполняться запрос. Например, часто требуются данные о том, какие дисциплины ведут преподаватели. Чтобы не создавать отдельные запросы по каждому преподавателю, можно создать один запрос с параметрами, где в качестве параметра будет использоваться фамилия преподавателя. При каждом вызове этого запроса вам будет предложено ввести фамилию преподавателя, а затем на экран будут выведены все поля, которые вы указали в запросе, например фамилия, имя, отчество преподавателя и читаемая им дисциплина.

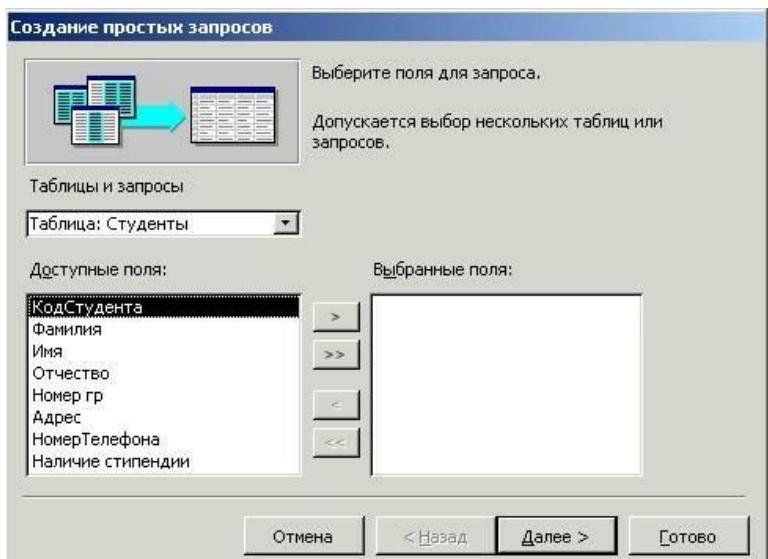


Рис. Окно выбора вариантов построения запросов

Для создания нового запроса надо в окне базы данных выбрать вкладку *Запросы* и щелкнуть по кнопке <Создать>. Откроется окно "Новый запрос", вид которого представлен на рис. 2.8. В окне вы должны выбрать один из пяти пунктов: Конструктор, Простой запрос, Перекрестный запрос, Повторяющиеся записи. Записи без подчиненных. *Конструктор* позволит вам самостоятельно создать любой тип запроса, но этот режим рекомендуется пользователям, уже имеющим некоторый опыт создания запросов. *Простой запрос* позволит создать с помощью *Мастера* запрос на выборку из определенных полей таблиц или других запросов. (Это наилучший способ создания запроса для начинающих пользователей.) Что такое *перекрестный запрос*, рассматривалось выше. При выборе пункта *Повторяющиеся записи* будет создан запрос на поиск повторяющихся записей в простой таблице или в запросе, а при выборе пункта *Записи без подчиненных* - запрос на поиск записей, которым не соответствует ни одна запись в подчиненной таблице. Такой запрос используется для многотабличных баз данных.

У вас может возникнуть вопрос: как создавать запросы с параметрами и запросы на изменение, если при создании запроса они явно не указаны? Следует отметить, что основой для всех этих запросов является запрос на выборку, т.е. сначала необходимо определить набор данных, с которым хотите работать. Затем для созданного запроса на выборку надо перейти в режим конструктора. Задание параметров производится в строке *Условия отбора* для соответствующих полей. Подробнее это будет рассмотрено ниже при выполнении задания. Для доступа к запросам на изменение надо открыть пункт меню *Запрос* - в открывшемся списке вы увидите все виды запросов на изменение.

При выполнении запроса на выборку Access извлекает записи из таблиц и формирует результирующий набор данных. Он выглядит, как таблица, хотя и не является ею. Результирующий набор данных является *динамическим* (или *виртуальным*) набором записей и не хранится в базе данных.

После закрытия запроса результирующий набор данных этого запроса прекращает свое существование. Хотя сам по себе динамический набор данных больше не существует, помните, что данные, которые в нем содержались, остаются в базовых таблицах.

При сохранении запроса остается только структура запроса - перечень таблиц, список полей, порядок сортировки, ограничения на записи, тип запроса и т.д.

При сохранении в базе данных запрос, по сравнению с результирующим набором данных, имеет ряд преимуществ:

- на физическом носителе информации (обычно это жесткий диск) требуется меньший объем пространства;
- запрос может использовать обновленные версии любых записей, измененных со времени последнего запуска запроса.

При каждом выполнении запрос обращается к базовым таблицам и снова создает результирующий набор данных. Поскольку сам по себе результирующий набор данных не сохраняется, запрос автоматически отображает любые изменения, произошедшие в базовых таблицах с момента последнего запуска этого запроса (даже в реальном времени в многопользовательской среде).

Для сохранения запроса следует выполнить следующие действия. Выполните команду **Файл**, **Сохранить** или щелкните по кнопке <Сохранить> на панели инструментов. Если вы впервые сохраняете запрос, введите новое имя запроса в диалоговом окне "Сохранение".

ЗАДАНИЕ

Формирование запросов на выборку.

1. На основе таблицы *Преподаватели* создайте простой запрос на выборку, в котором должны отображаться фамилии, имена, отчества преподавателей и их должность.
2. Данные запроса отсортируйте по должностям.
3. Сохраните запрос.
4. Создайте запрос на выборку с параметром, в котором должны отображаться фамилии, имена, отчества преподавателей и преподаваемые ими дисциплины, а в качестве параметра задайте фамилию преподавателя и выполните этот запрос для преподавателя *Гришина*.

ТЕХНОЛОГИЯ РАБОТЫ

1. Для создания простого запроса:

- в окне базы данных откройте вкладку *Запросы*;
- в открывшемся окне щелкните по кнопке <Создать>;
- из появившихся пунктов окна "Новый запрос" выберите **Простой запрос** и щелкните по кнопке <OK>;
- в появившемся окне в строке *Таблицы/запросы* выберите таблицу *Преподаватели* (если других таблиц или запросов не было создано, она будет одна в открывающемся списке);
 - в окне "Доступные поля" переведите выделение на параметр *Фамилия*,
 - щелкните по кнопке. Слово *Фамилия* перейдет в окно "Выбранные поля";
 - аналогично в окно "Выбранные поля" переведите поля "Имя", "Отчество", "Должность" (порядок важен - в таком порядке данные и будут выводиться);
- щелкните по кнопке <Далее>;
- в строке параметра *Задайте имя запроса* введите новое имя *Должности преподавателей*;
- щелкните по кнопке <Готово>. На экране появится таблица с результатами запроса.

2. Для сортировки данных:

- щелкните в любой строке поля "Должность";
- отсортируйте данные по убыванию. Для этого щелкните по кнопке  на панели инструментов или выполните команду **Записи, Сортировка, Сортировка по убыванию**.

3. Для сохранения запроса:

- щелкните по кнопке  или выполните команду **Файл, Сохранить**;
 - закройте окно запроса.
- ### 4. Для создания запроса на выборку с параметром:
- создайте запрос на выборку для следующих полей таблицы *Преподаватели*: "Фамилия", "Имя", "Отчество", "Преподаваемая дисциплина". Запрос создавайте аналогично тому, как это делалось в п.1;
 - задайте имя запросу *Преподаваемые дисциплины*;
 - щелкните по кнопке <Готово>. На экране появится таблица с результатами запроса;
 - перейдите в режиме конструктора, щелкнув по кнопке или выполнив команду **Вид, Конструктор**;
 - в строке параметра *Условия отбора* для поля "Фамилия" введите фразу (скобки тоже вводить): [Ведите фамилию преподавателя]

- выполните запрос, щелкнув по кнопке  на панели инструментов или выполнив команду **Запрос, Запуск**;

Примечание. Вышеописанным способом запрос выполняется только в режиме конструктора. Для того чтобы выполнить запрос из другого режима, надо открыть вкладку *Запросы*, выделить требуемый запрос и щелкнуть по кнопке <Открыть>.

• в появившемся окне введите фамилию *Гришин* и щелкните по кнопке <OK>. На экране появится таблица с данными о преподавателе *Гришине* - его имя, отчество и преподаваемая им дисциплина;

- сохраните запрос;
- закройте окно запроса.

Практическое занятие № 36. Управление данными с помощью языка SQL.

Цель: рассмотреть управление данными с помощью языка SQL.

Практическое задание

На основе таблицы *Преподаватели* создайте отчет с группированием данных по должностям.

ТЕХНОЛОГИЯ РАБОТЫ

Для создания отчета:

- откройте вкладку *Отчеты* и щелкните по кнопке <Создать>;
- в открывшемся окне выберите пункт *Мастер отчетов*;
- щелкните по значку раскрывающегося списка в нижней части окна;
- выберите из появившегося списка таблицу *Преподаватели*;
- щелкните по кнопке <OK>, В появившемся окне выберите поля, которые будут присутствовать в отчете. В данном примере присутствовать будут все поля из таблицы, поэтому щелкните по кнопке >>;
- щелкните по кнопке <Далее>;
- в появившемся окне присутствует перечень полей. Переведите выделение на поле "Должность";
 - щелкните по кнопке. Таким образом вы задаете группировку данных по должности;
 - щелкните по кнопке <Далее>;
 - параметры появившегося окна оставим без изменений, поэтому щелкните по кнопке <Далее>;
 - в появившемся окне выберите стиль оформления отчета;
 - щелкните по кнопке <Далее>;
 - в появившемся окне введите название отчета *Преподаватели*;
 - щелкните по кнопке <Готово>. На экране появится сформированный отчет;
 - просмотрите, а затем закройте отчет.

ЗАДАНИЕ

Создание форм для ввода данных в таблицы.

1. Создайте форму *Студенты*.
2. Заполните данными таблицу *Студенты* посредством формы *Студенты*.
3. Создайте форму *Дисциплины*.
4. Заполните данными таблицу *Дисциплины* посредством формы *Дисциплины*.
5. Создайте форму *Оценки*.
6. Заполните данными таблицу *Оценки* посредством формы *Оценки*.

ТЕХНОЛОГИЯ РАБОТЫ

1. Для создания формы *Студенты*:

- откройте вкладку *Формы*;
- щелкните по кнопке <Создать>;
- в открывающемся списке выберите таблицу *Студенты*,
- выберите пункт Автоформа: ленточная;
- щелкните по кнопке <OK>. Форма для ввода данных создана;

Примечание. Если вас не удовлетворяет макет, вы можете перейти в режим конструктора и изменить макет, передвигая и изменения размеры элементов - заголовков полей и ячеек для ввода данных. Достаточно щелкнуть по элементу - он выделяется прямоугольной рамкой, и вы можете изменять размеры и двигать элемент. Если вы хотите изменить другие параметры элемента, надо, по выделенному элементу щелкнуть правой клавишей мыши, и откроется окно свойств элемента. В силу ограниченности объема раздела описать все свойства нет возможности, но их можно изучить самостоятельно по справочной системе, а многие свойства понятны уже из своего названия.

2. Заполните данными, приведенными в таблицу *Студенты* посредством формы.

Таблица

Код студента	Фамилия	Имя	Отчество	Номер группы	Телефон	Стипендия
1	Арбузов	Николай	Николаевич	151	64-45-78	Да
2	Киршин	Петр	Валерьевич	151	25-45-89	Да
3	Кривинский	Сергей	Николаевич	151	45-55-52	Нет
4	Крылова	Елена	Петровна	151	64-56-88	Да
5	Кульчий	Григорий	Викторович	151	25-45-12	Да
6	Патрикеев	Олег	Борисович	152	56-52-56	Нет
7	Перлов	Кирилл	Николаевич	152	55-55-57	Нет
8	Соколова	Наталия	Петровна	152	20-45-32	Нет
9	Степанская	Ольга	Витальевна	152	45-78-23	Да
10	Тимофеев	Сергей	Трофимович	152	56-23-89	Да

Примечание. Переход между ячейками лучше выполнять клавишей <Tab> либо мышью. Существуют и другие варианты перехода по строкам или полям с помощью различных клавиш и их комбинаций. Обычно их используют опытные пользователи, не любящие работать с мышью.

Закройте форму, задав ей имя *Студенты*.

3. Создайте форму *Дисциплины* аналогично п.1.

4. Заполните данными, таблицу *Дисциплины* посредством, формы и закройте форму, задав ей имя *Дисциплины*.

Код дисциплины	Название
1	Информатика
2	Математика
3	Физика
4	Экономика

5. Создайте форму Дисциплины аналогично п. 1.

6. Заполните данными таблицу Оценки. Посредством формы закройте форму, задав ей имя Оценки.

Таблица Оценки

Код студента	Код дисциплины	Оценки	Код студента	Код дисциплины	Оценки
1	1	4	6	1	5
1	2	5	6	2	4
1	3	4	6	3	5
1	4	4	6	4	4
2	1	5	7	1	4
2	2	5	7	2	3
2	3	4	7	3	4
2	4	4	7	4	3
3	1	3	8	1	3
3	2	5	8	2	5
3	3	4	8	3	5
3	4	3	8	4	4
4	1	4	9	1	4
4	2	4	9	2	4
4	3	5	9	3	4
4	4	4	9	4	4
5	1	5	10	1	5
5	2	5	10	2	5
5	3	5	10	3	5
5	4	5	10	4	5

Практическое занятие № 37. Создание текстовой веб-страницы.

Цель: Изучить основы гипертекстовой разметки документов

Теория

Создание Web-сайтов реализуется с использованием языка разметки гипертекстовых документов HTML. Технология HTML состоит в том, что в обычный текстовый документ вставляются управляющие символы (тэги) и в результате мы получаем Web-страницу. Браузер при загрузке Web-страницы представляет ее на экране в том виде, который задается тэгами.

Основными достоинствами HTML-документов являются:

- малый информационный объем;
- возможность просмотра на персональных компьютерах, оснащенных различными операционными системами.

Для создания Web-страниц используются простейшие текстовые редакторы, которые не включают в создаваемый документ управляющие символы форматирования текста. В качестве такого редактора в Windows можно использовать стандартное приложение Блокнот.

Рассмотрим, как создаются Web-сайты, на примере разработки тематического сайта «Компьютер». Сначала необходимо разработать проект сайта, то есть определить, сколько Web-страниц будет входить в сайт, какова будет их тематика и как они будут связаны между собой.

Пусть наш сайт кроме титульной страницы «Компьютер» будет содержать:

- страницу «Программы», содержащую классификацию программного обеспечения;
- страницу «Словарь», содержащую словарь компьютерных терминов;
- страницу «Комплектующие» содержащую таблицу с основными характеристиками процессора;
- страницу «Анкета», содержащую анкету для посетителей сайта.

Практическое задание

1. Открыть окно текстового редактора Блокнот.

Вид Web-страницы задается тэгами, которые заключаются в угловые скобки. Тэги могут быть одиночными или парными, для которых обязательно наличие открывающего и

закрывающего тегов (такая пара тэгов называется контейнером). Закрывающий тэг содержит прямой слэш (/) перед обозначением. Тэги могут записываться как прописными, так и строчными буквами.

HTML- код страницы помещается внутрь контейнера <HTML> </HTML> . Без этих тэгов браузер не в состоянии определить формат документа и правильно его интерпретировать. Web-страница разделяется на две логические части: заголовок и содержание.

Заголовок Web-страницы заключается в контейнере<HEAD> </HEAD> и содержит название документа и справочную информацию о странице (например, тип кодировки), которая используется браузером для ее правильного отображения.

Название Web-страницы содержится в контейнере <TITLE></TITLE> и отображается в строке заголовка браузера при просмотре страницы. Назовем нашу Web-страницу «Компьютер».

```
<HEAD>
<TITLE> Компьютер</TITLE>
</HEAD>
```

Основное содержание страницы помещается в контейнер<BODY></BODY> и может включать текст, таблицы, бегущие строки, ссылки на графические изображения и звуковые файлы и так далее. Поместим для начала на страницу текст «Все о компьютере»:

```
<BODY>
Все о компьютере
</BODY>
```

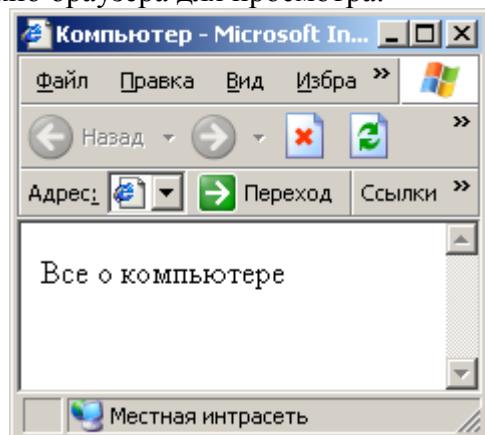
Созданную Web-страницу необходимо сохранить в виде файла. Принято сохранять титульный файл сайта, то есть тот, который первый загружается в браузер, под именем index.htm. В качестве расширения файла Web-страницы можно также использовать html.

Рекомендуется создать для размещения сайта специальную папку и сохранять все файлы разрабатываемого сайта в этой папке.

Необходимо различать имя файла index.htm, то есть имя, под которым Web-страница хранится в файловой системе, и собственно имя Web-страницы «Компьютер», которая высвечивается в верхней строке окна браузера и в первую очередь анализируется поисковыми системами. Имя Web-страницы должно в максимальной степени соответствовать ее содержанию.

2. В окне приложения Блокнот ввести HTML -код Web- страницы. Сохранить файл под именем index.htm в папке сайта. Загрузить этот файл в окно браузера для просмотра.

```
<HTML>
<HEAD>
<TITLE> Компьютер</TITLE>
</HEAD>
<BODY>
Все о компьютере
</BODY>
</HTML>
```



Форматирование текста. Пока страница выглядит не слишком привлекательно. Мелкий шрифт и черный текст на белом фоне почти не обращают на себя внимания. С помощью HTML -тэгов можно задать различные параметры форматирования текста.

Размер шрифта для имеющихся в тексте заголовков задается тэгами от <H1>, (самый крупный) до <H6> (самый мелкий). Заголовок страницы целесообразно выделить самым крупным шрифтом.

```
<H1> Все о компьютере </H1>
```

Некоторые тэги имеют атрибуты, которые являются именами свойств и могут принимать определенные значения. Так, заголовок по умолчанию выровнен по левому краю страницы, однако принято заголовок размещать по центру. Задать тип выравнивания заголовка для тэга заголовка позволяет атрибут ALIGN, которому требуется присвоить определенное значение. Выравнивание по правой границе окна реализуется с помощью записи ALIGN="right", а по центру — ALIGN="center".

```
<H1 ALIGN="center"> Все о компьютере </H1>
```

С помощью тэга FONT и его атрибутов можно задать параметры форматирования шрифта любого фрагмента текста. Атрибут FACE позволяет задать гарнитуру шрифта (например, FACE="Arial"), атрибут SIZE — размер шрифта, (например, SIZE=4), атрибут COLOR — цвет шрифта. (например, COLOR ="blue").

Значение атрибута COLOR можно задать либо названием цвета (например "red", "green", "blue"), либо его шестнадцатеричным значением. Шестнадцатеричное представление цвета использует RGB- формат "#RRGGBB", где две первые шестнадцатеричные цифры задают интенсивность красного (red), две следующие — интенсивность зеленого (green) и две последние — интенсивность синего (blue) цветов. Минимальная интенсивность цвета задается шестнадцатеричным числом 00, а максимальная — FF. Легко догадаться, что синему цвету будет соответствовать значение "#0000FF".

Таким образом, задать синий цвет заголовка можно с помощью тэга FONT с атрибутом COLOR:

```
<FONT COLOR="blue">  
<H1 ALIGN="center"> Все о компьютере </H1>  
</FONT>
```

Заголовок целесообразно отделить от остального содержания страницы горизонтальной линией с помощью одиночного тэга<HR>.

Разделение текста на абзацы производится с помощью контейнера<P></P>. При просмотре в браузере абзацы отделяются друг от друга интервалами. Для каждого абзаца можно задать определенный тип выравнивания.

На титульной странице обычно размещается текст, кратко описывающий содержание сайта. Поместим на титульную страницу текст, разбитый на абзацы с различным выравниванием:

```
<P ALIGN= "left"> На этом сайте вы сможете получить различную информацию о  
компьютере, его программном обеспечении и ценах на компьютерные комплектующие.</P>
```

```
<P ALIGN= "right"> Терминологический словарь познакомит вас с компьютерными  
терминами, а также вы сможете заполнить анкету.</P>
```

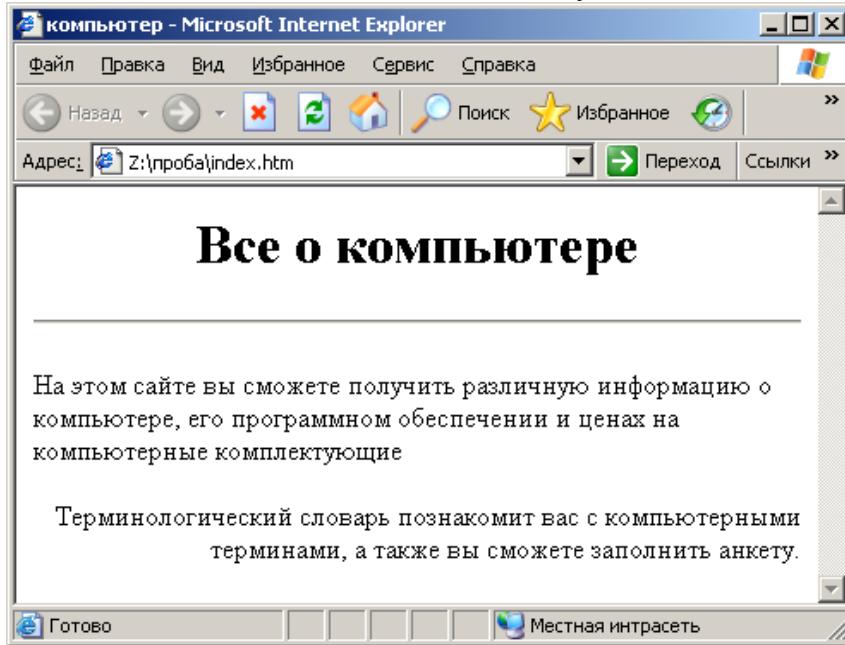
Таким образом, если вставить в текст определенную последовательность тэгов, то мы получим Web-страницу, содержащую отцентрированный заголовок синего цвета, выводимый крупным шрифтом и отделенный горизонтальной линией от остального текста.

3. В окне приложения Блокнот в контейнер<BODY> вставить последовательность тэгов и просмотреть результат в браузере.

```
<FONT COLOR="blue">  
<H1 ALIGN="center"> Все о компьютере </H1>  
</FONT>  
<HR>
```

<P ALIGN="left"> На этом сайте вы сможете получить различную информацию о компьютере, его программном обеспечении и ценах на компьютерные комплектующие</P>

<P ALIGN="right"> Терминологический словарь познакомит вас с компьютерными терминами, а также вы сможете заполнить анкету.</P>



Практическое занятие № 38. Создание веб-страницы, включающей мультимедийные объекты (рисунки, звуковые данные, видео).

Цель: Создание веб-страницы, включающей мультимедийные объекты (рисунки, звуковые данные, видео).

Теория

Вставка изображений. На Web-страницы обычно помещают изображения, чтобы сделать их визуально более привлекательными. На Web-страницах могут размещаться графические файлы трех форматов GIF, JPG, PNG. Если рисунок сохранен в другом формате, то его необходимо предварительно преобразовать в один из вышеуказанных форматов с помощью графического редактора. Для этих целей можно использовать редактор Photo Editor, который входит в пакет Microsoft Office.

Практическое задание

На титульной странице создаваемого сайта уместно разместить изображение того объекта, которому посвящен сайт (в нашем случае — компьютера). Многочисленные фотографии компьютеров можно найти, например, на сайтах производителей и продавцов компьютерной техники.

4. В папке учебные файлы найти изображение компьютера и сохранить его в файле с именем computer.gif в каталоге сайта.

Для вставки изображения используется тэг с атрибутом SRC, который указывает на место хранения файла на локальном компьютере или в Интернете. Если графический файл находится на локальном компьютере в том же каталоге, что и файл Web-страницы, то в качестве значения атрибута SRC достаточно указать только имя файла.

Если файл находится в другом каталоге на данном локальном компьютере, то значением атрибута должно быть полное имя файла.

Если файл находится на удаленном сервере в Интернете, то должен быть указан URL - адрес этого файла.

Иллюстрации на Web-страницах стали неотъемлемым элементом дизайна. Однако пользователи иногда в целях экономии времени отключают в браузере загрузку графических

изображений и читают только тексты. Поэтому, чтобы не терялся смысл и функциональность страницы, вместо рисунка должен выводиться поясняющий текст.

Поясняющий текст выводится с помощью атрибута ALT, значением которого является текст, поясняющий, что должен был бы увидеть пользователь на рисунке:

```
<IMG SRC="computer.gif" ALT="компьютер">
```

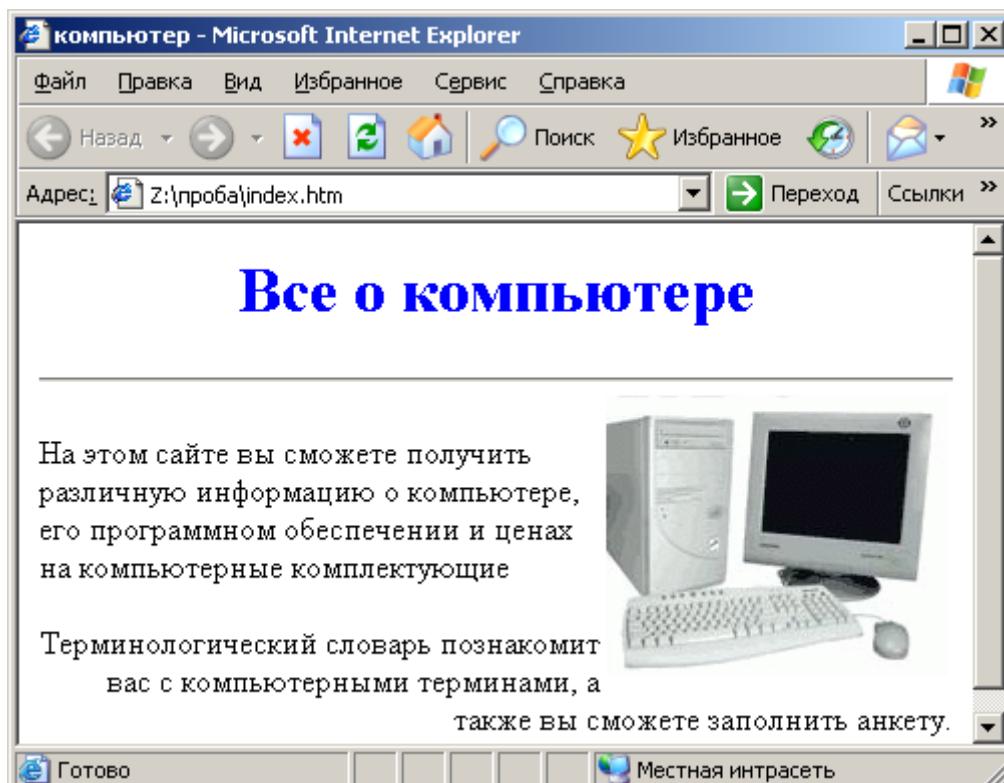
Расположить рисунок относительно текста различными способами позволяет атрибут ALIGN тэга , который может принимать пять различных значений: TOP (верх), MIDDLE (середина), BOTTOM (низ), LEFT (слева) и RIGHT (справа).

Для того чтобы рисунок располагался по правому краю текста, тэг вставки изображения должен принять следующий вид:

```
<IMG SRC="computer.gif" ALT="Компьютер" ALIGN="right">
```

5. В окне приложения Блокнот в контейнер <BODY> вставить перед абзацами текста тэг вставки изображения, просмотреть результат в браузере.

```
<IMG SRC="computer.gif" ALT="Компьютер" ALIGN="right">
```



Практическое занятие № 39. Оформление страницы с помощью каскадных таблиц стилей.

Цель: создание страниц сайта

Практические задания

1. Создать страницы «Программы», «Словарь», «Комплектующие» и «Анкета» и сохранить их в файлах с именами software.htm, glossary.htm, hardware.htm и anketa.htm в каталоге сайта.

Программное обеспечение

1. Системные программы
2. Прикладные программы
 - текстовые редакторы;
 - графические редакторы;
 - электронные таблицы;
 - системы управления базами данных.
3. Системы программирования

Процессор
Центральное устройство компьютера, производящее обработку информации в двоичном коде.

Оперативная память
Устройство в котором хранятся программы и данные.

	Процессоры			
	Частота (Мгц)	Шина данных	Шина адреса	Адресуемая память
Pentium	60-166	64	32	4Гб
Pentium II	200-300	64	36	64 Гб
Pentium III	450-1000	64	36	64 Гб
Pentium 4	1000-3100	64	36	64 Гб

Таблица №1

Анкета

Пожалуйста, введите ваше имя

e-mail:

Укажите, к какой группе пользователей вы себя относите:

- учащийся
- студент
- учитель

Какие из сервисов Интернета вы используете наиболее часто:

- www
- e-mail
- ftp

Internet Explorer

Какую еще информацию вы хотели бы видеть на нашем сайте?

Отправить **Очистить**

2. На титульной странице создать панель навигации.

компьютер

Все о компьютере

На этом сайте вы сможете получить различную информацию о компьютере, его программном обеспечении и ценах на компьютерные комплектующие

Терминологический словарь познакомит вас с компьютерными терминами, а также вы сможете заполнить анкету.



[[Программы](#)] [[Словарь](#)] [[Комплектующие](#)] [[Анкета](#)]

[e-mail: mailbox@provaider.ru](mailto:mailbox@provaider.ru)

Практическое занятие № 40. Обработка цифровых фотографий (кадрирование, исправление перспективы, коррекция уровней, коррекция цвета).

Цель: Изучить базовые операции при редактировании изображений

Практическое задание

1. В папке учебные файлы найдите файл Eagle.psd



Используя инструмент **Магнитное лассо** , выделяйте отдельные чёрные перья и с помощью инструмента **Заливка**  раскрасьте орла в разные цвета.



Сравните

2. Откройте Файлы men.jpg и landscape.jpg



Перейдите к рисунку с изображением человека. Постарайтесь улучшить качество изображения. Впишите изображение мужчины в пейзаж.



Практическое занятие № 41. Ретушь цифровых фотографий.

Цель: Изучить инструменты ретуши

Практическое задание.

Выполнить ретуширование старой фотографии

Например

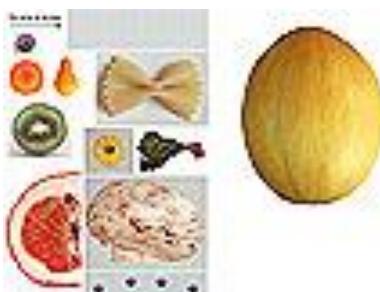


Практическое занятие № 42. Многослойные изображения.

Цель: Изучить работу со слоями

Практическое задание.

1. Из папки Учебные файлы откройте файл vegetables.psd



2. Используя инструменты выделения, выделяя поочередно каждые овощи и фрукты создайте новый рисунок



Практическое занятие № 43. Анимированные изображения.

Цель: Изучить фильтры для создания анимации

Практическое задание.

Скачайте изображение Невалышки. Используя различные инструменты выполните анимацию наклонов Невалышки в разные стороны



Практическое занятие № 44. Векторная графика.

Цель: Изучить отличия векторной графики от растровой

Теория

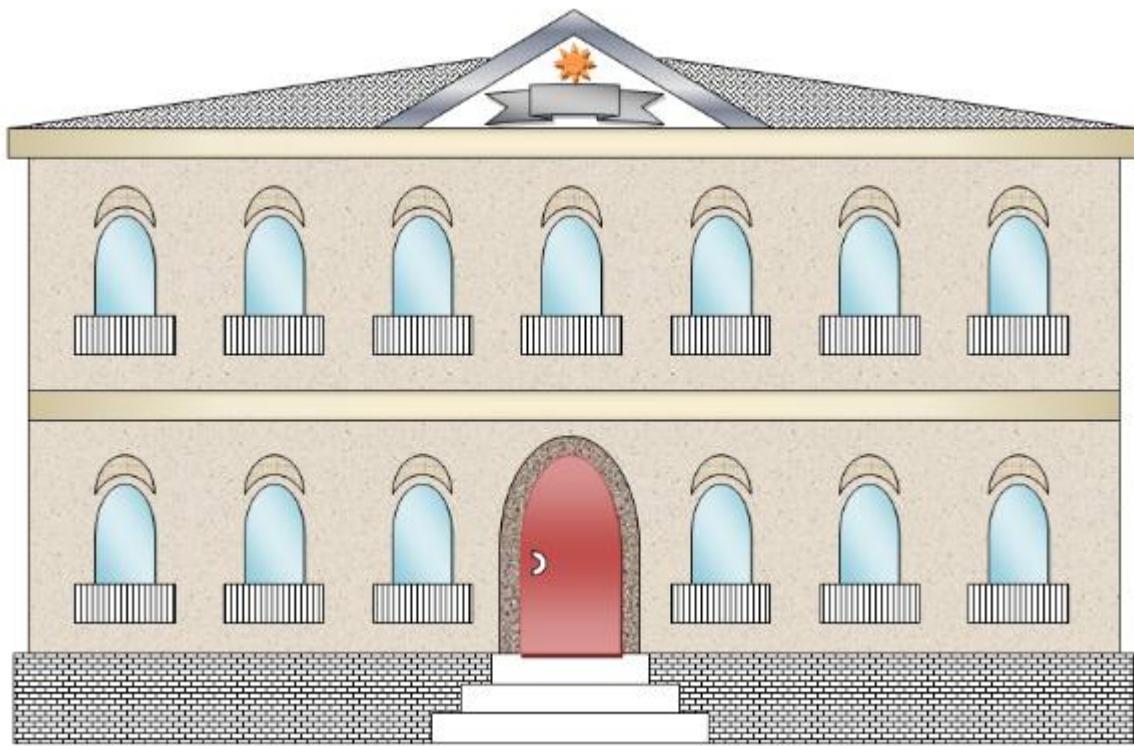
Векторная графика — это категория компьютерной графики, которая для представления изображения использует фигуры, построенные по математическим формулам.

Векторная графика часто противопоставляется растровой. И правильно делает: они отличаются на фундаментальном уровне со всеми вытекающими.

Любое растровое изображение состоит из матрицы (сетки) пикселей. Каждому из этих пикселей приписано его местоположение и цвет. И растровое изображение содержит в себе эту информацию о каждом своём пикселе. Этим объясняется зависимость «веса» растрового изображения от его размера — чем больше в нём пикселей, тем больше информации ему нужно в себе хранить.

Практическое задание

В текстовом процессоре используя автофигуры изобразить рисунок.



Практическое занятие № 45. Создание простых трёхмерных моделей.

Цель: Рассмотреть понятие 3D моделирования. Выяснить достоинства и недостатки 3D-моделей. Рассмотреть основные этапы процесса создания трехмерной модели.

Теория

3D-моделирование — это процесс создания трехмерной модели объекта. Задача 3D-моделирования — разработать визуальный объемный образ желаемого объекта. С помощью трехмерной графики можно и создать точную копию конкретного предмета, и разработать новое, даже нереальное представление до сего момента не существовавшего объекта.

Практическое задание

1. Рассмотреть подходы для создания объёмного изображения
2. Используя ресурсы Интернет выяснить достоинства и недостатки программ для 3D-моделирования.

Название программы	Достоинства	Недостатки

СПИСОК ЛИТЕРАТУРЫ

1. Цветкова М.С. Информатика: учебник / М.С. Цветкова, И.Ю. Хлобыстова. - 6-е изд.,стер. - М.: ИЦ Академия, 2020. - 352 с.,ил. - (Профессиональное образование).
2. Цветкова М.С. Информатика. Практикум для профессий и специальностей естественно - научного и гуманитарного профилей: учеб. пособие / М.С. Цветкова, И.Ю. Хлобыстова. - 3-е изд., стер. - М.: ИЦ Академия, 2019. - 240 с. - (Профессиональное образование)
3. Гребенюк Е.И. Технические средства информатизации: учебник / Е.И. Гребенюк, Н.А. Гребенюк. - 10-е изд. стер. - М.: ИЦ Академия, 2016. - 352 с. - (Профессиональное образование).