

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ ДОНЕЦКОЙ НАРОДНОЙ РЕСПУБЛИКИ ГОСУДАРСТВЕННОЕ УЧРЕЖДЕНИЕ «УЧЕБНО-МЕТОДИЧЕСКИЙ ЦЕНТР СРЕДНЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ»

ГПОУ «ШАХТЕРСКИЙ ТЕХНИКУМ КИНО И ТЕЛЕВИДЕНИЯ ИМЕНИ А.А. ХАНЖОНКОВА»



Открытая дистанционная научно-методической конференция «Инновационные методы и традиционные подходы при проведении лабораторных и практических занятий по компьютерным дисциплинам» среди педагогических работников

образовательных учреждений среднего профессионального образования

СБОРНИК МАТЕРИАЛОВ КОНФЕРЕНЦИИ 26.01.2017г.

Донецк –Шахтерск

Содержание.

Раздел 1. Методика проведения лабораторных и практических работ по информатике, ИКТ и компьютерной графике Бережной А.А. Практическая работа «Система частиц в Blender» по дисциплине «Колньютерная графика» 4 Куликовская Л.В. Лабораторная работа «Microsoft WORD. Форматирование текста» 11 Лебедлицев В.Н. Лабораторная работа «Программирование линейных алгоритмов» 14 Ироскокова О.Н. Практическая работа «Разработка инфологической модели и создание структуры реляционной базы данных. Формирование сложных запросов» 24 Раздел 2. Методика проведения лабораторных и практических работ по специальным компьютерным дисциплинам. Авилова С.А. Практическая работа «Использование функций MS EXCEL в профессиональной деятельности» 37 Басова Г.Н., Елиссева А.Ю. Практическая работа «Форматирование текстовых фрагментов html- страницы» 43 Бурым Е.В. Практическая работа «Форматирование локальной сети» 53 Долинкин А.Ю. Лабораторная работа «Файлы пакстной обработки» 60 Бережная Е.В. Лабораторная работа «Файлы пакстной обработки» 60 Корокошенко Е.Ю. Лабораторная работа «Создание функции пользователя на языке VISUAL ВАSIC в среле ЕХСЕL» 79 Чигринец А.В. Лабораторная работа «Табличный процессор EXCEL. Применение логических, математических и статистических функций» 83 Скрипий А.А. Лабораторная работа «Работа сформани» 87	Введение.	3
Методика проведения лабораторных и практических работ по информатике, ИКТ и компьютерной графике Бережной А.А. Бережной А.А. Практическая работа «Система частиц в Blender» по дисциплине «Компьютерная графика» 4 Куликовская Л.В. 11 Лабораторная работа «Містоsoft WORD. Форматирование текста» 11 Лабораторная работа «Программирование линсйных алгоритмов» 14 Прококова О.Н. 14 Практическая работа «Разработка инфологической модели и создание структуры реляционной базы данных. Формирование сложных запросов» 24 Раздел 2. Методика проведения лабораторных и практических работ по специальным компьютерным дисциплинам. 37 Авилова С.А. Практическая работа «Использование функций MS EXCEL в профессиональной дятельности» 37 Басова Г.Н., Елисеева А.Ю. Практическая работа «Форматирование текстовых фрагментов html- страницы» 43 Бурым Е.В. 19 37 Пабораторная работа «Идоректирование локальной сети» 53 Дабораторная работа «Использование локальной сети» 60 Бережия Е.В. 31 Практическая работа «Идоректирование локальной сети» 53 Дабораторная работа «Использователя на языке VISUAL 84 Крокошенко Е.Ю.	Раздел 1.	
информатике, ИКТ и компьютерной графике Бережной А.А. Практическая работа «Система частиц в Blender» по дисциплине «Компьютерная графика» 4 Куликовская Л.В. Лабораторная работа «Шогозоft WORD. Форматирование текста» 11 Лебединцев В.И. Лабораторная работа «Программирование линейных алгоритмов» 14 Проскокова О.Н. Практическая работа «Разработка инфологической модели и создание структуры реляционной базы данных. Формирование сложных запросов» 24 Раздел 2. Методика проведения лабораторных и практических работ по специальным компьютерным дисциплинам. Авилова С.А. Практическая работа «Использование функций MS EXCEL в профессиональной деятельности» 37 Басова Г.Н., Елисеева А.Ю. Практическая работа «Форматирование текстовых фрагментов html- страницы» 43 Бурым Е.В. Практическая работа «Проектирование локальной сети» 53 Долинкин А.Ю. Лабораторная работа «Файлы пакетной обработки» 60 Бережная Е.В. Лабораторная работа «Исследование гs-триггера» 71 Овсейчук Т.В. Практическое работа «Создание функции пользователя на языке VISUAL BASIC в среде EXCEL» 79 Чигринец А.В. Лабораторная работа «Табличный процессор EXCEL. Применение логическия, математическия и статистических функций» 83 Скрипий А.А. Лабораторная работа «Работа сформация покелов 53 Лабораторная работа «Создание функции пользователя на языке VISUAL BASIC в среде EXCEL» 79 Чигринец А.В. Лабораторная работа «Создание функции пользователя на языке VISUAL BASIC в среде EXCEL» 79	Методика проведения лабораторных и практических работ по	
Бережной А.А. Практическая работа «Система частиц в Blender» по дисциплине «Компьютерная графика» 4 Куликовская Л.В. 11 Лабораторная работа «Microsoft WORD. Форматирование текста» 11 Лабораторная работа «Microsoft WORD. Форматирование текста» 14 Проскокова О.Н. 14 Практическая работа «Разработка инфологической модели и создание структуры реляционной базы данных. Формирование сложных запросов» 24 Раздел 2. 12 Методика проведения лабораторных и практических работ по специальным компьютерным дисциплинам. 14 Авилова С.А. 19 Практическая работа «Использование функций MS EXCEL в профессиональной деятельности» 37 Басова Г.Н., Елиссева А.Ю. 14 Практическая работа «Использование текстовых фрагментов html-страницы» 33 Бурым Е.В. 34 Практическая работа «Проектирование локальной сети» 53 Долинкин А.Ю. 36 Лабораторная работа «Изучение фильтра отображения пакетов» 64 Крокошенко Е.Ю. 71 Лабораторная работа «Создание функции пользователя на языке VISUAL 83 Вахог с среде ЕХСЕL» 79<	информатике, ИКТ и компьютерной графике	
Практическая работа «Система частиц в Веленет» по дисциплине «Колпьютерная графика» 4 Куликовская Л.В. Лабораторная работа «Microsoft WORD. Форматирование текста» 11 Лебедницев В.И. Лабораторная работа «Программирование линейных алгоритмов» 14 Проктическая работа «Разработка инфологической модели и создание структуры реляционной базы данных. Формирование сложных запросов» 24 Раздел 2. Мстодика проведения лабораторных и практических работ по специальным компьютерным дисциплинам. Авилова С.А. Практическая работа «Использование функций MS EXCEL в профессиональной деятельности» 37 Басова Г.Н., Елисеева А.Ю. Практическая работа «Форматирование текстовых фрагментов html- страницы» 43 Бурым Е.В. Практическая работа «Проектирование локальной сети» 53 Долинкин А.Ю. Лабораторная работа «Изучение фильтра отображения пакетов» 64 Крокошенко Е.Ю. Лабораторная работа «Создание функции пользователя на языке VISUAL BASIC в среде EXCEL» 79 Чигринец А.В. Лабораторная работа «Табличный процессор EXCEL. Применение логическия, математическия и статистическия функций» 83 Скриний А.А.	Бережной А.А.	
«Колпьютерная графика» 4 Куликовская Л.В. 11 Лабораторная работа «Microsoft WORD. Форматирование текста» 11 Лабораторная работа «Программирование линейных алгоритмов» 14 Проксокова О.Н. 14 Практическая работа «Разработка инфологической модели и создание структуры реляционной базы данных. Формирование сложных запросов» 24 Раздел 2. Методика проведения лабораторных и практических работ по специальным компьютерным дисциплинам. 24 Авилова С.А. Практическая работа «Использование функций MS EXCEL в профессиональной деятельности» 37 Басова Г.Н., Елиссева А.Ю. 19 14 Практическая работа «Форматирование текстовых фрагментов html-страницы» 37 Бурым Е.В. 19 14 Практическая работа «Проектирование локальной сети» 53 Долинкин А.Ю. 60 60 Лабораторная работа «Изучение фильтра отображения пакетов» 64 Крокошенко Е.Ю. 71 Овсейчук Т.В. Практическое работа «Создание кункции пользователя на языке VISUAL ВАSIC в среде ЕХСЕL» 79 Чигринец А.В. Лабораторная работа «Создание функции пользователя на языке VISUAL ВАSIC в среде ЕХСЕL» 79 Чигринец А.В. Лабораторная работа «Таб	Практическая работа «Система частиц в Blender» по дисциплине	
Куликовская Л.В. 11 Лабораторная работа «Microsoft WORD. Форматирование текста» 11 Лабораторная работа «Программирование линейных алгоритмов» 14 Проскокова О.Н. 14 Практическая работа «Разработка инфологической модели и создание структуры реляционной базы данных. Формирование сложных запросов» 24 Раздел 2. Методика проведения лабораторных и практических работ по специальным компьютерным дисциплинам. 37 Авилова С.А. Практическая работа «Использование функций MS EXCEL в профессиональной деятельности» 37 Басова Г.Н., Елиссева А.Ю. Практическая работа «Форматирование текстовых фрагментов html-страницы» 43 Бурым Е.В. 19 37 Пабораторная работа «Изручение фильтра отображения пакетов» 64 Крокошенко Е.Ю. 71 0всейчук Т.В. Практическое работа «Исследование гs-тригтера» 71 0всейчук Т.В. Практическое работа «Исследование rs-триттера» 71 0всейчук Т.В. Практическое работа «Создание функции пользователя на языке VISUAL 83 Крокошенко Е.Ю. 71 0всейчук Т.В. 71 Пабораторная работа «Исследование гs-тригтера» 71 0всейчук Т.В. 79 Чигринец А.В. Лабораторная	«Компьютерная графика»	4
Лабораторная работа «Містозоft WORD. Форматирование текста» 11 Лебединцев В.И. 14 Лабораторная работа «Программирование линейных алгоритмов» 14 Ироскокова О.Н. 14 Практическая работа «Разработка инфологической модели и создание структуры реляционной базы данных. Формирование сложных запросов» 24 Раздел 2. Мстодика проведения лабораторных и практических работ по специальным компьютерным дисциплинам. 37 Авилова С.А. Практическая работа «Использование функций MS EXCEL в профессиональной деятельности» 37 Басова Г.Н., Елиссева А.Ю. Практическая работа «Форматирование текстовых фрагментов html-страницы» 43 Бурым Е.В. 11 35 Пабораторная работа «Изучение фильтра отображения пакетов» 64 Крокошенко Е.Ю. 71 36 Лабораторная работа «Исследование гs-триггера» 71 Овсейчук Т.В. 71 36 Практическое работа «Исследование гs-триггера» 71 Овсейчук Т.В. 71 37 Пабораторная работа «Исследование гs-триггера» 71 Овсейчук Т.В. 79 71 Овсейчук Т.В. 79 79 Практическое работа «Создание функции пользователя на язык	Куликовская Л.В.	
Лебединцев В.И. 14 Лабораторная работа «Программирование линейных алгоритмов» 14 Проскокова О.Н. 14 Практическая работа «Разработка инфологической модели и создание структуры реляционной базы данных. Формирование сложных запросов» 24 Раздел 2. 14 Методика проведения лабораторных и практических работ послециальным компьютерным дисциплинам. 24 Авилова С.А. 17 Практическая работа «Использование функций MS EXCEL в профессиональной деятельности» 37 Басова Г.Н., Елиссева А.Ю. 37 Практическая работа «Форматирование текстовых фрагментов html-страницы» 43 Курьм Е.В. 37 Пабораторная работа «Форматирование локальной сети» 53 Долинкин А.Ю. 34 Лабораторная работа «Ироектирование локальной сети» 53 Абораторная работа «Изучение фильтра отображения пакетов» 64 Крокошенко Е.Ю. 71 Лабораторная работа «Исследование гs-тритера» 71 Овсейчук Т.В. 79 Практическое работа «Создание функции пользователя на языке VISUAL 83 ВАSIC в среде ЕХСЕL» 79 Чигринец А.В. Лабораторная работа «Табличный процессор ЕХСЕL. Применение логических,	Лабораторная работа «Microsoft WORD. Форматирование текста»	11
Лабораторная работа «Программирование линеиных алгоритмов» 14 Ироскокова О.Н. Практическая работа «Разработка инфологической модели и создание структуры реляционной базы данных. Формирование сложных запросов» 24 Раздел 2. Методика проведения лабораторных и практических работ по 24 Методика проведения лабораторных и практических работ по специальным компьютерным дисциплинам. 36 Авилова С.А. Практическая работа «Использование функций MS EXCEL в 37 Басова Г.Н., Елиссева А.Ю. 37 Практическая работа «Форматирование текстовых фрагментов html- 37 страницы» 43 Бурым Е.В. 37 Пабораторная работа «Файлы пакетной обработки» 60 Бережная Е.В. 34 Лабораторная работа «Создание фильтра отображения пакетов» 64 Крокошенко Е.Ю. 71 Лабораторная работа «Создание функции пользователя на языке VISUAL 85 ВАSIC в среде ЕХСЕL» 79 Чигринец А.В. 74 Лабораторная работа «Табличный процессор ЕХСЕL. Применение логических, математических и статистических функций» 83 Скрипий А.А. 79 Чигринец А.В. 79 Лаборатор	Лебединцев В.И.	14
Проскокова О.Н. Практическая работа «Разработка инфологической модели и создание структуры реляционной базы данных. Формирование сложных запросов» 24 Раздел 2. Методика проведения лабораторных и практических работ по специальным компьютерным дисциплинам. Авилова С.А. Практическая работа «Использование функций MS EXCEL в профессиональной деятельности» 37 Басова Г.Н., Елисеева А.Ю. Практическая работа «Форматирование текстовых фрагментов html- страницы» 43 Бурым Е.В. Практическая работа «Проектирование локальной сети» 53 Долинкин А.Ю. Лабораторная работа «Изучение фильтра отображения пакетов» 64 Крокошенко Е.Ю. Лабораторная работа «Создание функции пользователя на языке VISUAL BASIC в среде EXCEL» 79 Чигринец А.В. Лабораторная работа «Табличный процессор EXCEL. Применение логических, математических и статистических функций» 83 Скрипий А.А. Лабораторная работа «Работа с формами» 87	Лабораторная работа «Программирование линейных алгоритмов»	14
Практическая работа «Разраоотка инфологической модели и создание 24 структуры реляционной базы данных. Формирование сложных запросов» 24 Раздел 2. Методика проведения лабораторных и практических работ по специальным компьютерным дисциплинам. Авилова С.А. Практическая работа «Использование функций MS EXCEL в апрофессиональной деятельности» 37 Басова Г.Н., Елисеева А.Ю. Практическая работа «Форматирование текстовых фрагментов html- страницы» 43 Бурым Е.В. Проектирование локальной сети» 53 Долинкин А.Ю. Лабораторная работа «Файлы пакетной обработки» 60 Бережная Е.В. Лабораторная работа «Исследование гs-тригтера» 71 Овсейчук Т.В. 71 Овсейчук Т.В. 71 Практическое работа «Создание функции пользователя на языке VISUAL ВАSIC в среде EXCEL» 79 Чигринец А.В. Лабораторная работа «Табличный процессор EXCEL. Применение логических, математических и статистических функций» 83 Скрипий А.А. Лабораторная работа «Работа с формами» 87	проскокова О.н.	
структуры реляционнои оазы данных. Формирование сложных запросов» 24 Раздел 2. Методика проведения лабораторных и практических работ по специальным компьютерным дисциплинам. Авилова С.А. Практическая работа «Использование функций MS EXCEL в профессиональной деятельности» 37 Басова Г.Н., Елисеева А.Ю. 37 Практическая работа «Форматирование текстовых фрагментов html-страницы» 43 Бурым Е.В. 33 Практическая работа «Проектирование локальной сети» 53 Долинкин А.Ю. 60 Бережная Е.В. 60 Лабораторная работа «Изучение фильтра отображения пакетов» 64 Крокошенко Е.Ю. 71 Овсейчук Т.В. 71 Практическое работа «Создание функции пользователя на языке VISUAL 8ASIC в среде EXCEL.» Лабораторная работа «Табличный процессор EXCEL. Применение логических, математических и статистических функций» 83 Скрипий А.А. 79 Лабораторная работа «Работа с формами» 87	Практическая работа «Разработка инфологической модели и создание	•
Раздел 2. Методика проведения лабораторных и практических работ по специальным компьютерным дисциплинам. Авилова С.А. Практическая работа «Использование функций MS EXCEL в профессиональной деятельности» 37 Басова Г.Н., Елисеева А.Ю. Практическая работа «Форматирование текстовых фрагментов html-страницы» страницы» 43 Бурым Е.В. Практическая работа «Форматирование локальной сети» 53 Долинкин А.Ю. 53 Лабораторная работа «Файлы пакетной обработки» 60 Бережная Е.В. 71 Лабораторная работа «Исследование гs-тригтера» 71 Овсейчук Т.В. 71 Практическое работа «Создание функции пользователя на языке VISUAL 8ASIC в среде EXCEL» Лабораторная работа «Табличный процессор EXCEL. Применение логических, математических и статистических функций» 83 Скрипий А.А. 74 Лабораторная работа «Работа с формами» 87	структуры реляционнои оазы данных. Формирование сложных запросов»	24
Методика проведения лабораторных и практических работ по специальным компьютерным дисциплинам. Авилова С.А. Практическая работа «Использование функций MS EXCEL в профессиональной деятельности» 37 Басова Г.Н., Елисеева А.Ю. 37 Практическая работа «Форматирование текстовых фрагментов html- страницы» 43 Бурым Е.В. 43 Практическая работа «Форматирование локальной сети» 53 Долинкин А.Ю. 53 Лабораторная работа «Файлы пакетной обработки» 60 Бережная Е.В. 64 Лабораторная работа «Изучение фильтра отображения пакетов» 64 Крокошенко Е.Ю. 71 Овсейчук Т.В. 71 Практическое работа «Создание функции пользователя на языке VISUAL 83 ВАSIC в среде ЕХСЕL» 79 Чигринец А.В. 71 Лабораторная работа «Табличный процессор ЕХСЕL. Применение логических, математических и статистических функций» 83 Скрипий А.А. 83 Лабораторная работа «Работа с формами» 87	Раздел 2.	
специальным компьютерным дисциплинам. Авилова С.А. Практическая работа «Использование функций MS EXCEL в профессиональной деятельности» 37 Басова Г.Н., Елисеева А.Ю. Практическая работа «Форматирование текстовых фрагментов html- страницы» 43 Бурым Е.В. Практическая работа «Проектирование локальной сети» 53 Долинкин А.Ю. Лабораторная работа «Файлы пакетной обработки» 60 Бережная Е.В. Лабораторная работа «Изучение фильтра отображения пакетов» 64 Крокошенко Е.Ю. Лабораторная работа «Исследование гs-триггера» 71 Овсейчук Т.В. Практическое работа «Создание функции пользователя на языке VISUAL ВАSIC в среде EXCEL» 79 Чигринец А.В. Лабораторная работа «Табличный процессор EXCEL. Применение логических, математических и статистических функций» 83 Скрипий А.А. Лабораторная работа «Работа с формами» 87	Методика проведения лабораторных и практических работ по	
Авилова С.А. Практическая работа «Использование функций MS EXCEL в профессиональной деятельности» 37 Басова Г.Н., Елисеева А.Ю. 19 Практическая работа «Форматирование текстовых фрагментов html- 43 страницы» 43 Бурым Е.В. 53 Практическая работа «Проектирование локальной сети» 53 Долинкин А.Ю. 60 Бережная Е.В. 64 Крокошенко Е.Ю. 71 Лабораторная работа «Изучение фильтра отображения пакетов» 64 Крокошенко Е.Ю. 71 Овсейчук Т.В. 71 Практическое работа «Создание функции пользователя на языке VISUAL 84 ВАSIC в среде EXCEL» 79 Чигринец А.В. 71 Лабораторная работа «Создание функции пользователя на языке VISUAL 83 Скрипий А.А. 73 Лабораторная работа «Создание функции пользователя на языке VISUAL 83 Скрипий А.А. 79 Лабораторная работа «Табличный процессор EXCEL. Применение 70 Лабораторная работа «Работа с формами» 83	специальным компьютерным дисциплинам.	
Практическая работа «Использование функций MS EXCEL в профессиональной деятельности» 37 Басова Г.Н., Елисеева А.Ю. Практическая работа «Форматирование текстовых фрагментов html- страницы» 43 Бурым Е.В. 43 Практическая работа «Проектирование локальной сети» 53 Долинкин А.Ю. 53 Лабораторная работа «Файлы пакетной обработки» 60 Бережная Е.В. 64 Крокошенко Е.Ю. 71 Лабораторная работа «Исследование гs-тригтера» 71 Овсейчук Т.В. 71 Практическое работа «Создание функции пользователя на языке VISUAL 79 Чигринец А.В. 71 Лабораторная работа «Табличный процессор EXCEL. Применение логических, математических и статистических функций» 83 Скрипий А.А. 73	Авилова С.А.	
профессиональной деятельности» 37 Басова Г.Н., Елисеева А.Ю. Практическая работа «Форматирование текстовых фрагментов html- страницы» 43 Бурым Е.В. 1 Практическая работа «Проектирование локальной сети» 53 Долинкин А.Ю. 53 Долинкин А.Ю. 60 Бережная Е.В. 60 Лабораторная работа «Файлы пакетной обработки» 60 Крокошенко Е.Ю. 64 Лабораторная работа «Изучение фильтра отображения пакетов» 64 Крокошенко Е.Ю. 71 Овсейчук Т.В. 71 Практическое работа «Создание функции пользователя на языке VISUAL 79 Чигринец А.В. 79 Лабораторная работа «Табличный процессор EXCEL. Применение логических, математических и статистических функций» 83 Скрипий А.А. 71 Лабораторная работа «Работа с формами» 87	Практическая работа «Использование функций MS EXCEL в	
Басова Г.Н., Елисеева А.Ю. Практическая работа «Форматирование текстовых фрагментов html- страницы» 43 Бурым Е.В. 53 Практическая работа «Проектирование локальной сети» 53 Долинкин А.Ю. 53 Лабораторная работа «Файлы пакетной обработки» 60 Бережная Е.В. 60 Лабораторная работа «Изучение фильтра отображения пакетов» 64 Крокошенко Е.Ю. 71 Лабораторная работа «Исследование гs-триггера» 71 Овсейчук Т.В. 71 Практическое работа «Создание функции пользователя на языке VISUAL 851С в среде ЕХСЕL» Лабораторная работа «Табличный процессор ЕХСЕL. Применение логических, математических и статистических функций» 83 Скрипий А.А. 79 Лабораторная работа «Работа с формами» 87	профессиональной деятельности»	37
Практическая работа «Форматирование текстовых фрагментов html- 43 страницы» 43 Бурым Е.В. 53 Практическая работа «Проектирование локальной сети» 53 Долинкин А.Ю. 53 Лабораторная работа «Файлы пакетной обработки» 60 Бережная Е.В. 60 Лабораторная работа «Изучение фильтра отображения пакетов» 64 Крокошенко Е.Ю. 71 Лабораторная работа «Исследование гs-триггера» 71 Овсейчук Т.В. 71 Практическое работа «Создание функции пользователя на языке VISUAL 8ASIC в среде EXCEL» ВАSIC в среде EXCEL» 79 Чигринец А.В. Лабораторная работа «Табличный процессор EXCEL. Применение логических, математических и статистических функций» 83 Скрипий А.А. Лабораторная работа «Работа с формами» 87	Басова Г.Н., Елисеева А.Ю.	
страницы» 43 Бурым Е.В. Практическая работа «Проектирование локальной сети» 53 Долинкин А.Ю. 53 Долинкин А.Ю. 60 Бережная работа «Файлы пакетной обработки» 60 Бережная Е.В. 64 Лабораторная работа «Изучение фильтра отображения пакетов» 64 Крокошенко Е.Ю. 71 Лабораторная работа «Исследование гs-триггера» 71 Овсейчук Т.В. 71 Практическое работа «Создание функции пользователя на языке VISUAL 8ASIC в среде EXCEL» ВАSIC в среде EXCEL» 79 Чигринец А.В. 71 Лабораторная работа «Табличный процессор EXCEL. Применение логических, математических и статистических функций» 83 Скрипий А.А. 74 Лабораторная работа «Работа с формами» 87	Практическая работа «Форматирование текстовых фрагментов html-	
Бурым Е.В. 11 Практическая работа «Проектирование локальной сети» 53 Долинкин А.Ю. 30 Лабораторная работа «Файлы пакетной обработки» 60 Бережная Е.В. 64 Лабораторная работа «Изучение фильтра отображения пакетов» 64 Крокошенко Е.Ю. 71 Лабораторная работа «Исследование гs-триггера» 71 Овсейчук Т.В. 71 Практическое работа «Создание функции пользователя на языке VISUAL 85 ВАSIC в среде ЕХСЕL» 79 Чигринец А.В. 71 Лабораторная работа «Табличный процессор ЕХСЕL. Применение 83 Скрипий А.А. 83 Лабораторная работа «Работа с формами» 87	страницы»	43
Практическая работа «Проектирование локальной сети» 53 Долинкин А.Ю. Лабораторная работа «Файлы пакетной обработки» 60 Бережная Е.В. 60 Лабораторная работа «Изучение фильтра отображения пакетов» 64 Крокошенко Е.Ю. 71 Лабораторная работа «Исследование гs-триггера» 71 Овсейчук Т.В. 71 Практическое работа «Создание функции пользователя на языке VISUAL 83 ВАSIC в среде ЕХСЕL» 79 Чигринец А.В. Лабораторная работа «Табличный процессор ЕХСЕL. Применение логических, математических и статистических функций» 83 Скрипий А.А. 71 Лабораторная работа «Работа с формами» 87	Бурым Е.В.	
Долинкин А.Ю. Лабораторная работа «Файлы пакетной обработки» 60 Бережная Е.В. Лабораторная работа «Изучение фильтра отображения пакетов» 64 Крокошенко Е.Ю. 71 Лабораторная работа «Исследование гs-триггера» 71 Овсейчук Т.В. 71 Практическое работа «Создание функции пользователя на языке VISUAL 79 Чигринец А.В. 79 Лабораторная работа «Табличный процессор EXCEL. Применение 79 Огических, математических и статистических функций» 83 Скрипий А.А. Лабораторная работа «Работа с формами» 87	Практическая работа «Проектирование локальной сети»	53
Лабораторная работа «Файлы пакетной обработки» 60 Бережная Е.В. Лабораторная работа «Изучение фильтра отображения пакетов» 64 Крокошенко Е.Ю. 71 Лабораторная работа «Исследование rs-триггера» 71 Овсейчук Т.В. 71 Практическое работа «Создание функции пользователя на языке VISUAL 79 Чигринец А.В. 79 Лабораторная работа «Табличный процессор ЕХСЕL. Применение 79 Оксейчук, математических и статистических функций» 83 Скрипий А.А. 74 Лабораторная работа «Работа с формами» 87	Долинкин А.Ю.	
Бережная Е.В. Лабораторная работа «Изучение фильтра отображения пакетов» 64 Лабораторная работа «Исследование rs-триггера» 71 Лабораторная работа «Исследование rs-триггера» 71 Овсейчук Т.В. 71 Практическое работа «Создание функции пользователя на языке VISUAL 79 ВАЅІС в среде ЕХСЕL» 79 Чигринец А.В. 71 Лабораторная работа «Табличный процессор ЕХСЕL. Применение 83 Скрипий А.А. 83 Лабораторная работа «Работа с формами» 87	Лабораторная работа «Файлы пакетной обработки»	60
Лабораторная работа «Изучение фильтра отображения пакетов» 64 Крокошенко Е.Ю. Лабораторная работа «Исследование гs-триггера» 71 Овсейчук Т.В. Практическое работа «Создание функции пользователя на языке VISUAL 79 Чигринец А.В. 79 4 Лабораторная работа «Табличный процессор EXCEL. Применение 83 Скрипий А.А. 83 Лабораторная работа «Работа с формами» 87	Бережная Е.В.	
Крокошенко Е.Ю. Лабораторная работа «Исследование гs-триггера» 71 Овсейчук Т.В. Практическое работа «Создание функции пользователя на языке VISUAL 79 ВАЅІС в среде ЕХСЕL» 79 Чигринец А.В. 71 Лабораторная работа «Табличный процессор ЕХСЕL. Применение 79 Скрипий А.А. 83 Лабораторная работа «Работа с формами» 87	Лабораторная работа «Изучение фильтра отображения пакетов»	64
Лабораторная работа «Исследование гs-триггера» 71 Овсейчук Т.В. Практическое работа «Создание функции пользователя на языке VISUAL BASIC в среде EXCEL» 79 Чигринец А.В. Лабораторная работа «Табличный процессор EXCEL. Применение логических, математических и статистических функций» 83 Скрипий А.А. Лабораторная работа «Работа с формами» 87	Крокошенко Е.Ю.	
Овсейчук Т.В. Практическое работа «Создание функции пользователя на языке VISUAL BASIC в среде EXCEL» 79 Чигринец А.В. 79 Лабораторная работа «Табличный процессор EXCEL. Применение 83 Скрипий А.А. 83 Лабораторная работа «Работа с формами» 87	Лабораторная работа «Исследование rs-триггера»	71
Практическое работа «Создание функции пользователя на языке VISUAL BASIC в среде EXCEL» 79 Чигринец А.В. Лабораторная работа «Табличный процессор EXCEL. Применение логических, математических и статистических функций» 83 Скрипий А.А. Лабораторная работа «Работа с формами» 87	Овсейчук Т.В.	
 ВАЅІС в среде ЕХСЕL» Чигринец А.В. Лабораторная работа «Табличный процессор ЕХСЕL. Применение логических, математических и статистических функций» 83 Скрипий А.А. Лабораторная работа «Работа с формами» 	Практическое работа «Создание функции пользователя на языке VISUAL	
Чигринец А.В. Лабораторная работа «Табличный процессор EXCEL. Применение логических, математических и статистических функций» 83 Скрипий А.А. Лабораторная работа «Работа с формами» 87	BASIC в среде EXCEL»	79
Лабораторная работа «Табличный процессор EXCEL. Применение логических, математических и статистических функций» 83 Скрипий А.А. Лабораторная работа «Работа с формами» 87	Чигринец А.В.	
логических, математических и статистических функций» 83 Скрипий А.А. Лабораторная работа «Работа с формами» 87	Лабораторная работа «Табличный процессор EXCEL. Применение	
Скрипий А.А. Лабораторная работа «Работа с формами» 87	логических, математических и статистических функций»	83
Лабораторная работа «Работа с формами» 87	Скрипий А.А.	
	Лабораторная работа «Работа с формами»	87
Чубарьян А.С.	Чубарьян А.С.	
Лабораторная работа «Использование линейного блочного кодирования» 93	Лабораторная работа «Использование линейного блочного кодирования»	93

Введение.

Стандартом предусмотрено выполнение лабораторных работ творческого характера с самостоятельным выбором подходов к решению практических задач, изучаемых процессов, систем, явлений и иных объектов, анализом результатов и формулированием выводов.

Предусмотрено право преподавателя в рамках стандарта учебной дисциплины выбирать методы и средства проведения лабораторных работ, обеспечивающих высокое качество учебного процесса. При проведении лабораторных работ студент руководствуется методическими указаниями (рекомендациями), утвержденными кафедрой.

Рекомендации содержат цель и задачи, теоретическое обоснование, описание установки, методику проведения работы и другие указания.

При проведении инновационных работ, в отличие от типовых, студентам сообщаются необходимые теоретические сведения и предлагается самостоятельно продумать порядок проведения работы, определить перечень подлежащих определению (измерению) параметров, последовательность проведения процедур и хода выполнения работы.

Проведение конференции позволяет произвести демонстрацию своего опыта преподавателями компьютерных дисциплин Донецкой Народной Республики.

В данном сборнике представлены методические разработки лабораторных и практических работ по компьютерным дисциплинам самого широко спектра – от «Информатики» до «Безопасности функционирования компьютерных сетей».

3

ПРАКТИЧЕСКАЯ РАБОТА № 12 «СИСТЕМА ЧАСТИЦ В BLENDER» ПО ДИСЦИПЛИНЕ «КОМПЬЮТЕРНАЯ ГРАФИКА»

Бережной Александр Александрович ГПОУ «Шахтерский техникум» ГО ВПО «Донецкий национальный университет экономики и торговли имени Михаила Туган-Барановского», преподаватель дисциплины «Компьютерная графика», специалист

План занятия № 17

Дисциплина: КОМПЬЮТЕРНАЯ ГРАФИКА

Дата

Группа ХКМ 1/1

Тема занятия Системы частиц в Блендере

Вид занятия практическое

Тип занятия занятие по первоначальному формированию умений и навыков;

Формы обучения _____ фронтальная, индивидуальная

Цели занятия:

учебные Выявление и закрепление знаний по теме «3D моделирование».

Выявление умения студентов работать с 3D-редактором «Blender».

Изучение особенностей использования системы частиц для моделирования больших массивов однотипных объектов.

развивающие Способствовать формированию и развитию познавательного интереса учащихся к дисциплине

Способствовать формированию и развитию самостоятельности учащихся

Способствовать развитию абстрактного мышления

воспитательные Воспитание творческого отношения к учебной деятельности

Формируемые компетенции

- Использование для познания окружающего мира различных естественнонаучных методов: наблюдение, измерение, моделирование;
- Овладение способами решения теоретических и экспериментальных задач моделирования и обработки информации;
- Владение монологической и диалогической речью. Способность понимать точку зрения собеседника и признавать право на иное мнение; облекать свою мысль в графическую форму.

Образовательные технологии _____ совместное традиционное обучение

Место проведения _____кабинет № 236

Оснащение:

методическое: учебная программа; инструкция;

материально-техническое: ПК.

Межпредметные связи _____ Физика, Информатика,

Список литературы: Березовский В.С. Основы компьютерной графики: учебное пособие/ В.С. Березовский. - К.: Изд. группа BHV, 2011. – 400 с.

ИНСТРУКТИВНО-МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ВЫПОЛНЕНИЮ ПРАКТИЧЕСКОЙ РАБОТЫ № 12

Тема: Системы частиц в Блендере

Мотивация: Часто возникает необходимость в создании большого количества повторяющихся объектов. Например, дождинки, камешки на земле, пылинки или снежинки, волосы, наконец. Сделать их без дополнительного инструментария затруднительно. Для таких случаев в Блендере используются системы частиц, то есть системы, состоящие из эмиттера (источника повторяющихся объектов) и частиц (повторяющихся объектов), исходящих из эмиттера. Системы бывают двух типов: эмиссия и волосы.

1. В случае «эмиссии» частицы выпадают из эмиттера с определенным ускорением в определенный промежуток времени.

2. В случае «волос» частицы располагаются на поверхности эмиттера.

Есть несколько различных типов частиц, как для первого, так и для второго случая.

Цель данной работы – создание при помощи нескольких систем частиц модели цветущей лужайки.

Ход работы.

1. Подготовка, настройка окружения и освещения.

1.1. Окружение(задний план) настраивается во вкладке «World».

для создания эффекта неба ставится галочка на «Blend Sky» и выбираются цвета:

а) цвет горизонта (Horizon Color) - белый (FFFFF);

б) цвет зенита (Zenith Color) - голубой (3A95FF) (Рисунок1).

1.2. Освещение складывается из рассеянного света, который

можно создать, включив Ambient Occlusion в той же вкладке «World» Для прямого освещения добавим на сцену лампу Point(Add—Lamp—Point), включим в настройках лампы тени(Ray Shadow).

2. Создание эмиттера.

Создадим основу для нашей системы частиц. Добавим на сцену плоскость (Add — Mesh — Plane), перейдем в режим редактирования и подразделим плоскость. Для этого применим Subdivide в Т-панели несколько раз (Рисунок 2.). Subdivide разделяет только выделенные оранжевым ребра, поэтому выделенным должны быть все ребра плоскости. Выделить всё можно при помощи клавиши «латинское А». Эта клавиша в первое нажатие снимает выделение со всех

элементов, а следующим нажатием выделяет всё. Систему частиц можно добавить к

World Vorld Vorld E Vorld E

Рисунок 1.

1 • 1 • 1 • Close



Рисунок 2.

эмиттеру и без подразделения. Подразделение требуется для возможности более тонко управлять результатами использования системы частиц.

3. Добавление системы частиц типа «волосы».

Перейдем в объектный режим и назначим плоскости систему частиц, выбрав тип системы «волосы» (Рисунок 3). Из нашей плоскости теперь торчит вверх 1000 волосков. Настроим их теперь так, чтобы было похоже на траву: а) Уменьшим длину волос

111 11 11 11 11 11 1 4

Рисунок 3.

Length.

б) Для того, чтобы они

были разной длины, увеличим Random во вкладке **Render.** в) Включим Children. Это сделает для каждого волоса несколько копий-«потомков» расположенных вокруг «родителя» по заданным правилам. Количество потомков задается чуть ниже в полях Display – количество потомков на превью и Render – количество потомков на финальном рендере. Поставим там и там по 10, и выберем метод Simple или Interpolated. При этом во вкладке появились настройки «потомков». Покрутите Clump и Shape, которые отвечают за форму пучка на концах и в начале пучка потомков. Попробуйте, как воздействуют на форму потомков другие настройки в этой вкладке. (Рисунок 4)



Рисунок 4

4. Настройка материалов.

4.1. Материал земли (Рисунок 5).

4.2. Второй материал – материал травы (Рисунок 6). Для этого нажмем плюсик на списке материалов этого объекта, чем создадим новый материал (минусом выделенный материал удаляется с объекта). При этом материалом по умолчанию для этого объекта остается верхний в списке, все материалы ниже видны только тогда, когда их назначат отдельным частям объекта, в данном случае волосам(траве).



Рисунок 5.



Рисунок 6.

Кроме зеленого цвета во вкладке Strand назначим толщину волоса-травинки у корня Root и на конце Tip.

4.3. Назначение материала травы частицам.

Теперь перейдем в настройки системы частиц и назначим им Material .001(Рисунок 7).

((-			
	00 8 7	0 🛛	‡ 💟	
ParticleSystem			6	0 +
	Constant I			
Settings:	Parti	cleSettings		F + X
Туре:	Hair	ŧ	Seed:	0)
Regrow	Advance	ed	(Segments	: 5 🖻
▼ Emission				
(* Number:	1000)	Hair Le	ngth:	0.780
Use Modifier Stack				
► ■ Hair dynamics				
▼ Render				
Material.001	;	Parent:	0	Ì
🗹 Emitter		Unboi	n	
Parents		Died		
None	Path	Objec	t	Group
Strand render		Timing:		
Adaptive render		Absol	ute Path Time	e.
	5)	Start:		0.000
Pixel:	3)	End:		1.000
B-Spline		Random	n	0.662
(* Steps:	3)			
► Display				
▼ Children				
None	Sim	nple	Intera	olated

Рисунок 7.



Первый уровень сложности выполнения работы.

Рисунок 8.

5. Создание объекта для второй системы частиц, использующей этот объект в качестве образца для частиц.

5.1. Лепестки. Откроем другой слой, перейдем на вид сверху (клавиша 7 на цифровой клавиатуре справа). Добавим на сцену окружность, перейдем в режим редактирования, выберем элемент «ребро»(Edge) и применим «шахматное выделение» (Рисунок 9).



Рисунок 9.

Выберем тип центра трансформаций (точку, относительно которой будут производиться вращения, масштабирования частей этого объекта). В данном случае трансформации будут происходить относительно геометрического центра объекта. Теперь экструдируем выделенные ребра (латинское «Е» и ЛКМ), после этого масштабируем(«S», растянуть мышкой, после нажать ЛКМ). Получились такие лепестки. Назначим им материал белого цвета.

5.2. Сердцевина. Перейдем в объектный режим и добавим на сцену сферу. Масштабируем ее по оси Z (S, потом Z, потом сжать мышкой и нажать ЛКМ)

5.3. Стебель.

Теперь добавим цилиндр, увеличим его по оси Z, а потом уменьшим весь, чтобы получился тонкий стебель. Назначим ему зеленый материал. Передвинем его под цветок (Рисунок 10). 5.4. Объединение в один объект.

Выделим с Shift все части цветка и объединим в один объект (Ctrl+J). Это необходимо, чтобы назначить объединенный объект частицей-волосом в новой системе частиц, которую мы создадим на нашей плоскости-полянке.



Рисунок 10.

5.5.Перенос центра объекта.

Необходимо переставить центр объекта в нижнюю точку стебля, потому что именно эта точка будет находиться на плоскости у каждого цветка-частицы. Для этого перейдем в вид спереди и кликнем ЛКМ на нижнюю границу стебля, потом перейдем в вид справа и опять кликнем ЛКМ на нижнюю границу стебля, чтобы поставить курсор

в двух проекциях. После этого найдем в Т-панели «Set origin» и выберем «Origin to 3D cursor» После этого центр объекта перейдет в точку курсора.

Попробуйте вращать цветок, и вы увидите, что он вращается вокруг нового центра.

В свойствах объекта переименуйте полученный объект в «цветок» (Рисунок 11).

6. Добавление второй системы частиц, изображающей цветы.

C Shift включите оба слоя (с поляной и цветком), и уменьшите цветок, чтобы он был сопоставим с размерами поляны.

Теперь выделите плоскость и назначьте ей еще одну систему частиц типа «волосы» (Рисунок 12).

В настройках этой системы выберите тип частиц «Hair». Количество частиц установите равным 100 (чтоб не было слишком густо), во вкладке Render выберите тип отображения частицы Object, ниже в соответствующем поле выберите объект «цветок» и

поставьте галочки на Rotate и Scale, чтоб при вращении и масштабировании исходного

	View Search	All Scenes	1	5
	-		※ 封	2
		J	Averaliser	
a . e	- O Lineace			
	ретон)			
-	storm			
1000				
Locatio	a .	Rotation:		Sale
Locatio	0.07126	Rotation:	0* +	× X

Рисунок 11.



Рисунок 12.

цветка изменялись и цветки-частицы. Цветы будут расти вверх от поверхности полянки, если цветок-образец будет расположен параллельно оси X, головкой направо. Теперь изменением параметров Size и Random Сделайте так, чтобы цветы высовывались из травы и были слегка разного размера (Рисунок 13).



Рисунок 13.

Второй уровень сложности выполнения работы.



Рисунок 14.

7. Управление расположением частиц на эмиттере при помощи Vertex Group.

7.1. Выделение области эмиттера, на которой будут расположены частицы. Для того, чтобы ромашки росли только на определенных участках полянки, выделим полянку перейдем в вид сверху и в режим редактирование (Edit Mode), снимем выделение (латинская A) и выделением областью окружностью (латинская C и зажатая Левая кнопка мыши) выделим те места полянки, где будут расти ромашки. Для выхода из режима выделения областью надо нажать Правую кнопку мыши.

7.2. Сохранение группы вершин (Vertex Group).

Для этого надо добавить плюсом группу вершин и назначить ею выделенные вершины, нажав Assign. Группа вершин Group сохранена (Рисунок 15). 7.3. Использование группы вершин в настройках частиц.

В настройках системы частиц во вкладке «Группы вершин (Vertex Groups)» выберем созданную нами

		New	Simula	1 International
		None	Simple	Interpolated
S S Plane + V Plane		► Field Weights		
V Plane	E	► Force Field Settings		
V Normals		Vertex Groups		
Auto Smooth 🚺 Double Sided		(Density:	Group	
Angle: 160*		Length		
Texture Space		Clump:	oup -	
Vertex Groups		Kink:		
Sec. Grap	-	Roughness 1:		
		Roughness 2:		
		Roughness End:		
• -		► Custom Propertie		
Assign Remove Select De	select			

Рисунок 15.

Рисунок 16.

группу Group в поле параметра «Density» (Рисунок 16). В результате частицы-цветы появятся только на участках эмиттера, соответствующих группе вершин Group.

Третий уровень сложности выполнения работы.



Рисунок 17.

Таким способом можно сделать системы частиц для разных цветов и при помощи групп вершин расположить их по полянке (Рисунок 18).



Рисунок 18.

8. Сохранение результата. Сохраните рендер в формате JPEG. Сделайте выводы о проделанной работе.

ЛАБОРАТОРНАЯ РАБОТА «MICROSOFT WORD. ФОРМАТИРОВАНИЕ ТЕКСТА»

Куликовская Лилия Викторовна ГПОУ «Донецкий промышленноэкономический колледж», преподаватель компьютерных дисциплин, квалификационной категории «специалист высшей категории»

Цель: приобрести практические привычки оформления текста с помощью разных операций форматирования.

Задача

- 1. Создать новый документ и загрузить у него текст «Шрифт».
- 2. Выполнить операции форматирования абзацев, символов.
- 3. Оформить страницу.
- 4. Сохранить документ.

Контрольные вопросы.

- 1. Что означает «форматирование текста»?
- 2. Какие операции можно выполнить при форматировании символов?
- 3. Как это можно выполнить?
- 4. Как можно выполнить форматирование абзацев?
- 5. Какие операции относятся к форматированию страницы?
- 6. Как их можно выполнить?

Ход работы:

- 1. Запустить текстовый процессор WORD.
- 2. Выполните правила начала работы с новым документом. Сохраните новый документ под именем Ваша фамилия +Л.р.5; установите поля: верхнее 1,5 см, нижние 1,5 см, правое 1,5 см, левое 2 см.
- 3. На диске Е: откройте в папке «Работа с текстом» документ «Шрифт» (Приложие 1). Выполните это, не закрывая созданный документ
- 4. Скопируйте весь текст в документ Л. р5 и внимательно прочитайте его.
- 5. Выполните следующие операции форматирования со всеми абзацами: Установите:
 - 5.1. абзацное отступление 1,5 см (для третьего абзаца - 1 см с помощью маркера на линейке форматирования);
 - 5.2. междустрочное расстояние полуторное;
 - 5.3. интервал после каждого абзаца 4 пункта;
- 6. Выровняйте абзацы согласно текста (документ «Шрифт»):
 - 6.1. 1,2,3 по ширине;
 - 6.2. 4 по центру;
 - 6.3. 5 по левому краю;
 - 6.4. 6 по правому краю.
- 7. Для второго абзаца установите:

- 7.1. рамка объемная;
- 7.2. тип обрамления линия с тенью;
- 8. Выполните следующие операции форматирования символов:
 - 8.1. установите тип и изображение шрифта согласно тексту каждого абзаца;
 - 8.2. размер символов и цвет в названии типа шрифта (например, Arial) выберите на свое усмотрение. В каждом случае выбирайте разный цвет и размер
- 9. Установите обрамление всей страницы:
 - 9.1. тип рамка;
 - 9.2. рисунок по выбору;
 - 9.3. заливка по выбору;
- 10. Покажите преподавателю.
- 11. Закройте документ.
- 12. Выключите компьютер.

Используемая литература:

1. Симонович З. В., Евсєєв Р. А. Алексеев А. Г. Специальная информатика: Учебное пособие. - Г.: АСТ-ПРЕС: Інфорком - Пресс, 2002. 56-65 з.

Приложение 1

Этот абзац имеет выравнивание по ширине. Оформлен он обычным начертанием шрифта Arial. Таким способом оформляют абзацы книг, газет и т.п. Вид привычный, но в тоже время не наиболее выразительный для специального оформления. Для абзаца устанавливаются расстояния от его левой границы до границы полосы (втяжка слева), от правой границы (втяжка справа), расстояние между предыдущим абзацем и данным (отбивка сверху), между данным абзацем и последующим (отбивка снизу).

Отступ для первой строки, иначе называемый абзацным отступом, может устанавливаться положительным, как для данного абзаца. Оформлен он начертанием Курсив того же шрифта Arial.

Так и отрицательным, иначе называемым выступом (или, иногда, используется термин «форточка», заимствованный из полиграфии), как для этого абзаца. Чтобы задать выступ первой строки, обычно нужно установить отступ для всех строк, а для первой значение абзацного выступа принять равным отрицательной величине. Выступ должен быть меньше втяжке всех остальных строк (т.е. первая строка не может быть левее границы абзаца). Для данного абзаца установлена также втяжка справа. Оформлен он обычным жирным начертанием того же шрифта Arial.

Этот абзац имеет выравнивание по центру. Обычно так оформляют заголовки. Не пытайтесь выставить текст по центру за счет добавления пробелов: при смене размеров листа бумаги, для различных шрифтов нужно подбирать число пробелов. А выключка по центру всегда будет располагать текст по центру полосы набора. Текст, выключенный по центру, лучше смотрится, если в нем запрещены переносы слов. Оформлен он одновременно жирным с курсивом начертанием того же шрифта Arial.

Данный текст имеет выравнивание по левому краю. Этот тип оформления характерен для иностранной документации. Большинство писем и документов, выполненных на обычных пишущих машинках, также имеет тип выключки влево. При оформлении документов целесообразно использовать этот вариант для оформления писем (на формат целого листа бумаги), а также для оформления текста в несколько узких колонок (не более 3-4 на полосу). Оформлен этот абзац начертанием шрифта Courier New Cyr.

Выравнивание по правому краю. Так располагают подписи, а также специальные элементы оформления. Если Вы оформляете текст, то учитывайте, что культура оформления не в последнюю очередь зависит от разнообразия оформительских приёмов, использованных в одном документе: примечание большого количества шрифтов, различных вариантов оформления абзацев считается плохим вкусом. Особенно это надо учитывать тем пользователям, кто, начав работать на компьютере, испытывает удовольствие от возможности оформить текст с помощью большинства имеющихся в наличии шрифтов. Этот абзац оформлен жирным начертанием с подчеркиванием шрифта Courier New Cry.

Абзацная линейка, которая отражается по команде Вид – Линейка, позволяет менять отступы и величину красной строки. Верхний левый маркер меняет положение красной строки, средний меняет левый абзацный отступ, нижний сдвигает красную строку и левый абзацный отступ вместе. Правый маркер служит для изменения правого отступа.

«ПРИМЕРЫ ИНСТРУКЦИЙ И ЗАДАНИЙ К ЛАБОРАТОРНЫМ РАБОТАМ ПО ДИСЦИПЛИНЕ «ИНФОРМАТИКА И ИКТ». РАЗДЕЛ ПРОГРАММИРОВАНИЕ»

Лебединцев Виктор Иванович ГПОУ «Торезский горный техникум им. А.Ф.Засядько» преподаватель информатики, специалист первой квалификационной категории.

Лабораторная работа №

Тема: «Программирование линейных алгоритмов» Цель: овладеть практическими навыками составления линейных программ на языке Паскаль и выполнения их на компьютере. Оборудование: ПК IBM PC

Теоретические сведения

Тело любой программы можно рассматривать как последовательность операторов. Операторы – это описание каких-либо действий с переменными и константами. Программа, в которой операторы выполняются последовательно один за другим, називается линейной. В языке Паскаль операторы отделяются друг от друга точкой с запятой («;»). Составной оператор – это последовательность операторов, заключенных в операторные скобки Begin и End. Begin – это не оператор, поэтому после него точка с запятой не ставится.

Для программирования линейного вычислительного процесса необходимо знать операторы присваивания, ввода и вывода.

Оператор присваивания служит для вычисления значения выражения и присваивания этого значения переменной. Например, X:= A+B;

$$X := 5 + SIN(X);$$

 $Z := 2 * X + 3/Y - 3;$

При выполнении оператора присваивания сначала вычисляется значение выражения, записанного справа, а затем вычисленное значение присваивается переменной, записанной слева.

Оператор ввода. Ввод информации осуществляется с помощью операторов:

READ (b1, b2, ..., bn); READLN (b1, b2, ..., bn); READLN;

где **b1**, **b2**, ..., **bn** – имена переменных, значения которых вводятся. Оператор **READLN**, в отличие от **READ**, кроме ввода данных обеспечивает переход к началу новой строки.

Оператор вывода. Для вывода информации используются операторы:

WRITE (b1, b2, ..., bn); WRITELN (b1, b2, ..., bn); WRITELN;

где **b1, b2, ..., bn** – имена переменных, значения которых выводятся. Оператор **WRITELN** обеспечивает переход к новой строке после ввода последнего значения.

Пример составления линейной программы.

<u>Задача.</u> Составить программу вычисления площади треугольника, если известны три его стороны.

При составлении программы будем использовать формулу Герона

 $S = \sqrt{p(p-a)(p-b)(p-c)}$, где p – полупериметр треугольника; p=(a+b+c)/2.

Текст программы:

```
Program Treugolnik;
Var A,B,C,P,S:Real;
Begin
Write ('Введите длины сторон А, В, С');
Readln (A, B, C);
P:=(A+B+C)/2;
S:=SQRT(P*(P-A)*(P-B)*(P-C));
Writeln ('Площадь треугольника S=', S:6:2)
End.
```

Блок - схема алгоритма



<u>Ход работы</u>

- 1. Инструктаж по технике безопасности.
- 2. Получить задание.
- 3. Составьте программу и блок-схему решения задачи.

4. Введите текст программы в компьютер, откомпилируйте его, получите результат, покажите его преподавателю и запишите в отчет.

5. Сохраните программу на рабочем диске.

6. Выйдите из системы программирования

Контрольные вопросы

- 1. Что в программировании понимается под термином «оператор»?
- 2. Какая программа называется линейной?
- 3. Для чего предназначен оператор присваивания?
- 4. С помощью каких операторов осуществляется ввод и вывод информации?
- 5. В чём разница между операторами Read и Readln; Write и Writeln?

Задания к теме: «Линейные программы» Карточка №1

- 1. Составить программу, определяющую объем цилиндра, если известны его высота h и радиус основания r (V= $\pi r^2 h$)
- 2. Составить программу определения пути равноускоренно движущегося тела,

если известны начальная скорость тела V₀, ускорение а и время t (Путь S=V₀t+ $\frac{at^2}{2}$)

3. Составить программу вычисления функции $y=2\sin|z|+\cos z$, где $z=\ln x^2$. Исходные данные x=2,7

Карточка №2

1. Составить программу, вычисляющую площадь трапеции, если известны её основания и высота. (Площадь трапеции $S = \frac{a+b}{2} \cdot h$, где а и в – основания

трапеции, h – высота)

2. Составить программу нахождения площади треугольника по известным сторонам и углу между ними (S= $\frac{1}{2}ab\sin\gamma$)

3. Составить программу вычисления функции $y=\sin^2|z|$, где $z=\ln(x+2)$. Исходные данные x=1

Карточка №3

1. Определить работу постоянного тока при напряжении U=5 В. Значения силы тока I и времени t вводятся с клавиатуры (A=IUt)

2. Составить программу, определяющую кинетическую энергию тела Ek,

движущегося со скоростью V, если его масса равна m ($E_k = \frac{mv^2}{2}$)

3. Составить программу для вычисления функции $y=\ln|z+x^2|$,

где
$$z = \frac{\sqrt{a+b}}{2}$$
. Исходные данные: a=3,1; b=4,5

Карточка №4

1. Составить программу, которая определяет длину окружности по известному радиусу ($l = 2\pi R$, где l – длина окружности, R – радиус)

2. Составить программу, которая определяет ускорение свободного падения для планеты массой М и радиусом R ($g_n = G \frac{M}{R^2}$, где константа G=6,67·10⁻¹¹ $\frac{H \cdot M^2}{\kappa c^2}$. Например, для Земли Мз $\approx 6 \cdot 10^{24}$ кг, R3 $\approx 6,4 \cdot 10^6$ м; для Марса Мм $\approx 6,4 \cdot 10^{23}$ кг, Rм $\approx 3,4 \cdot 10^6$ м для Юпитера Мю $\approx 1,9 \cdot 10^{27}$ кг, Rю $\approx 70,07 \cdot 10^6$ м)

3. Составить программу вычисления функции $z=x\ln y^2$, где $y=\sqrt{\sin^2 x}$. Значение х ввести с клавиатуры

Карточка №5

1. Составить программу, определяющую площадь ромба, если известны длины его диагоналей. (Площадь ромба $S = \frac{1}{2} d_1 d_2$, где d_1 и d_2 – длины диагоналей ромба) 2. Составить программу определения пути равноускоренно движущегося тела, если известны начальная скорость тела V₀, ускорение а и время t (Путь S=V₀t+ $\frac{at^2}{2}$)

3. Составить программу для вычисления функции

$$z = \sqrt{\frac{5+x}{x+3}}$$
, где $x = e^{|2a|}$ Значение $a = 3,2$

Карточка №6

1. Составить программу, которая определяет внутреннюю энергию одноатомного идеального газа U, если давление газа p=0,1 Па. Объем газа вводится с клавиатуры.

 $(U = \frac{3}{2}pV)$

2. Составить программу, определяющую объем пирамиды по известной площади основания и высоте (V= $\frac{1}{2}Sh$, где S – площадь основания, h- высота трапеции)

3. Составить программу вычисления функции $y=\sin^2|z|$, где $z=\ln(x+2)$. Исходные ланные x=1

Карточка №7

1. Составить программу, определяющую длину средней линии трапеции по известным основаниям (средняя линия трапеции равна полусумме оснований)

2. Составить программу, определяющую потенциальную энергию тела массой т, поднятого над Землей на высоту h (Потенциальная энергия Ep=mgh, где g=9,8 м/c²) 3. Составить программу вычисления функции $z=\sin x^2+\cos y^2$,

где $y=\sqrt{x+2}$. Значение х ввести с клавиатуры.

Карточка №8

1.Составить программу, которая определяет среднее арифметическое трех введенных с клавиатуры чисел.

2. Составить программу нахождения энергии заряженного конденсатора, емкостью 800 пФ ($800 \cdot 10^{-12}$ Ф), если напряжение между его обкладками составляет U=2B (W= $\frac{CU^2}{2}$)

3. Составить программу вычисления функции $z=x^2+e^{y+2}$, где $y=\frac{\sin a}{2,5}$ Исходные

данные: x=5; a=3.

Карточка №9

1. Составить программу определения силы, действующей на тело массой m, которое движется с ускорением а (По II закону Ньютона F=ma)

2. Составить программу нахождения площади треугольника по известным сторонам

и углу между ними (Площадь треугольника $S = \frac{1}{2}ab\sin\gamma$, где а и в – стороны

треугольника, ү- угол между этими сторонами)

3. Составить программу вычисления функции z=(tgy+2)·2^x,

где у= $\sqrt{x+2}$. Значение x ввести с клавиатуры.

Карточка №10

1. Составить программу, определяющую площадь прямоугольника по известным сторонам (Площадь прямоугольника вычисляется по формуле S=ab, где а и b – стороны прямоугольника)

2. Составить программу определения периода колебаний Т математического маятника длиной *l*. ($T = 2\pi \sqrt{\frac{l}{g}}$, где ускорение свободного падения g=9.8 м/c²)

3. Составить программу вычисления функции $y = 2\sin|z| + \cos z$, где $z = \ln x^2$. Исходные данные: x=2,7

Карточка №11

1. Составить программу, определяющую площадь параллелограмма по известным основанию а и высоте h (S=ah)

2. Составить программу пересчета температур. Пользователь должен вводить температуру в градусах по Цельсию, а программа должна выдавать эту же температуру в градусах по Фаренгейту и Кельвину (Если С – температура по

Цельсию, F - по Фаренгейту, K – по Кельвину, то K=C+273; F= $\frac{9}{5}C$ + 32)

3. Составить программу вычисления функции $z=\sqrt{\sin x^2 + 1}$, где $x=\ln|y|$. Значение y=1,5

Карточка №12

1. Составить программу, вычисляющую площадь трапеции, если известны её основания и высота. (Площадь трапеции $S = \frac{a+b}{2} \cdot h$, где а и в – основания трапеции, h - высота)

2. Составить программу, определяющую длину гипотенузы прямоугольного треугольника по известным длинам катетов (Из теоремы Пифагора следует, что длина гипотенузы с= $\sqrt{a^2 + b^2}$)

3. Составить программу вычисления ϕ ункции z=2cosx²+у, где y=tgx. Значение x ввести с клавиатуры

Карточка №13

1. Составить программу, определяющую центростремительное ускорение тела *а*_{*u*}, движущегося со скоростью v по окружности радиусом R (Центростремительное ускорение $a = \frac{v^2}{R}$)

2. Составить программу определения площади треугольника S со сторонами a, b, c. (По формуле Герона площадь треугольника $S = \sqrt{p(p-a)(p-b)(p-c)}$, где

полупериметр
$$p = \frac{a+b+c}{2}$$
)

3. Составить программу вычисления функции $y=2\sin|z|+\cos z$, где $z=\ln x^2$. Исходные данные x=2,7

Карточка №14

1. Составить программу, определяющую силу тяжести FT, действующую на тело массой т вблизи поверхности Земли (FT=mg, где ускорение свободного падения g $=9.8 \text{ M/c}^2$)

2. Составить программу, вычисляющую площадь полной поверхности цилиндра с высотой h и радиусом основания r. ($S=2\pi r(r+h)$)

3. Составить программу вычисления функции $y=\sin^2|z|$, где $z=\ln(x+2)$. Исходные данные x=1,5

Карточка №15

1. Составить программу, определяющую площадь боковой поверхности конуса S, если известны его образующая *l* и радиус основания r (S= πrl)

Составить программу, вычисляющую механическую работу силы F, под 2. действием которой тело совершило перемещение S, если угол между направлением действия силы и перемещением равен α (Механическая работа A=Fscos α)

3. Составить программу вычисления функции $z = \sqrt{x + 2y}$, где $y = \frac{x}{2.5}$. Исходные

данные: x=4

Карточка №16

1. Составить программу, определяющую импульс тела массой m, движущегося со скоростью *v*. (Импульс тела p=mv)

2. Составить программу нахождения площади сферы радиусом R. (S = $4\pi R^2$)

3. Составить программу вычисления функции y=tgz+5, где z=|x+2|. Значение x ввести с клавиатуры

Карточка №17

1. Составить программу, определяющую силу тока I в однородном проводнике, обладающим сопротивлением R, если напряжение на его концах равно U. (По закону $I = \frac{U}{R}$) Ома

2. Составить программу, вычисляющую площадь полной поверхности конуса с образующей *l* и радиусом основания г (Площадь полной поверхности конуса $S = \pi r(l+r)$)

3. Составить программу вычисления функции $z = \frac{x^2}{2v}$, где $y = \cos(2x)$. Значение x

ввести с клавиатуры.

Карточка №18

1. Составить программу, вычисляющую объем пирамиды V, имеющей площадь основания S и высоту h. (Объем пирамиды V= $\frac{1}{2}Sh$)

2. Составить программу определения работы постоянного тока А при напряжении U=5B, силе тока I и времени прохождения тока t (Работа постоянного тока A=IUt)

3. Составить программу вычисления функции $y=z^2$, где $z=\ln|x+2|$. Значение x ввести с клавиатуры.

Карточка №19

1. Составить программу, определяющую мощность постоянного тока при силе тока I и напряжении U (P=IU)

2. Составить программу нахождения объема шара радиусом R. (Объем шара V= $\frac{4}{3}\pi R^3$)

3. Составить программу вычисления функции $y=2\sin|z|$, где $z=x^2$. Исходные данные: x=5.2

Карточка №20

1. Составить программу, вычисляющую площадь трапеции, если известны её

основания и высота. (Площадь трапеции $S = \frac{a+b}{2} \cdot h$, где а и в – основания

трапеции, h – высота)

2. Составить программу, вычисляющую количество теплоты Q, выделившееся в однородном проводнике сопротивлением R=7 Ом, силе тока I=5 А за время t (По закону Джоуля-Ленца количество теплоты Q=I²Rt)

3. Составить программу вычисления функции $y=\sqrt{z^2+5}$, где $z=e^x$. Исходные ланные: x=2

Карточка №21

1. Составить программу, определяющую объем цилиндра, если известны его высота h и радиус основания r (V= $\pi r^2 h$)

2.Составить программу определения периода свободных электромагнитных колебаний Т в контуре без активного сопротивления при индуктивности L и емкости C. (По формуле Томсона $T=2\pi\sqrt{LC}$)

3. Составить программу вычисления функции у= $2\cos(z-2)$, где $z=\sqrt{3x}$. Значение x=5,1

Карточка №22

1.Составить программу, которая определяет среднее арифметическое трех введенных с клавиатуры чисел.

2.Составить программу, определяющую собственную циклическую частоту ω_0 свободных электромагнитных колебаний в контуре с индуктивностью L и емкостью

С. (Собственная циклическая частота $\omega_0 = \frac{1}{\sqrt{LC}}$).

3. Составить программу вычисления функции $z=e^{x+y}$, где $y=\sqrt{\frac{x}{2,5}}$. Значение х

ввести с клавиатуры.

Карточка №23

1. Составить программу, которая определяет длину окружности по известному радиусу ($l = 2\pi R$, где l – длина окружности, R – радиус)

2. Составить программу определения периода колебаний Т математического

маятника длиной *l*. ($T = 2\pi \sqrt{\frac{l}{g}}$, где ускорение свободного падения g=9.8 м/с²)

3. Составить программу для вычисления функции $y=z^2$, где $z=\ln|x+2|$. Значение х ввести с клавиатуры.

Карточка №24

1. Составить программу, определяющую площадь ромба, если известны длины его диагоналей. (Площадь ромба $S = \frac{1}{2}d_1d_2$, где d_1 и d_2 – длины диагоналей ромба)

2. Составить программу, вычисляющую механическую работу силы F, под действием которой тело совершило перемещение S, если угол между направлением действия силы и перемещением равен α (Механическая работа A=Fscos α)

3. Составить программу вычисления функции z=2cos(x²+y), где y=ctgx. Значение х ввести с клавиатуры

Карточка №25

1. Составить программу, определяющую импульс тела массой m, движущегося со скоростью v. (Импульс тела p=mv)

2. Составить программу, вычисляющую площадь полной поверхности конуса с образующей l и радиусом основания г (Площадь полной поверхности конуса $S = \pi r(l+r)$)

3. Составить программу вычисления функции $y = \sqrt{z^2 + 5}$, где $z = \frac{e^x}{2}$. Исходные

данные: х=3

Карточка №26

1. Составить программу, определяющую потенциальную энергию тела массой m, поднятого над Землей на высоту h (Потенциальная энергия Ep=mgh, где g=9,8 м/c²) 2. Составить программу нахождения площади сферы радиусом R. (S = $4\pi R^2$)

3. Составить программу вычисления функции $z=\sqrt{\cos x^2 + 1}$, где $x=\ln|y|$. Значение y=1,5

ПРАКТИЧЕСКАЯ РАБОТА «РАЗРАБОТКА ИНФОЛОГИЧЕСКОЙ МОДЕЛИ И СОЗДАНИЕ СТРУКТУРЫ РЕЛЯЦИОННОЙ БАЗЫ ДАННЫХ. ФОРМИРОВАНИЕ СЛОЖНЫХ ЗАПРОСОВ»

Проскокова Ольга Николаевна ГПОУ «Амвросиевский индустриальноекономический колледж» преподаватель информатики и спецдисциплин, зав. лабораторией специалист высшей квалификационной категории

I. 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 года рождения. Но если студенты, имеющие одинаковые фамилии, имеют разные года 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 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 [2].

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-комногим* 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 [2].

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, заработная плата).

Рассмотрим связь между объектами *Студенты* и *Дисциплины*. Студент изучает несколько дисциплин, что соответствует многозначной связи и отражено на рис. 3.1 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).



Рисунок 1 - Типы связей между объектами Студенты, Дисциплины и

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 оценки. Каждый студент имеет оценки по нескольким дисциплинам, поэтому связь между объектами *Студенты* и *Оценки* будет *Один-ко-многим* (1:М). Каждую дисциплины и *Оценки* также будет *Один-ко-многим* (1:М). В результате получаем информационно-логическую модель базы данных, приведенную на рис. 3.2.



Рисунок 2 – Информационно – логическая модель реляционной базы даннях

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 рис. 3.3, где жирными буквами выделены ключевые поля.



Рисунок 3 - 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 db_l, 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 табл. 1;

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

Таблица 1 – Структура записи таблицы «Студенты»

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 табл. 2.

Таблица 2 - Структура записи таблицы «Дисциплины»

Имя поля	Тип данных	Размер поля
Код дисциплины	Чисповой	Цепое
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, поэтому для работы используйте эту таблицу с одним лишь изменением – в соответствии с рис. 3.5 в структуру таблицы надо добавить поле «Код дисциплины» и заполнить его в соответствии с данными табл. 3.2.

5. Создайте структуру таблицы *Оценки* аналогично п. 2 в соответствии с табл. 3. Таблица 3 – Структура записи таблицы «Оценки»

Имя поля	Тип данных	Размер поля
Код студента	Чисповой	Целое
Код дисциплины	Чисповой	Целое
Оценки	Чисповой	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 удалятся 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. 4;



Рисунок 4. Схема данных

ЗАДАНИЕ 3. 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 изменяя 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 табл. 4, таблицу Студенты посредством формы.

Код	Фамилия	Имя	Отчество	Нотер	Телефон	Стипендия
студента				труппы		
1	Арбузов	Николай	Николаевич	151	260-15-63	Да
2	Киршин	Петр	Валерьевич	151	110-67-82	Да
3	Кривинский	Сергей	Николаевич	151	172-97-21	Нет
4	Крылова	Елена	Петровна	151	130-31-87	Да
5	Kyme ynn	Григорий	Викторович	151	269-53-75	Да
6	Патрижев	Omer	Борисович	152	234-11-63	Нет
7	Перлов	Кирипп	Николаевич	152	312-21-33	Нет
8	Соколова	Наталия	Петровна	152	166-87-24	Нет
9	Степанская	Ольга	Витальевна	152	293-43-77	Да
10	Тимофеев	Сергей	Трофимович	152	260-11-57	Да

Таблица 4 Студенты

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, приведенными в табл. 5, таблицу Дисциплины посредством, формы и закройте форму, задав ей имя Дисциплины.

5. Создайте форму Дисциплины аналогично п. 1.

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

Таблица 5 Дисциплины

Код дисциплины	Название дисциплины
1	Информатика
2	Математика
3	Физика
4	Экономика

Таблица 6 Оценки

Код	Код	Оценки	Код	Код	Оценки
студента	дисциплины		студента	дисцоплоны	
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	I	5
5	2	5	10	2	5
5	3	5	10	3	5
5	4	5	10	4	5

II. ФОРМИРОВАНИЕ СЛОЖНЫХ ЗАПРОСОВ

В *перекрестном запросе* отображаются результаты статистических расчетов (такие, как суммы, количество записей, средние значения), выполненных по данным из одного поля таблицы. Эти результаты группируются по двум наборам данных, один из которых расположен в левом столбце таблицы, а второй - в верхней строке. Например, нам надо узнать средний стаж работы ассистентов, доцентов и профессоров на разных кафедрах (на основе таблицы *Преподаватели*). Перекрестный запрос позволит легко решить эту задачу, создав таблицу, в которой заголовками строк будут служить должности, заголовками столбцов - названия кафедр, а в ячейках будут рассчитаны средние значения стажа преподавателей.

Запрос на изменение - это запрос, который за одну операцию вносит изменения в несколько записей. Существует четыре типа запросов на изменение; на удаление, обновление и добавление записей, а также на создание таблицы.

Запрос на удаление удаляет группу записей, удовлетворяющих заданным условиям, из одной или нескольких таблиц. С помощью запроса на удаление можно удалять только всю запись, а не отдельные поля внутри нее.

Запрос на обновление записей вносит общие изменения в группу записей одной или нескольких таблиц. Например, на 10 процентов увеличилась заработная плата ассистентов, Запрос на обновление позволит быстро внести эти изменения в таблицу Преподаватели.

Запрос на добавление добавляет группу записей из одной или нескольких таблиц в конец одной или нескольких таблиц. Например, появилось несколько новых преподавателей, а также база данных, содержащая сведения о них. Чтобы не вводить все данные вручную, их можно добавить в таблицу Преподаватели.

Запрос на создание таблицы создает новую таблицу на основе всех или части данных из одной или нескольких таблиц. Например, на основе таблицы Преподаватели можно создать новую таблицу, содержащую данные только о профессорах.

Задание

1 Разработайте запрос с параметрами о студентах заданной группы, в котором при вводе в окно параметров номера группы (в примере это 151 или 152) на экран должен выводиться состав этой группы.

2 Создайте запрос, в котором выводятся оценки студентов заданной группы по заданной дисциплине.

3 Создайте перекрестный запрос, в результате которого создаєтся выборка, отражающая средний балл по дисциплинам в группах.

4 Разработайте запрос на увеличение на 10% заработной платы тех преподавателей, кто получает менее 500 р.

5 Создайте запрос на удаление отчисленных студентов.

6 Разработайте запрос на создание базы данных отличников.

7 Для всех созданных вами запросов разработайте формы.

Технология работы

1 Для создания запроса с параметрами о студентах заданной группы:

- откройте вкладку Запросы',

- щелкните по кнопке <Создать>;

- в появившемся окне выберите *Простой запрос* и щелкните по кнопке <OK>;

- в окне в строке Таблицы/запросы выберите из списка таблицу Студенты;

- перенесите все поля из окна "Доступные поля" в окно "Выбранные поля";

- щелкните по кнопке <Далее>. Выводить надо все поля, поэтому еще раз щелкните по кнопке <Далее>;

- в появившемся окне введите имя запроса Группа;

- щелкните по кнопке <Готово>. На экране появится таблица с данными запроса. Но вам надо, чтобы при выполнении запроса выяснялся номер группы. Для этого перейдите в режим конструктора;

- в строке *Условия отбора* для поля "Номер группы" введите фразу (скобки то же вводить): *[Введите номер группы]*;

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

- в появившемся окне введите *151* и щелкните по кнопке <OK>. На экране появится таблица с данными о студентах 151-й группы;

- сохраните запрос и закройте таблицу запроса.

2 Для создания запроса, в котором выводятся оценки студентов заданной группы по заданной дисциплине:

- на вкладке Запросы щелкните по кнопке «Создать»;

- выберите *Простой запрос* и щелкните по кнопке <OK>;

- выберите таблицу *Студенты* и перенесите поля "Фамилия", "Имя", "Отчество", "Номер группы" в окно "Выделенные поля" (выделяя нужное поле и щелкая по кнопке).

Внимание! В дальнейшем под фразой В таблице ... выберите поле ... будем понимать выбор таблицы, выбор поля и перенос его в окно "Выделенные поля".

- в таблице Дисциплины выберите поле "Название дисциплины";

- в таблице *Оценки* выберите поле "Оценки". Вы сформировали шесть полей запроса - они связаны между собой посредством схемы данных;

- щелкните по кнопке <Далее>;

- в появившемся окне введите имя запроса *Оценки группы*, затем щелкните по ячейке *Изменение структуры запроса* (в ней должна появиться черная точка) - это позволит сразу перейти в режим конструктора;

- щелкните по кнопке <Готово>;

- в строке Условия отбора для поля "Номер группы" введите фразу. [Введите номер группы],

- в строке Условия отбора для поля "Название дисциплины" введите фразу: [Введите название дисциплины]

- выполните запрос;

- в первом появившемся окне введите *152*, затем щелкните по кнопке <OK>, во втором - введите *Информатика* и щелкните по кнопке <OK>. На экране появится таблица со списком 152-й группы и оценками по информатике;

- сохраните запрос и закройте таблицу запроса.

3 Создайте перекрестный запрос о среднем балле в группах по дисциплинам. Но такой запрос строится на основе одной таблицы или одного запроса, в связи, с чем надо сначала сформировать запрос, в котором были бы поля "Номер группы", "Название дисциплины" и "Оценки". Для этого:

- на вкладке Запросы щелкните по кнопке «Создать»;

- выберите *Простой запрос* и щелкните по кнопке <OK>;

- выберите из таблицы Студенты поле "Номер группы";

- выберите из таблицы Дисциплины поле "Название дисциплины" ;
- выберите из таблицы Оценки поле "Оценки";
- щелкните по кнопке <Далее>;

- в появившемся окне введите имя запроса Дисциплины оценки группы;

- щелкните по кнопке <Готово>;
- сохраните запрос и закройте таблицу запроса.
- Теперь можно создавать перекрестный запрос. Для этого:
- на вкладке Запросы щелкните по кнопке «Создать»;
- выберите Перекрестный запрос и щелкните по кнопке <OK>;
- щелкните по ячейке Запросы, выберите Дисциплины оценки группы и <Далее>;
- поле "Название дисциплины" переведите в окно доступных полей и <Далее>;
- выделите поле "Номер группы" и щелкните по кнопке <Далее>;

- выберите функцию AVG для оставшегося поля «Оценки», т.е. среднее и щелкните по кнопке <Далее>;

- введите название запроса *Средние оценки* и щелкните по кнопке <Готово>. Откроется таблица перекрестного запроса. Обратите внимание на то, что Access создает еще итоговое значение средних оценок по дисциплинам;

- закройте таблицу запроса.

- 4 Для создания запроса на изменение заработной платы преподавателей:
- на вкладке Запросы щелкните по кнопке «Создать»;
- выберите Простой запрос;
- в таблице *Преподаватели* выберите поле <Зарплата>;
- щелкните по кнопке <Далее>;
- в появившемся окне введите имя запроса Изменение зарплаты;
- щелкните по ячейке Изменение структуры запроса;
- щелкните по кнопке <Готово>;
- в строке Условия отбора введите <500;
- откройте пункт меню Запрос и выберите Обновление;
- в строке конструктора запроса Обновление в поле "Зарплата" введите: [Зарплата] 1,1;
- выполните запрос, подтвердив готовность на обновление данных;
- закройте запрос, подтвердив его сохранение;
- откройте форму Преподаватели;
- просмотрите изменение заработной платы у преподавателей, получающих меньше 500 р.;
- закройте форму.
- 5 Для создания запроса на отчисление студента гр.152 Перлова Кирилла Николаевича:
- на вкладке Запросы щелкните по кнопке «Создать»;
- выберите Простой запрос;
- в таблице Студенты выберите поля "Фамилия", "Имя", "Отчество", "Номер группы";
- щелкните по кнопке <Далее>, затем в появившемся окне снова щелкните по кнопке <Далее>;
- в появившемся окне введите имя запроса Отчисленные студенты;

щелкните по ячейке Изменение структуры запроса;

- щелкните по кнопке <Готово>;

- в строке Условия отбора введите; в поле "Фамилия" - Перлов, в поле "Имя" - Кирилл, в поле "Отчество" - Николаевич, в поле "Номер группы" - 152; - откройте пункт меню Запрос и выберите Удаление;

- просмотрите удаляемую запись, щелкнув по кнопке или выполнив команду **Вид, Режим таблицы**; если отчисляемый студент выбран правильно, то перейдите в режим конструктора и выполните запрос. Если условия отбора сделаны неправильно, измените их;

- закройте запрос;

- откройте форму Студенты и удостоверьтесь в удалении записи о студенте Перлове;

- закройте форму.

6 Для создания запроса на создание базы данных отличников:

- на вкладке Запросы щелкните по кнопке «Создать»;

- выберите Простой запрос;

- в таблице *Студенты* выберите поля "Фамилия", "Имя", "Отчество" и "Номер группы", а в таблице *Оценки* - поле "Оценки";

- щелкните по кнопке <Далее>;

- в появившемся окне введите имя запроса Отличники;

- щелкните по ячейке Изменение структуры запроса;

- щелкните по кнопке <Готово>;

Примечание. Для создания этого запроса надо воспользоваться операцией группировки. Будем считать отличниками тех студентов, которые набрали за четыре экзамена 20 баллов. Операция группировки позволит просуммировать оценки студентов по всем экзаменационным дисциплинам.

для выполнения групповых операции щелкните на панели инструментов по кнопке или выполните команду Вид, Групповые операции;

- в строке *Групповые операции* поля "Оценки" щелкните по ячейке *Групповые операции*. Откройте раскрывающийся список и выберите функцию SUM;

- в строке Условия отбора поля "Оценки" введите 20;

- просмотрите создаваемую базу, щелкнув по кнопке или выполнив команду **Вид, Режим таблицы**;

- перейдите в режим конструктора;

- выполните команду Запрос, Создание таблицы;
- введите имя таблицы *Студенты-отличники* и щелкните по кнопке <OK>;
- подтвердите создание таблицы;
- закройте с сохранением запрос;
- откройте вкладку Таблицы;

- откройте таблицу *Студенты-отличники*. Удостоверьтесь в правильности создания таблицы. Закройте таблицу.

7 Для каждого из созданных запросов создайте форму (можно рекомендовать автоформу в столбец или ленточную автоформу) для удобного просмотра данных.

Используемая литература

1. Семакин И.Г. и др. Информатика и ИКТ. Базовый уровень. Учебник для 10 кл. 5 изд. – М. :БИНОМ. Лаборатория знаний, 2015. – 264

2. Семакин И.Г. и др. Информатика и ИКТ. Базовый уровень. Учебник для 11 кл. 5 изд.

– М. :БИНОМ. Лаборатория знаний, 2015. – 224

3. Семакин И.Г. и др. Информатика и ИКТ. Базовый уровень. Практикум для 10-11 кл 5 изд. – М. :БИНОМ. Лаборатория знаний, 2015. – 283

4. Семакин И. Г. Информатика. Задачник-практикум: 8—11 классы: т. т.1, 2 5 изд. – М. :БИНОМ. Лаборатория знаний, 2015. – 215

5. Мартишин С.А., Симонов В.А., Храпченко М.В. Проектирование и реализация баз данных в СУБД MySQL с использованием MySQL Workbench: Учебное пособие. - М.: ИД. "Форум" : ИНФРА - М. 2012-160с

6. Симонович С.В. Информатика базовый курс: Учебник для вузов. 3-е изд. Стандарт 3-го поколения.-Спб.: Питер, 2012-640с.

7. Безручко В. Т. Информатика курс лекций: Учебное пособие. - М.: ИД. "Форум" : ИНФРА - М. 2013-432.c
ПРАКТИЧЕСКАЯ РАБОТА

«ИСПОЛЬЗОВАНИЕ ФУНКЦИЙ MS EXCEL В ПРОФЕССИОНАЛЬНОЙ ДЕЯТЕЛЬНОСТИ» ПО УЧЕБНОЙ ДИСЦИПЛИНЕ «ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ПРОФЕССИОНАЛЬНОЙ ДЕЯТЕЛЬНОСТИ» СПЕЦИАЛЬНОСТИ 38.02.05 ТОВАРОВЕДЕНИЕ И ЭКСПЕРТИЗА КАЧЕСТВА ПОТРЕБИТЕЛЬСКИХ ТОВАРОВ

Авилова Светлана Алексеевна ГПОУ «Шахтёрский техникум» ГО ВПО «Донецкий национальный университет экономики и торговли имени Михаила Туган-Барановского», преподаватель информатики и информационных технологий в профессиональной деятельности, специалист высшей квалификационной категории

В соответствии с требованиями программы профессиональной подготовки специалистов специальности 38.02.05 «Товароведение и экспертиза качества потребительских товаров» будущие специалисты должны владеть информационной культурой, анализировать и оценивать информацию с использованием информационно-коммуникационных технологий.

Основная цель дисциплины «Информационные технологии в профессиональной деятельности» – формирование у будущих специалистов теоретических знаний и практических навыков решения профессиональных задач с использованием современных информационных технологий, развитие умения работы с персональным компьютером на высоком пользовательском уровне, овладение методами и программными средствами обработки деловой информации, навыками работы со специализированными компьютерными программами.

На изучение темы «Электронные таблицы MS Excel» раздела «Технология работы в прикладном программном обеспечении» выделяется 30 часов. Изучение данной темы имеет важное практическое применение в работе товароведа-эксперта.

В методической разработке практического занятия по теме «Использование функций MS Excel в профессиональной деятельности» основное внимание уделено проблеме внедрения современных активных методик обучения, в частности интерактивных методов, таких как деловая игра, презентация мини-проектов, компьютерное тестирование.

Использование программы презентации, сети Интернет при проведении практического занятия содействует повышению у студентов уровня компьютерной грамотности, использованию современных технологий, развивает творческие способности, умение применять теоретический материал в практической деятельности, повышает информационную культуру.

Методическая разработка будет полезной для преподавателей информатики и информационных технологий в профессиональной деятельности для учебных заведений среднего профессионального образования, так как раскрывает современный подход к проведению практического занятия.

Цель работы: Научиться использовать мастер функций MS Excel для решения профессиональных задач

Оборудование: РС ІВМ, подключённый к сети Интернет, мультимедийный проектор

Программное обеспечение: Операционная система Windows, программа MS Excel, программа компьютерной презентации Power Point

Залание:

- 1. Получить индивидуальное задание в сети Интернет на сайте техникума.
- 2. Загрузить Электронную таблицу MS Excel.
- 3. Выполнить необходимые расчёты.
- 4. Оформить отчёт по результатам выполнения задания.

При оформлении отчёта необходимо написать в тетради:

- задание;
- функции, которые используются при решении задания; -
- написать формулы с показателями;
- сформулировать выводы
- 5. Защитить отчёт.

Методические рекомендации по выполнению практической работы

1. Загружаем программу Excel: «Пуск» - «Программы» - «MS Excel»

2. После запуска MS Excel создаём новый документ Excel: «Создать» -«Новая книга» - «ОК» или загружаем имеющийся файл.

3. После появления на экране Рабочей книги, выполняем необходимые расчёты с помощью Мастера функций . Щёлкаем мышью по кнопке Панели инструментов fx. После появления диалогового окна выбираем необходимую категорию функций. Далее выбираем необходимую функцию, заносим данные для расчётов и нажимаем кнопку «ОК».

Пример	выполнения	задания	практической	работы
--------	------------	---------	--------------	--------

Наименование продукции	Цена, руб./ кг
Варёные колбасы	200,37
Сосиски	230,47
Полукопчёные	280,95
Копчёности	360,19
Твёрдокопчёные	400,51
Сыровяленые	450,52
Конские	550,75
Таблина 1 Ассортимент колб	асных излелий

Ассортимент колбасных изделий

Наименование	Январь,	Февраль,	Март,	Апрель,	Май,	Июнь,
продукции	КГ	КГ	КГ	КГ	КГ	КГ
Варёные колбасы	250	234	256	231	223	213
Сосиски	312	290	289	27	261	247
Полукопчёные	300	295	280	287	292	322
Копчёности	72	84	85	89	100	102
Твёрдокопчёные	25	35	36	39	55	57
Сыровяленые	17	20	24	24	28	30
Конские	5	7	9	10	11	10

Таблица 2

Спрос на продукцию

Методика выполнения заданий практической работы

Задание 1:

Предсказать значение спроса для следующего месяца.

Функция ПРЕДСКАЗ() имеет следующий синтаксис:

= ПРЕДСКАЗ(х; известные значения у; известные значения х).

Для определения спроса на июль месяц необходимо установить курсор в ячейку на пересечении вида продукции и спроса на июль месяц, например H4 для варёных колбас, ввести в ячейку формулу:

= ПРЕДСКАЗ(Н3;В4:G4;В3:G3)

В ячейке появляется значение спроса на июль месяц на основании данных за 6 месяцев по данному виду продукции.

Аналогично определяется спрос и по другим видам продукции.

Задание 2

Используя данные Таблицы 2 вычислить минимальный и максимальный удельный вес каждого вида продукции за шесть месяцев.

Для расчёта минимального и максимального удельного веса сначала необходимо определить общее количество продукции, проданной по каждому виду продукции. Для этого вводим формулу, например, для Варёных колбас, в ячейку H2:

=CYMM(B2:G2)

После того, как будут подсчитаны суммы по всем видам продукции, посчитываем общий итог по всем видам продукции. В ячейку Н9 вводим формулу:

=CYMM(H2:H8)

Определяем удельный вес для каждого вида продукции. Нвапример, для Варёных колбас: в ячейку I2 вводим формулу: =H2/H9

Для определения минимального удельного веса среди всей продукции в ячейку I10 вводим формулу: =МИН(I3:I9)

Для определения максимального удельного веса среди всей продукции в ячейку I10 вводим формулу: =MAKC(I3:I9)

Вывод: Минимальный удельный вес продаж за шесть месяцев имеет продукция Конские, он равняется 0,01. Максимальный удельный вес продаж за шесть месяцев имеет продукция Полукопчёные, он равняется 0,3

Задание 3

Построить диаграммы максимальних и минимальных удельных весов для всех видов продукции.

Для построения диаграмм сначала нужно определить минимальные и максимальне удельные веса по всем видам продукции за шесть месяцев, а потом среди этих удельных весов минимальный и максимальный.

Для расчёта минимального удельного веса по определённой продукции необходимо ввести формулу, например для Варёных колбас, в ячейку J2:

= MUH(B2/H2;C2/H2;D2/H2;E2/H2;F2/H2;G2/H2)

Для расчёта максимального удельного веса по определённой продукции необходимо ввести формулу, например для Варёных колбас, в ячейку К2:

= MAKC(B2/H2;C2/H2;D2/H2;E2/H2;F2/H2;G2/H2)

Диаграммы – графическое представление числовых значений. Графики можно создавать как в обычном рабочем листе, помещая её рядом с данными, так и в отдельном листе, который называется листом диаграмм. Сначала нужно указать, какие данные нужно отобразить в виде графика, для этого выделяем диапазон ячеек, в которых находятся выходные данные A4:A7 и B4:B7. Во втором диалоговом окне мастера диаграмм выбираем Тип диаграммы. В третьем диалоговом окне указываем вид выбранной диаграммы. Далее указываем название диаграммы.

Строим диаграмму по столбцу «Минимальный удельный вес». Для этого выполняем следующие действия: Вставка-Диаграмма-График. Указываем диапазон: J2:J8.

Строим диаграмму по столбцу «максимальный удельный вес». Для этого выполняем следующие действия: «Вставка-Диаграмма-График. Указываем диапазон К2:К8.

Задание 4

Найти с помощью функции «Поиск решения» количество каждого вида продукции, которое необходимо продать, чтобы выручка от продаж была максимальной при известной фиксированной цене. При условии, что общий объём продажи равен 2000 кг. А удельные веса каждого вида продукции должны находиться в промежутке между минимальным и максимальным удельным весом.

Вызываем функцию «Поиск решения», которая находиться в режиме «Данные». Устанавливаем целевую ячейку I20, в которой находится значения выручки, равной максимальному значению. Изменяем ячейки P13:P19, в которых находится количество проданной продукции за шесть месяцев. При этом вводим следующие ограничения:

количество проданной продукции не должно равняться 0;

удельный вес каждого вида продукции должен быть > минимального удельного веса и меньше максимального удельного веса.

Задание 5

Найти с помощью инструмента «Подбор параметра» значение цены для полукопчёных колбас, при котором увеличиться прибыль на 3%.

Определяем значение увеличения прибыли для Полукопчёных колбас. Умножаем значение выручки на 0,03. Рассчитываем прибыль для Полукопчёных колбас. Соответственно увеличится и общая сумма прибыли по всем видам продукции.

Вызываем «Подбор параметра» в режиме «Данные»

Устанавливаем в ячейке I20 значение 1579540. Изменяя значения ячейки цены Полукопчёных колбас (B5) находим значение = 289,38

Вывод: Для увеличения прибыли от продажи Полукопчёных колбас на 3%, их цена за килограмм должна равняться 289,38

Задание 6

1. Считается, что копчёности, твердокопченые и сыровяленые колбасы относятся к виду «Деликатесы». Определить процентное соотношение между деликатесами и основной продукцией.

Подсчитываем выручку по Деликатесам. Для этого вводим формулу:

=СУММ(І16:І18).

Получаем значение выручки = 5668,46

Находим соотношение по формуле:

=I21/I20

Получаем 23%. Строим круговую диаграмму.

Задание 7

Дирекция магазина решает начать закупку новых видов продукции. Их цена 170 рублей за килограмм. Проанализировать целесообразность нововведения (прибыль не должна снизиться), если известно, что их минимальный удельный вес -0,05, а максимальный -0,07.

Задание решаем с помощью функции «Поиск решения». Целевой ячейкой устанавливаем ячейку со значеним прибыли. Она должна раняться 1549153. Изменяя значения ячеек, которые содержат информацию о количестве проданной продукции, находим количество проданной продукции с учётом Новой продукции.

При расчёте необходимо учитывать следующие ограничения:

Удельные веса должны быть >0,01; удельный вес по Новой продукции должен быть больше 0,05 и меньше 0,07.

Вывод: В результате решения получаем, что закупка новой продукции не повлияет на увеличение прибыли.

Используемая литература:

 Войтюшенко Н.М. Информатика и компьютерная техника: Уч. пос. баз. подготовки для студ. экон. и тех. специальностей дн. и заоч. форм обучения /Н.М.Войтюшенко, А.И.Остапец. – Донецк: ДонНУЭТ, 2014 – 485 с.

- 2. Михеева Е.В. Информационные технологии в профессиональной деятельности: учеб. пособие для студ. сред. проф. образования. – 9-е изд., стер. – М.: Издательский центр «Академия», 2014. – 256 с.
- Михеева Е.В. Практикум по информационным технологиям в профессиональной деятельности: учеб. пособие для студ. учреждений сред. проф. образования. – 10е изд., стер. – М.: Издательский центр «Академия», 2014. – 192 с.

ПРАКТИЧЕСКАЯ РАБОТА «ФОРМАТИРОВАНИЕ ТЕКСТОВЫХ ФРАГМЕНТОВ НТМL-СТРАНИЦЫ» ПО ДИСЦИПЛИНЕ МДК.01.01. «ТЕХНОЛОГИИ СОЗДАНИЯ И ОБРАБОТКИ ЦИФРОВОЙ МУЛЬТИМЕДИЙНОЙ ИНФОРМАЦИИ» ПМ.01. «ВВОД И ОБРАБОТКА ЦИФРОВОЙ ИНФОРМАЦИИ»

Басова Галина Николаевна, Горловское высшее профессиональное училище, преподаватель профессиональной подготовки по профессии «Мастер по обработке цифровой информации», специалист высшей категории Елисеева Анна Юрьевна, мастер производственного обучения

Цель работы:

– изучить теги для форматирования текста на веб-странице;

– создать веб-страницу, которая содержит абзацы, текстовые заголовки, текст различного начертания, различный формат абзацев, маркированные и нумерованные списки.

НЕОБХОДИМЫЙ ТЕОРЕТИЧЕСКИЙ АСПЕКТ

Особенности текста в HTML

Для изменения вида текста существует достаточно большое количество различных тегов. Это и немудрено, ведь текст самый популярный вид информации.

Прежде чем редактировать код веб-страницы, следует принять во внимание некоторые особенности, которые присущи HTML при работе с текстом.

– Любое количество пробелов идущих подряд, в браузере отображается как один.

Сколько бы вы не поставили пробелов между словами, это никак не повлияет на конечный вид текста. Это же правило относится к символам табуляции и переносу текста. Поэтому не ставьте лишних пробелов, поскольку это лишь увеличит общий объем файла, но никак не изменит вид документа в браузере. Приведенные ниже строки будут отображаться на веб-странице одинаково, несмотря на их разное написание.

<**p>**Один. Два. Три.</**p**> <**p**> Один. Два. Три.</**p**> <**p**> Один. Два. Три.</**p**>

Исключением из этого правила является тег , внутри которого любое число пробелов отображается именно так, как оно указано в коде.

Нет расстановки переносов в тексте

HTML не поддерживает расстановку переносов в словах, как это делают текстовые редакторы, иначе говоря, все слова пишутся целиком без их разбиения. Это

условие несущественно, пока не используется выравнивание текста по ширине. В этом случае блок текста выравнивается по левому и правому краю. Короткие строки при этом растягиваются за счет автоматического добавления пробелов между словами. Иногда пустые блоки между словами настолько велики, что портят внешний вид страницы и ухудшают читабельность текста.

Текст занимает ширину окна браузера

Если вы просто напишете одну длинную строку в коде HTML, то в браузере она будет отформатирована, чтобы текст поместился по ширине в окно. Переносы текста будут добавлены автоматически в местах пробела или дефиса. Что произойдет, если в тексте нет ни того, ни другого символа? Браузер не сможет создать переносы и отобразит текст одной строкой. Если она шире окна браузера, то неминуемо появится горизонтальная полоса прокрутки [1].

Формат абзацев и переноса строки

Как правило, блоки текста разделяют между собой абзацами (параграфами). По умолчанию между параграфами существует небольшой вертикальный отступ, называемый отбивкой. Синтаксис создания абзацев следующий.

Абзац 1

Абзац 2

Каждый абзац начинается с тега и заканчивается необязательным закрывающим тегом .

В любой книге для выделения следующего абзаца используется отступ первой строки, еще называемый «красная строка». Это позволяет читателю легко отыскивать взглядом новую строку и повышает, таким образом, читабельность текста. На вебстранице этот прием обычно не используется, а для разделения абзацев применяется отбивка.

При использовании тега между абзацами возникают слишком большие отступы. От них можно избавиться, если в местах переноса строк добавлять тег < br >. В отличие от абзаца, тег переноса строки < br > не создает дополнительных вертикальных отступов между строками и может применяться практически в любом тексте [1].

Выравнивание текста

Выравнивание текста определяет его внешний вид и ориентацию краев абзаца и может выполняться по левому краю, правому краю, по центру или по ширине. Наиболее распространенный вариант — выравнивание по левому краю, когда слева текст сдвигается до края, а правый остается неровным. Выравнивание по правому краю и по центру в основном используется в заголовках и подзаголовках. Следует иметь в виду, что при использовании выравнивания по ширине, в тексте между словами могут появиться большие интервалы, что не очень красиво.

Для установки выравнивания текста обычно используется тег параграфа c атрибутом align, который определяет способ выравнивания. Также блок текста допустимо выравнивать с помощью тега < div > c аналогичным атрибутом align. Он может принимать следующие значения:

- left выравнивание по левому краю, задается по умолчанию;
- right выравнивание по правому краю;
- center выравнивание по центру;

• justify — выравнивание по ширине (одновременно по правому и левому краю). Это значение работает только для текста, длина которого более, чем одна строка. Атрибут align можно применять как для текста, так и для заголовков [1].

Заголовки

Теги, делающие текст заголовками

<*h*1></*h*1>, <*h*2></*h*2>, <*h*3></*h*3>, <*h*4></*h*4>, <*h*5></*h*5>, <*h*6></*h*6>

Эти теги выделяют текст в виде заголовков. Т.е. каждый заголовок начинается с новой строки, выделен полужирным шрифтом и имеет свой размер (заголовок первого уровня самый большой, шестого - самый маленький).

Эти теги могут использоваться с параметром горизонтального выравнивания *align*.

Начертание текста

Форматирование текста — средства его изменения, такие как выбор начертания шрифта и использование эффектов, позволяющих менять вид текста. В табл. 1.1 перечислены основные теги, которые применяются для изменения оформления текста.

Код НТМL	Описание	Пример
Tекст	Жирное начертание текста	Текст
<i>Текст</i>	Курсивное начертание текста	Текст
^{Teкст}	Верхний индекс	e=mc ²
_{Teкcт}	Нижний индекс	H_2O
<pre>Teкcт</pre>	Текст пишется как есть, включая все пробелы	Текст
Teкcт	Курсивный текст	Текст
<pre>strong>Teкcт</pre>	Жирное начертание текста	Текст

Таблица 1.1. Теги для форматирования текста

Любые теги форматирования текста можно использовать совместно друг с другом. Их порядок в данном случае не важен. Здесь главное соблюдать порядок вложенности тегов. Тег, который открыт первым, должен быть закрыт последним.

Tekcm

Каждый раз все предыдущие теги мы помещаем в новые. Это и называется порядком вложенности. Вы можете использовать разные теги для оформления текста, главное не нарушать порядок вложенности [1].

Ter font и его параметры

Теги ** указывают параметры шрифта текста:

face - название шрифта. Названий шрифтов можно привести несколько, через запятую. В этом случае, если первый указанный шрифт не будет найден (вы же не знаете, какие шрифты установлены на компьютере пользователя), браузер станет использовать следующий по списку.

size - размер шрифта в условных единицах от 1 до 7. По умолчанию размер шрифта равен 3.

color - цвет текста (по умолчанию - черный).

Существуют два способа задания цвета: по имени и указанием шестнадцатеричного кода цвета.

С именными цветами (их 156) все просто, смотрим в соответствующую таблицу (табл. 1.2), выбираем понравившийся цвет и пишем его имя в значение параметра (например, color="blue").

Но гораздо больший выбор предоставляет второй способ. Здесь мы можем выбирать из миллиона цветов, указав его шестнадцатеричный код. Этот код представляет собой 6 цифр и начинается с символа "#". Получить его можно, например, в программе Photoshop [3].

Создание маркированных списков в НТМL

В языке HTML создание маркированного списока (список без нумерации) осуществляется парным тегом

Тег указывает браузеру, что перед ним маркированный список. Для указания пункта списка используется парный тег ...

Маркеры могут принимать один из трех видов: круг (по умолчанию), окружность и квадрат. Для выбора стиля маркера используется атрибут type тега . Допустимые значения приведены в табл. 1.2.

Тип списка	Код НТМL	Пример
Список с маркерами в виде круга	<ul type="disc"> 	ПервыйВторойТретий
Список с маркерами в виде окружности	<ul type="circle"> 	 Первый Второй Третий
Список с квадратными маркерами	<ul type="square"> 	 Первый Второй Третий

Таблица 1.2. Стили маркеров списка

Вид маркеров может незначительно различаться в разных браузерах, а также при смене шрифта и размера текста.

Любой список можно сделать вложенным. Для этого нужно внутри пункта списка (тег) разместить еще один список. Вложенность списков может быть бесконечной [2].

Создание нумерованных списков в НТМL

Создание нумерованного списка, в языке html, осуществляется тегами и , и и тегами пункта списка . Так же как и у маркированного списка в нумерованном можно делать вложенные списки. Для этого нужно внутри пункта списка (тег) разместить еще один список.

Для указания типа нумерованного списка применяется атрибут type тега . Его возможные значения приведены в табл. 1.3.

Тип списка Код HTML П	Пример
-----------------------	--------

Арабские числа	<ol type="1"> 	 Чебурашка Крокодил Гена Шапокляк
Прописные буквы латинского алфавита	<ol type="A"> 	А. ЧебурашкаВ. Крокодил ГенаС. Шапокляк
Строчные буквы латинского алфавита	<ol type="a"> 	а. Чебурашкаb. Крокодил Генас. Шапокляк
Римские числа в верхнем регистре	<ol type="I"> 	 I. Чебурашка II. Крокодил Гена III. Шапокляк
Римские числа в нижнем регистре	 type="i"> 	i. Чебурашкаii. Крокодил Генаiii. Шапокляк

Таблица 1.3. Стили маркеров списка

Чтобы начать список с определенного значения, используется атрибут start тега **При этом не имеет значения**, какой тип списка установлен с помощью type, атрибут start одинаково работает и с римскими и с арабскими числами [3].

ПРАКТИЧЕСКАЯ ЧАСТЬ

Добавление текстовых заголовков в HTML-документе

- 1. Откройте программу Notepad++. Создайте документ example2.html.
- 2. Создайте стандартную структуру документа.
- 3. Добавьте в заголовок html-документа «Практическая работа №2».

4. Добавьте четыре абзаца с помощью тега и задайте его выравнивание (по левому, правому краю, по центру и по ширине) с помощью атрибута align.



Рис. 1.1 – Листинг кода добавления абзацев

5. Сохраните документ без закрытия и проверьте результат в браузере.

Текст выравнивание по левому краю, задается по умолчанию

Текст выравнивание по центру

Текст выравнивание по правому краю

Текст выравнивание по всей ширине страницы(одновременно по правому и левому краю). Это значение работает только для текста, длина которого более, чем одна строка. Следует иметь в виду, что при использовании выравнивания по ширине, в тексте между словами могут появиться большие интервалы, что не очень красиво.

Рис. 1.2 – Результат добавления абзацев в браузере

6. Добавьте в текст документа 6 заголовков разных уровней с использованием тегов <h1></h1>, <h2></h2>, <h3></h3>, <h4></h4>, <h5></h5>, <h6></h6> так, как показано на рисунке:

8	🛱 <body></body>
9	align="left">Текст выравнивание по левому краю, задается по умолчанию
10	align="center">Текст выравнивание по центру
11	align="right">Текст выравнивание по правому краю
12	Текст выравнивание по всей ширине страницы (одновременно по
	правому и левому краю). Это значение работает только для текста, длина которого
	более, чем одна строка. Следует иметь в виду, что при использовании выравнивания по
	ширине, в тексте между словами могут появиться большие интервалы, что не очень
	красиво. р
13	
14	<h1>Заголовок первого уровня</h1>
15	<h2>Заголовок второго уровня</h2>
16	<h3>Заголовок третьего уровня</h3>
17	<h4>Заголовок четвертого уровня</h4>
18	<h5>Заголовок пятого уровня</h5>
19	<h6>Заголовок шестого уровня</h6>

Рис. 1.3 – Листинг кода добавления заголовков

7. Сохраните документ и просмотрите результат в браузере

Заголовок первого уровня
Заголовок второго уровня
Заголовок третьего уровня
Заголовок четвертого уровня
Заголовок пятого уровня
Заголовок шестого уровня

Рис. 1.4 – Результат добавления заголовков в браузере

Выделение текстовых фрагментов в HTML-документе

8. Добавьте выделение фрагментов текста с помощью жирного (теги

u < b>) и курсивного начертания (теги $\langle i \rangle$ и $\langle em \rangle$) так, как показано на рисунке:

8	ų	<pre><pre>cpod3></pre></pre>
9		
0		<h1>Заголовок первого уровня</h1>
1		<h2>Заголовок второго уровня</h2>
2		<h3>Заголовок третьего уровня</h3>
3		<h4>Заголовок четвертого уровня</h4>
4		<h5>Заголовок пятого уровня</h5>
5		<h6>Заголовок шестого уровня</h6>
6		
7		Сильно выделенный текст
8		Жирный текст
9		
0		<i>Курсивный текст</i>
1		Выделенный курсивный текст
2		
3		Жирный курсивный текст
4		
5	H	
_		

Рис. 1.5 – Листинг кода с элементами форматирования текста

9. Сохраните документ. Просмотрите результат в браузере

Заголовок первого уровня
Заголовок второго уровня
Заголовок третьего уровня
Заголовок четвертого уровня
Заголовок пятого уровня
Заголжок шестого уровия
Сильно выделенный текст Жирный текст Курсивный текст Выделенный курсивный текст Жирный курсивный текст

Рис. 1.6 – Результат добавления элементов форматирования текста в браузере

10. Разбейте последний абзац на несколько абзацев, чтобы каждый фрагмент текста был с новой строки, используя тег



- Рис. 1.7 Листинг кода с разбиением на абзацы текстовых элементов
- 11. Сохраните документ. Просмотрите результат в браузере.



Рис. 1.8 – Результат разбиения на абзацы текстовых элементов в браузере **12.** Добавьте фрагменты текста с нижним (<sub>) и верхним (<sup>) индексом:

17	þ	Сильно выделенный текст
18		 Suphin texct
19		
20	þ	<i>Курсивный текст</i>
21		 br> Выделенный курсивный текст
22		
23	Þ	Жирный курсивный текст
24		
25		Формула квадратного уравнения: ax²+bx+c=0
26		Формула воды: H_Q
27	-	
28	L<	/html>

Рис. 1.9 – Листинг кода с форматированием текста

13. Сохраните документ. Просмотрите результат в браузере



Рис. 1.10 – Результат изменения индекса текстового элемента в браузере

Изменение внешнего вида текста в НТМL-документе

14. Измените внешний вид текста с использованием тега и его атрибутов:

25	Формула квадратного уравнения: ax²+bx+c=0
26	Формула воды: H₂0
27	🔁 Следующий текст будет
28	 dot face="Arial">co шрифтом Arial
29	<pre></pre>
30	<pre> со шрифтом Arial, оливкового цвета</pre>
31	<pre> cfont face="Arial" color="#FF1493" size="20pt">co mpu@rom Arial, posoBoro</pre>
	цвета, pasmepa 20 nynktos
32	

Рис. 1.11 – Листинг кода с использованием тега и его атрибутов

15. Сохраните документ. Просмотрите результат в браузере.

Заголовок первого уровня
Заголовок второго уровня
Заголовок третьего уровня
Заголовок четвертого уровня
Заголовок пятого уровня
Загазавае шестого уровия
Сильно выделенный текст Жирный текст
Курсценый төкст Выделенный курсценый төкст
Жирный курсивный текст
Формула квадратного уравнения: ax ² +bx+c=0
Формула воды: H 2 O
Следующий текст будет
со шрифтом Árial со шрифтом Árial, красного цвета со шрифтом Árial, опивкового цвета
со шрифтом Arial, оливкового цвета, размера 20 пунктов

Рис. 1.12 – Результат изменения начертания шрифта в браузере

Создание маркированных списков в НТМL-документе

16. Добавьте заголовок для маркированного списка, видоизменяя его с помощью тега .

17. Создайте двухуровневый маркированный список в документе. Для этого добавьте 2 уровня тегов c обязательными закрывающими тегами. Во втором уровне (вложенном списке) добавьте атрибут type с указанием типа маркера.



Рис. 1.13 – Листинг кода с использованием маркированных списков





Рис. 1.14 – Результат использования маркированных списков в браузере

Создание нумерованных списков в НТМL-документе

19. Добавьте заголовок для нумерованного списка, видоизменяя его с помощью тега .

20. Создайте двухуровневый нумерованный список в документе. Для этого добавьте 2 уровня тегов c обязательными закрывающими тегами. Во втором уровне (вложенном списке) добавьте атрибут type с указанием типа нумерации (атрибут type). Добавьте тег указания начального значения списка во втором уровне (атрибут start).



Рис. 1.15 – Листинг кода с использованием нумерованных списков

21. Сохраните документ. Просмотрите результат в браузере.



Рис. 1.16 – Результат использования нумерованных списков в браузере

РЕФЛЕКСИЯ

- 1. Какие трудности у Вас возникли при выполнении работы?
- 2. Как задается цвет текстового элемента веб-страницы?
- 3. Какой способ выравнивания позволяет задать тег ?

4. Какими методами можно провести выделение текстового фрагмента в вебдокументе?

5. Запишите в таблицу теги, с которыми вы познакомились на этой практической работе и их краткое описание (особенности использования).

		Параметры			
N⁰	Критерии	Выполнено	Выполнено частично	Не выполнено	
1	Листинг кода добавления				
	заголовков				
2	Изменения начертания шрифта в				
	браузере				
3	Использования нумерованных				
	списков в браузере				
4	Результат добавления заголовков в				
	браузере				
5	Цвет текстового элемента веб-				
	страницы				
6	Просмотр результата работы в				
	браузере согласно заданным				
	условиям				

САМООЦЕНКА

ИСПОЛЬЗУЕМАЯ ЛИТЕРАТУРА

- 1. Справочник HTML Режим доступа: <u>http://htmlbook.ru/</u>
- 2. Богомолова О. Б. Web-конструирование на HTML: практикум. М.: Бином. Лаборатория знаний, 2008. – 192 с.
- 3. Справочник HTML Режим доступа: <u>http://www.site-do.ru/</u>

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ПРАКТИЧЕСКОЙ РАБОТЕ: «ПРОЕКТИРОВАНИЕ ЛОКАЛЬНОЙ СЕТИ»

Бурым Елена Владимировна ГПОУ «Донецкий промышленноэкономический колледж», преподаватель компьютерных дисциплин, квалификационной категории «специалист высшей категории»

Тема работы: Проектирование локальной сети средствами программного пакета Cisco Packet Tracer.

Цель работы: Научиться проектировать сеть (логическую) из нескольких компьютеров и коммутаторов; Исследовать работу сети с помощью режима моделирования.

Средства для выполнения работы: ОС Windows, Cisco Packet Tracer 5.3

Порядок выполнения работы:

Сначала на рабочее поле программы переносятся необходимые компоненты схемы в соответствии с заданием (рабочие станции, повторители, коммутаторы, маршругизаторы). Для этого необходимо выбрать тип нужного устройства на панели

Network Component Вох и само устройство - Device-Type Selection Box.

Для составления проекта вам потребуются Оборудование и типы связи в Cisco Packet Tracer. Расположены они в левом нижнем углу рабочего окна программы: на слайде области 7, 8, 9.



Рисунок 1. Рабочее окно программы Cisco Packet Tracer

7. Network Component Box (Панель компонентов сети). Содержит устройства и связи, которые можно поместить на рабочее поле. Содержит панель Device-Type Selection Box i Device-Specific Selection Box.

8. Device-Type Selection Box (Панель выбора типа устройства). Содержит виды устройств и связей, доступных в Packet Tracer 5.3. Панель Device-Specific Selection Box изменится в зависимости от выбранного типа устройства.

9. Device-Specific Selection Box (Панель выбора определенного устройства). Содержит устройства одного типа, из которых надо выбирать то, которое необходимо поместить на рабочую область.



Пользовательские устройства и облако для многопользовательской работы



Рисунок 2. Устройства и связи, которые можно поместить на рабочее поле

Например, чтобы разместить в рабочей области компьютер, необходимо выбрать тип устройства End Devices, устройство PC-PT:

	3 📶	
Logical	Set Tiled Background	
3		
	PC-PT PC0	
1		
] Reset Netwo	rk	
Reset Netwo		
Reset Netwo	rk vices	

Рисунок 3. Размещение объекта в рабочей области программы

На закладке Config (свойств элемента) можно изменить отображаемое название (вместо заданного по умолчанию):

	Edit Hub2			
	Physical	Confi	9	
	GLOBAL	10	Global Settings	
ub-PT tub2	Settings		Display Name Hub2	0

Рисунок 4. Рредактирование названия объекта

Физическая комплектация оборудования в эмуляторе.

Например, выбираем на рабочее поле роутер Cisco 1841. Кликом на роутере открываем его физическую конфигурацию.

Маршрутизатор0			_ [] ×
Физическое простран	ство Конфигурация	я CLI	
молули -	Физ	вический вид устройс	тва
HWIC-4ESW	Maran		
HWIC-AP-AG-B	Увеличить	Оригинальный размер	Уменьшить
WIC-1AM			
WIC-1ENET			
WIC-1T	Terminet and a		Contraction of the second second
WIC-2AM			100
WIC-2T			
WIC-Cover			
-	<[
V	Настроить изображение для физического вида	Настр изображе логическ	ооить ение для ого вида
HWIC-4ESW предост	авляет 4 коммутацио	нных порта.	

Рисунок 5. Физическая комплектация оборудования в эмуляторе.

Слева, список модулей (2), которыми можно укомплектовать данный роутер. Справа - список свободных отверстий/разъемов (3). В них можно вложить эти модули. Понятно, что замену модулей нужно проводить при выключенном питании (1).

Обратите внимание, что подключение выбранным типом связи возможно только при условии наличия соответствующего разъема, то есть зависит от правильности выбора модуля.

После того, как устройства перенесены на рабочее поле, они соединяются кабелем - оптоволокном или медным. В случае если для данного оборудования не хватает какого-то порта (например, для оптоволокна), можно самостоятельно подключить этот модуль к устройству (предварительно отключив его от питания), или выбрать другой вариант оборудования на панели устройств.

Далее рабочим станциям (компьютерам и серверу) назначаются IP-адреса. Определение ip-адреса для компьютера PC1 (маска определена по умолчанию):

dit PC1	C	
Physical Config De	sktop	
IP Configurat	ion	X
Use DHCP		
IP Address	192.168.1.5	
Subnet Mask	255.255.255.0	

Рисунок 6. Определение ір-адреса для компьютера

Далее, после того, как были соединены узлы сети и присвоены адреса рабочим станциям, нужно проверить работу сети.

Самый простой способ исследования работы сети - команда Add Simple PDU. PDU - Protocol Data Unit.

	₽ _
Add Simple PDU	J (P)

Рисунок 7. Проверка работы сети. Передача пакета данных

Simple PDU - ping (посылается пакет ICMP от первого выбранного узла ко второму и обрато).

В данном случае передача пакета от узла РС1 к РС2 прошла успешно:



Рисунок 8. Передача пакета данных от узла РС1 к РС2

В режиме Simulation Mode можно изменять масштаб времени и проследить события, связанные с пересылкой пакетов, а также просмотреть их содержание с помощью команды Inspect:



Рисунок 9. Команда Inspect

Пример собранной сети для 4 компьютеров, 1 концентратора и 1 повторителя



Рисунок 10. Логическая сеть из нескольких компьютеров, концентратора и повторителя

Самостоятельное задание

1. Собрать сеть согласно варианту практической работы (таблица 1), расположение узлов выбрать произвольно.

2. Соединить узлы сети медным кабелем (Copper Straight-through или Copper Cross-over) и присвоить ip-адреса рабочих станций. Все компьютеры относятся к одной подсети.

3. Протестировать работу сети, убедиться, что сеть работает корректно.

4. Составить отчет о проделанной работе.

№ варианта	Количество рабочих станций (PC-PT)	Количество хабов (Hub-PT)	Количество коммутаторов (Switch-PT)
1	10	3	2
2	8	2	1
3	9	1	2
4	10	3	1
5	7	2	2
6	7	1	1
7	8	3	2
8	8	2	1
9	9	1	2
10	9	3	1
11	7	2	2
12	9	1	1
13	10	3	2
14	9	2	1
15	7	1	2
16	7	3	1
17	8	2	2
18	8	1	1
19	9	3	2
20	9	2	1

Таблица 2 – Варианты практической работы

Контрольные вопросы

1. Объясните назначение сетевого адаптера

2. Объясните назначение повторителя.

3. Какие изменения выполняет повторитель в пакете данных, которые передаются по сети?

4. Выполняет ли репитер фильтрацию сигнала?

5. Объясните назначение концентратора

6. Может ли концентратор определить источник или место назначения полученных данных?

7. Может ли концентратор передавать или принимать данные одновременно?

8. На каком уровне модели OSI работает концентратор?

9. В чем заключаются отличия концентратора от повторителя?

10. Объясните назначение моста

11. Может ли мост определить источник или место назначения полученных данных?

12. Где в мосту хранятся адреса передатчиков данных?

13. Может ли мост передавать или принимать данные одновременно?

14. На каком уровне модели OSI работает мост?

15. Можно ли использовать различные топологии и среды передачи в отдельных подсетях, а потом их объединять мостами?

16. Объясните назначение коммутатора

17. Допускает ли коммутатор дуплексную (одновременно в двух направлениях) передачу данных?

18. Может ли коммутатор передавать или принимать данные одновременно?

19. На каком уровне модели OSI работает коммутатор?

20. Можно ли использовать различные топологии и среды передачи в отдельных подсетях а потом их объединять коммутаторами?

21. Какое еще устройство кроме коммутатора обеспечивает дуплексную передачу данных?

22. Могут ли коммутаторы начинать передачу пакета не получив его полностью?

23. Каким образом в мостах создается внутренняя таблица соответствия МАС-адресов и номеров портов?

24. Объясните назначение маршрутизатора

25. На каком уровне модели OSI работает маршрутизатор?

26. Каким образом маршрутизатор выбирает маршрут передачи данных?

27. Имеет ли маршрутизатор свой адрес?

28. Может ли маршрутизатор объединять локальные сети с разными протоколами, например Ethernet и Token Ring?

Используемая литература:

1. Ю.А.Кулаков, С.В. Омелянский «Компьютерные сети. Выбор установка, использование и администрирование, », К: Юниор, 1999 -

2. Н.В.Максимов, И.И Попов, «Компьютерные сети», М: Форум Инфра-М., 2004.-336с

3. Основы работы с Cisco Packet Tracer /http://just-networks.ru/

ЛАБОРАТОРНАЯ РАБОТА «ФАЙЛЫ ПАКЕТНОЙ ОБРАБОТКИ» ПО ДИСЦИПЛИНЕ «ОРГАНИЗАЦИЯ АДМИНИСТРИРОВАНИЯ КОМПЬЮТЕРНЫХ СИСТЕМ»

Долинкин Алексей Юльевич ГПОУ «Шахтерский техникум кино и телевидения им. А.А.Ханжонкова», преподаватель спецдисциплин, специалист второй квалификационной категории

План занятия №_

Дисциплина: МДК.02.02 Организация администрирования компьютерных систем

Дата _____

Группа _____

Тема занятия: Файлы пакетной обработки.

Вид занятия: лабораторная работа.

Тип занятия: занятие по первоначальному формированию умений и навыков.

Цели занятия:

учебные приобрести практические навыки в работе с консольными командами OC Windows. Получить навыки создания и применения файлов пакетной обработки.

развивающие Развитие умений студентов обобщать полученные знания, проводить анализ и сравнения, делать необходимые выводы.

воспитательные Воспитание интереса к своей будущей профессии.

Формируемые компетенции: Администрировать локальные вычислительные сети и принимать меры по устранению возможных сбоев (ПК 2.1).

Администрировать сетевые ресурсы в информационных системах (ПК 2.2).

Обеспечивать сбор данных для анализа использования и функционирования программнотехнических средств компьютерных сетей (ПК 2.3).

Образовательные технологии: Индивидуальная.

Продолжительность занятия: 80 минут.

Место проведения: Лаборатория «Компьютерные сети».

Оснащение:

методическое рабочая программа учебной дисциплины, методическая разработка занятия, раздаточный материал.

материально-техническое персональный компьютер с OC Windows.

Межпредметные связи: Математика, Информатика, Основы программирования и баз данных, МДК.01.01 Организация, принципы построения и функционирования компьютерных сетей.

Список литературы:

 Поляк-Брагинский, А.В. Администрирование сети на примерах. / А.В. Поляк-Брагинский. - 2-е изд., перераб. и доп. - СПб.: БХВ-Петербург, 2008. - 432 с.: ил.
 Попов, А.В. Командные файлы и сценарии Windows Script Host. / А.В. Попов. - СПб. БХВ-Петербург, 2002. - 320 с.

Структура занятия

1.	Организационная часть.	2 мин.
2.	Актуализация опорных знаний.	2 мин.
	Контроль входного уровня знаний студентов.	
3.	Объявление темы, постановка цели и основных задач.	8 мин.
4.	Выполнение студентами задания лабораторной работы.	60 мин.
5.	Ответы на контрольные вопросы.	3 мин.
6.	Подведение итогов работы. Объявление оценок.	3 мин.
7.	Выдача домашнего задания.	2 мин.

Ход занятия

МДК.02.02 Организация администрирования компьютерных систем Лабораторная работа № _

Тема: Файлы пакетной обработки.

Цель: приобрести практические навыки в работе с консольными командами OC Windows. Получить навыки создания и применения файлов пакетной обработки.

Оборудование: персональный компьютер с OC Windows.

Порядок выполнения работы

Теоретическая часть

Пакетный (командный) файл (англ. batch file) - текстовый файл в MS-DOS, OS/2, Windows, содержащий последовательность команд, предназначенных для исполнения командным интерпретатором. После запуска пакетного файла программа-интерпретатор (COMMAND.COM для *.BAT и CMD.EXE для *.CMD) читает его строка за строкой и последовательно исполняет команды.

Пакетные файлы полезны для автоматического последовательного запуска команд и приложений. Примерами таких операций могут служить: обработка текстовых файлов, копирование, перемещение, переименование, удаление файлов, работа с каталогами, архивация, создание резерных копий и т.д.

Важным свойством командных является возможность использовать внутри них формальные параметры. При этом обращение к командному файлу приобретает вид:

file.bat параметр1 [параметр2]

Параметры, значения которых будут заданы при обращении к командному файлу, внутри файла будут иметь значения **%1**, **%2**,...,**%**N (где N < 10).

Пакетные файлы поддерживают операторы <u>if</u>, <u>goto</u> и <u>for</u> (в системах семейства <u>Windows NT</u> его возможности существенно расширены), что позволяет обрабатывать результаты выполнения предыдущих команд или приложений и в зависимости от этого выполнять дальше тот или иной блок команд (как правило, в случае удачного завершения приложение возвращает 0 в переменной errorlevel; в случае неудачного — 1 или большее значение).

Операторы, которые используются только в командных файлах:

ЕСНО OFF – блокирует выдачу на экран текстов команд.

ЕСНО ON – отменяет действие ЕСНО OFF.

ЕСНО <TEXT> – выводит текст или значение переменной (параметра) на экран.

REM – комментарий в тексте командного файла. Текст после **REM** не обрабатывается.

PAUSE – приостанавливает выполнение командного файла до нажатия любой клавиши.

GOTO – передает управление на метку (метка должна занимать отдельную строку и иметь первый символ двоеточие «:»). Может использоваться с оператором **IF**.

IF – позволяет проверить условие и выполнить команду в зависимости от результатов его проверки. В качестве условия могут быть:

1. Проверка кода завершения программы, выполняющейся перед оператором IF:

IF ERRORLEVEL <N> <команда>

2. Проверка наличия файла:

IF EXIST <имя или шаблон файла> <команда>

Пример применения (если файл отсутствует, то перейти на метку):

IF NOT EXIST D:\TASM ARX GOTO CREATE

3. Сравнение двух строк, которые могут быть заданными через параметры или получены любым другим способом:

IF %<N> == <текстовая строка > <команда>

Любое из этих условий может задаваться с использованием логического оператора **NOT.**

FOR – обеспечивает циклическое выполнение команд. При этом можно задать формальный параметр и список фактических параметров (обычно имена файлов), которые последовательно подставляются в текст исполняемой команды вместо формального параметра.

FOR <формальный параметр> IN (<список фактических параметров>) DO <команда>

Например, команда выполняет копирование на диск **D**: трех файлов с именем LAB08 и расширениями ASM, OBJ, COM:

FOR %%A IN (ASM OBJ COM) DO COPY LAB08.%%A D:

Практическая часть

0. Изучить теоретическую часть.

1. Открыть консоль Windows.

2. Перейти в каталог **D:\TASM.**

3. Создать командный файл в каталоге D:\TASM выполняющий следующие действия (для создания командного файла можно использовать любой текстовый редактор):

- создать каталог D:\TASM_ARX, предварительно проверить существование каталога, в зависимости от наличия или отсутствия каталога вывести сообщение «Каталог D:\TASM_ARX существует. Создание каталога не выполняется» или «Каталог D:\TASM_ARX отсутствует. Выполняется создание каталога»

- скопировать в каталог D:\TASM_ARX все файлы из каталога D:\TASM с расширением ASM.

- создать каталог D:\TASM_ARX\COM, предварительно проверить существование каталога, в зависимости от наличия или отсутствия каталога вывести сообщение «Каталог D:\TASM_ARX\COM существует. Создание каталога не выполняется» или «Каталог D:\TASM_ARX\COM отсутствует. Выполняется создание каталога»

- скопировать в каталог D:\TASM_ARX все файлы из каталога D:\TASM\COM с расширением COM.

- проверить наличие файлов TASM.EXE, TLINK EXE, TASM.BAT в каталоге D:\TASM_ARX, в случае отсутствия вывести об этом сообщение на экран и скопировать TASM.EXE и TLINK EXE из D:\TASM\BIN, TASM.BAT из D:\TASM.

- отключить режим отображения на экран выполняемой команды.

4. Текст командного файла записать в отчет.

5. В отчете ответить на следующие вопросы:

- 5.1. Как и зачем используется команда IF?
- 5.2. Как и зачем используется команда GOTO?
- 5.3. Как и зачем используется команда FOR?
- 5.4. Зачем используется команда PAUSE?
- 6. Предоставить отчет и созданный командный файл.
- 7. Защита отчета о проделанной работе.

Используемая литература:

1. Поляк-Брагинский, А.В. Администрирование сети на примерах. / А.В. Поляк-Брагинский. - 2-е изд., перераб. и доп. - СПб.: БХВ-Петербург, 2008. - 432 с.: ил.

2. Попов, А.В. Командные файлы и сценарии Windows Script Host. / А.В. Попов. - СПб. БХВ-Петербург, 2002. - 320 с.

ЛАБОРАТОРНАЯ РАБОТА № 7 «ИЗУЧЕНИЕ ФИЛЬТРА ОТОБРАЖЕНИЯ ПАКЕТОВ» ПО ДИСЦИПЛИНЕ «БЕЗОПАСНОСТЬ ФУНКЦИОНИРОВАНИЯ КОМПЬЮТЕРНЫХ СИСТЕМ»

Бережная Елена Владимировна ГПОУ «Шахтерский техникум кино и телевидения имени А.А.Ханжонкова», преподаватель дисциплины «Безопасность функционирования компьютерных систем», специалист высшей квалификационной категории

Тема занятия Изучение фильтра отображения пакетов

Вид занятия практическое

Тип занятия занятие совершенствования знаний, умений и навыков

Формы обучения индивидуальная

Цель занятия: приобретение навыков захвата сетевого трафика в сегменте локальной сети и анализа собранной информации с помощью программного анализатора протоколов Ethereal. Приобретение навыков работы с фильтрами.

Формируемые компетенции

ПК 3.1. Устанавливать, настраивать, эксплуатировать и обслуживать технические и программно-аппаратные средства компьютерных сетей.

ПК 3.3. Эксплуатация сетевых конфигураций.

Оборудование. Фрагмент локальной вычислительной сети на концентраторе или коммутаторе, включающий в себя рабочие станции с операционной системой Windows 2007 по количеству слушателей. Для установки необходимого программного обеспечения на рабочих станциях должны быть доступны инсталляционные пакеты библиотеки WinPCap (версия не ниже 2.3) и анализатора Ethereal (версия не ниже 0.10.11).

Межпредметные связи ____ Основы теории информации, Технология физического уромня передачи данных, Операционные системы

Список литературы:

1.Шаньгин В.Ф. Информационная безопасность компьютерных систем и сетей: учеб. пособие/ В.Ф.Шаньгин. – М.: ИД «ФОРУМ»: ИНФРА-М, 2011. – 416с.

2.Андрончик А. Н., Богданов В. В., Домуховский Н. А., Коллеров А. С., Синадский Н. И., Хорьков Д. А., Щербаков М. Ю. Защита информации в компьютерных сетях.

Практический курс: учебное пособие / А. Н. Андрончик, В. В. Богданов, Н. А.

Домуховский, А. С. Коллеров, Н. И. Синадский, Д. А. Хорьков, М. Ю. Щербаков; под ред. Н. И. Синадского. Екатеринбург: УГТУ-УПИ, 2008. 248 с.

3. Заборовский В.С., Мулюха В.А., Новопашенный А.Г. Сети ЭВМ и телекоммуникации. Анализ трафика в сетях коммутации пакетов: учеб.пособие/ В.С. Заборовский, В.А. Мулюха, А.Г. Новопашенный. – СПб.: СПбГПУ, 2010.

ИНСТРУКТИВНО-МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ВЫПОЛНЕНИЮ ЛАБОРАТОРНОЙ РАБОТЫ №7 «ИЗУЧЕНИЕ ФИЛЬТРА ОТОБРАЖЕНИЯ ПАКЕТОВ»

Теоретические сведения

Фильтрация. Часто, когда трафик в исследуемой сети достаточно интенсивен, за короткое время может быть захвачено большое количество пакетов. Анализ трафика при этом может быть весьма затруднителен. Чтобы упростить задачу, рекомендуется воспользоваться двумя видами фильтрации.

Фильтрация во время захвата производится с использованием правил Capture Filters. В этом случае будут захватываться (запоминаться) только пакеты, удовлетворяющие данным правилам. Такой отбор целесообразно производить для уменьшения количества анализируемых пакетов в том случае, когда исследователь заранее может определить интересующие его параметры фильтров. Следует учесть, что информация о пакетах, не удовлетворяющих фильтру, теряется.

Фильтр составляется (или выбирается) до начала захвата.

Фильтр вводится в специальную строку (Capture Filters) в диалоговом окне Capture Options.

При создании фильтров можно использовать логику and, or или not:

[not] **primitive** [and|or [not] **primitive** ...]

Операция сравнения «Равно» записывается с помощью двойного знака равенства «==» (допустимо использование «eq»). Другие операции сравнения записываются с помощью следующих операторов:

Основные операторы, используемые для фильтрации, указаны в таблице 1.

Обозначение	Оператор	Значение	
eq	==	Равенство	ip.addr == 10.0.0.5
ne	!=	Неравенство	ip.addr != 10.0.0.5
gt	>	Больше чем	<pre>frame.pkt_len > 10</pre>
lt	<	Меньше чем	<pre>frame.pkt_len < 10</pre>
ge	>=	Больше или равно чем	frame.pkt_len ge 0x100
le	<=	Меньше или равно чем	<pre>frame.pkt_len <= 0x20</pre>
and	& &	Логическое И	ip.addr==10.0.0.5 and tcp.flags.fin
or		Логическое ИЛИ	ip.addr==10.0.0.5 or ip.addr==192.1.1.1

Таблица1. Операторы и их значения фильтрации

xor	~~	Исключающее ИЛИ	Tr.dst[0:3] == 0.6.29 xor tr.src[0:3] == 0.6.29
not	!	Отрицание	not llc

Пример 1:

tcp port 23 and host 10.0.0.5

В данном примере захватывается telnet трафик, передаваемый и принимаемый хостом 10.0.0.5.

Пример 2:

tcp port 23 and not host 10.0.0.5

В данном примере захватывается весь telnet трафик, передаваемый и принимаемый всеми хостами, кроме 10.0.0.5

Примитивы, которые могут быть использованы для захвата, указаны в таблице2.

Таблица2. Н	Выражения	фильт	рации	иих	значения
-------------	-----------	-------	-------	-----	----------

Выражение Значение выражения		Пример записи		
frame.number	Номер кадра	<pre>frame.number == true</pre>		
frame.time	Время захвата кадра	<pre>frame.time == "Feb 1, 2016</pre>		
		09:00:00"		
frame.pkt_len	Длина кадра	<pre>frame.pkt_len == 48</pre>		
eth.dst	Заголовок Ethernet: MAC-	eth.dst ==		
	адресс назначения	01:00:5e:00:00:02		
eth.src	Заголовок Ethernet: MAC-	eth.src ==00:a0:cc:30:c8:db		
	адресс источника			
eth.type	Заголовок Ethernet: тип	eth.type == 0x0800		
	вложенного протокола			
arp.src.hw_mac	Заголовок протокола ARP:	arp.src.hw_mac ==		
	МАС-адресс источника	00:10:4b:30:c4:4a		
arp.dst.hw_mac	Заголовок протокола ARP:	arp.dst.hw_mac		
	МАС-адресс назначения	==00:00:00:00:00:00		
arp.src.proto_i	Заголовок протокола ARP:	arp.src. proto_ipv4 ==		
pv4	IP-адресс источника	10.1.0.2		
arp.	Заголовок протокола ARP:	arp. dst.proto_ipv4 ==		
dst.proto_ipv4	IP-адресс назначения	10.1.0.2		

Задание на работу и содержание отчета

1. Запустите Ethereal и разверните главное окно приложения на весь экран (для удобства работы).

Перед выполнением захвата сетевого трафика необходимо настроить параметры захвата или проконтролировать установленные значения некоторых из них так, чтобы собранная информация адекватно соответствовала решаемой задаче анализа трафика.

2. Выполните команду меню Capture⇒Options.

В открывшемся диалоговом окне устанавливаются следующие параметры захвата кадров (рис. 1):

- Interface — сетевой адаптер;

Очень важно выбрать соответствующий сетевой адаптер, иначе запись кадров будет производиться из другого сегмента сети! В компьютере, имеющем всего один сетевой адаптер, среди возможных сетевых интерфейсов часто присутствует контроллер удаленного доступа!

- Buffer size — размер буфера захвата (по умолчанию 1 Мб);

При малом размере буфера существует опасность того, что при его заполнении запись новых кадров будет производиться поверх записанных ранее!

- Capture packets in promiscuous mode — использование режима беспорядочного захвата.

Capture								
Interface:	NVIDIA r	Force	МСР	Net	tworking Adapt	er Dri	ver: \Device\NPF_{9424D6B1-A2A2-	
IP address: 1 Link-layer h	92.168.1.1 eader typ oackets in	L5 e: Eth promis	nerne scuou	t Js n	Buffer size:	1	megabyte(s)	
🔲 Limit eac	h packet i	to 68			🗘 bytes			
Capture Filt	er:							
Capture File(s)						Display Options	
File: Browse						Update list of packets in real time		
🔲 Use <u>m</u> ulti	ple files							
🗌 Next file e	every	1		*	megabyte(s)	*	Automatic scrolling in live capture	
Next file every 1			*	minute(s)	-	🔲 Hide capture info dialog		
Ring buffer with 2			🔹 files		19 B.			
Stop capt	ure after	1		tile(s)			-Name Resolution	
Stop Capture							Enable MAC name resolution	
📃 after	1		4. 14	packet(s)			Enable network name resolution	
🔲 after	1		A. V	m	megabyte(s)			
🔲 after	1	1		minute(s) 👻		*	Enable transport name resolution	

Рис. 1. Окно настройки параметров захвата

- Limit each packe tto — запись только нескольких первых байт (определяется установленным значением параметра) каждого кадра;

- Capture Filter — фильтр захвата;

Фильтр захвата экономит объем буфера, отбрасывая «лишний мусор», однако увеличивает нагрузку на процессор, вследствие чего некоторые кадры могут быть потеряны. Поэтому в некоторых случаях вместо фильтра записи предпочтительнее использовать фильтр отображения кадров в буфере, а запись производить без фильтрации!

- Capture File(s) — файл захвата;

Опция полезна при осуществлении захвата трафика в течение длительного периода времени.

- Stop Capture — условия автоматического завершения захвата;

– **Display Options** — отображение пакетов в реальном времени и автоматический скроллинг окна информации;

Опции увеличивают нагрузку на процессор, вследствие чего некоторые кадры могут быть потеряны.

– **Name Resolution** — разрешение имен на физическом, сетевом и транспортном уровнях.

3. Уберите маркер напротив опции «Capture packets in promiscuous mode» для захвата только «своих» кадров (кадры с широковещательным адресом также будут захватываться). В таком режиме работы число захваченных пакетов будет существенно меньше, что облегчит выполнение заданий.

С помощью фильтра отображения можно быстро убрать лишние пакеты. Выражение фильтрации может представлять собой просто название протокола, который присутствует в пакете на том или ином уровне вложенности. Например: arp — для отображения пакетов протокола ARP, tcp — для отображения пакетов, в которых присутствует заголовок протокола TCP.

4. Для отображения только ICMP-сообщений в строке ввода «Filter» (рис. 2) наберите «icmp» и нажмите кнопку «Apply».

🕝 (Untitled) - Ethereal	
<u> Eile Edit View Go Capture Analyze Statistics H</u> elp	
Eilter: icmp 🔻 Expression Clear	г Дррју
No Time Source Destination Protocol Info	A
29 1.172721 93.120.223.45 192.168.1.15 ICMP Destin	nation unreachable (Host unreachable)
32 1.262113 128.69.25.88 192.168.1.15 ICMP Destin 57 2.213086 78.182.205.47 192.168.1.15 TCMP Destin	nation unreachable (Host unreachable)
61 2.979014 192.168.1.15 192.168.1.25 ICMP Echo ((ping) request
64 2.980675 192.168.1.25 192.168.1.15 ICMP EChO ((ping) reply
94 3.980253 192.168.1.15 192.168.1.25 ICMP Echo ((ping) request
95 3.981950 192.168.1.25 192.168.1.15 ICMP Echo ((ping) reply
114 4.249204 188.242.5.174 192.168.1.15 ICMP DESCI 119 4.980227 192.168.1.15 192.168.1.25 ICMP Echo ((ping) request
120 4.981544 192.168.1.25 192.168.1.15 ICMP Echo ((ping) reply
145 5.783184 88.225.236.249 192.168.1.15 ICMP Destin 147 5.980286 192.168.1.15 192.168.1.25 ICMP Echo ((ping) request
148 5 981370 192 168 1 25 192 168 1 15 TCMP Echo ((nind) renly
۳	•
⊕ Frame 29 (94 bytes on wire, 94 bytes captured)	
	starM_a2:32:aa (00:30:67:a2:32:aa)
Internet Control Message Protocol	00.1.15 (192.100.1.15)
יד	-
0010 00 50 4c fc 00 00 37 01 38 54 5d 78 df 2d c0 a8 .PL7. 8T]x
0020 01 0f 03 01 f1 22 00 00 00 00 45 00 00 34 2a bb"	E4*.
0040 4d f1 9a 5f b3 2b 00 00 00 00 80 02 20 00 ca 16 M+	
File: "C:\Users\7272~1\AppData\Local\Temp\etherXXXCUR6QY" 20 KB 00:00:09 P: 236 D: 16 M: 0 I	Drops: 0

Рис.2 Отображения только ІСМР-сообщений

5. **Выясните, что будет отображено в буфере захвата** в случае использования фильтра, описанного с помощью выражений, приведенных в качестве вышеописанных примеров.

Значение любого выражения фильтрации возвращает переменную булевского типа. Таким образом, выражение udp означает присутствие в кадре заголовка протокола UDP, по аналогии с этим выражение tcp.flags.syn означает присутствие в заголовке протокола TCP бита синхронизации сессии в установленном состоянии (значение1). К любому из выражений можно применить операцию логического отрицания, заключив его в скобки и поставив перед ним знак отрицания «NOT» или«!». Например, выражение !(ip.addr == 10.0.0.1) означает, что из буфера отображения необходимо убрать все пакеты, в которых встречаетсяIP-адрес10.0.0.1.

🥝 (Untitled) - Ethereal						
<u>File Edit View G</u> o	<u>Capture</u> <u>Analyze</u> <u>S</u> tatist	ics <u>H</u> elp				
	🂓 🕞 🔚 🗙	r 🖓 📇 🛛 🖓	🔿 轮 🛧 💆 📗			
Eilter: tcp.srcport> 1023 Expression Clear Apply						
No. Time	Source	Destination	^p rotocol. Info	^		
181 7.015793 182 7.016726	192.168.1.15 192.168.1.15	128.71.250.147 78.185.183.22	TCP 62896 > 45773 TCP 62897 > 33148	3 [SYN] S 3 [SYN] S		
186 7.201389 187 7.270079 192 7 700092	40.42.200.255	192.108.1.15 185.42.98.225 46.42.206.253	TCP 14506 > 6289 TCP TCP ZeroWind TCP 62895 > 14506	dowProbe		
193 7.832281 194 7.884566	78.185.183.22 46.42.206.253	192.168.1.15 192.168.1.15	ICMP Destination (TCP 14506 > 6289	Inneachab 5 [RST, A		
197 8.005017 198 8.005065	192.168.1.15 192.168.1.15	178.187.193.45 178.91.93.129	TCP 62852 > 18669 TCP 62886 > 36499	9 [SYN] 5 5 [SYN] 5		
199 8.005088 200 8.005110 201 8.005123	192.168.1.15 192.168.1.15	176.215.248.53 85.106.118.252	TCP 62850 > 33600 TCP 62851 > 14992	0 [SYN] 5 2 [SYN] 5		
201 8.003135 202 8.010020 203 8.010065	192.168.1.15 192.168.1.15 192.168.1.15	88.227.62.210 78.169.33.62	TCP 62853 > 23833 TCP 62853 > 23833 TCP 62849 > 57767	L [SYN] S		
204 8.010091 205 8.010114	192.168.1.15 192.168.1.15	31.129.154.179 95.220.77.200	TCP 62881 > 18811 TCP 62882 > 30717	[<u>SYN]</u> 8 7 [SYN] 5 -		
•	<u>></u> m			×		
 Frame 204 (66 bytes on wire, 66 bytes captured) Ethernet II, Src: BiostarM_a2:32:aa (00:30:67:a2:32:aa), Dst: f0:7d:68:15:1b:c0 (Internet Protocol, Src: 192.168.1.15 (192.168.1.15), Dst: 31.129.154.179 (31.129. 						
		C POIC. 02001 (0200.	17. DSC POLC. 10011	+		
0000 f0 7d 68 15 0010 00 34 2b 49 0020 9a b3 f5 a1 0030 20 00 84 e2 0040 04 02	1b c0 00 30 67 a2 40 00 80 06 53 81 49 7b 03 9d 0b 88 00 00 02 04 05 b4	2 32 aa 08 00 45 00 f c0 a8 01 0f 1f 81 3 00 00 00 00 80 02 4 01 03 03 08 01 01	.}h0 g.2E. .4+I@ 5 I{			
File: "C:\Users\7272~1\AppData\Local\Temp\etherXXXXCUR6 P: 236 D: 179 M: 0 Drops: 0						

Рис.3 Синим выделен пакет, который комментируется внизу и порт его имеет значение больше, чем 1023. И так для всех пакетов.

🥝 (Untitled) - Ethereal						
<u>F</u> ile <u>E</u> dit <u>V</u> iew <u>G</u> o	<u>Capture</u> <u>Analyze</u> <u>S</u> tatisti	cs <u>H</u> elp				
🗐 🐖 🞯 🛞	💓 🗁 🗔 🗙	¢ 🕹 🗟 🗢	🔿 轮 🐴	⊉ 🔲		
Eilter: frame.pkt_len lt (66		Expression <u>C</u> I	lear <u>A</u> pply		
No. Time	Source	Destination	Protocol. Info	*		
195 7.966798 197 8.005017 199 8.005088 200 8.005110 202 8.010020 203 8.010065 213 8.390093 214 8.574990 215 8.608949 218 9.005073 219 9.005159 220 9.010068 221 9.010154 222 9.010203 223 9.010251 236 9.149176	BiostarM_a2:3c:12 192.168.1.15 192.168.1.1	Broadcast 178.187.193.45 176.215.248.53 85.106.118.252 88.227.62.210 78.169.33.62 46.42.206.253 192.168.1.15 Broadcast 95.10.239.148 5.166.75.79 188.235.39.162 93.120.223.45 217.64.129.151 95.152.46.41 Broadcast	ARP Who TCP 6285 TCP 6285	has 192.168 2 > 18669 0 > 33600 1 > 14992 3 > 23831 9 > 57767 5 > 14506 6 > 62895 has 192.168 7 > 58862 8 > 61460 4 > 46922 5 > 19953 6 > 42025 ≡ 9 > 60887 has 192.16		
٠ [an.					
Frame 236 (60 bytes on wire, 60 bytes captured) Ethernet II, Src: BiostarM_a2:2b:8b (00:30:67:a2:2b:8b), Dst: Broadcast (ff:f Address Resolution Protocol (request)						
•				۰.		
0000 ff ff ff ff 0010 08 00 06 04 0020 00 00 00 00 0030 00 00 00 00	ff ff 00 30 67 a2 00 01 00 30 67 a2 00 00 c0 a8 01 01 00 00 00 00 00 00 00	2b 8b 08 06 00 01 2b 8b c0 a8 01 19 00 00 00 00 00 00 00 00	0 0	g.+ g.+		
File: "C:\Users\7272~1\AppData\Local\Temp\etherXXX P: 236 D: 97 M: 0 Drops: 0						

Рис.4 Размер пакетов не превышает 66 байт.

Составить и записать фильтры показа (Display Filter),

обеспечивающие отображение пакетов по заданным признакам:

- 6. Всего трафика с заданного МАС адреса
- 7. Всего трафика между заданными МАС адресами
- 8. Всего трафика с заданного ІР адреса
- 9. Всего трафика на заданный IP адрес
- 10. Всего трафика на заданный IP адрес, кроме трафика ARP
- 11. Всего широковещательного
- 12. Широковещательного без ARP запросов
- 13. Трафика с заданного ІР адреса на заданный ІР адрес
- 14. Всего трафика на заданный IP адрес, port

15. Трафика с заданного IP адреса на заданный IP адрес, исключая трафик на заданный port

- 16. Только ІСМР трафика
- 17. Только DNS ответов

Результаты отразить в отчете.

ЛАБОРАТОРНАЯ РАБОТА «ИССЛЕДОВАНИЕ RS-ТРИГГЕРА» ПО ДИСЦИПЛИНЕ «ЦИФРОВАЯ СХЕМОТЕХНИКА»

Крокошенко Екатерина Юрьевна Донецкое ВПГУ преподаватель профессионально-теоретической подготовки, категория специалист

Аннотация

При проведении лабораторных работ по МДК 01.01 «Цифровая схемотехника» мною используется метод учебного проекта. Метод учебного проекта - целенаправленная, в целом самостоятельная деятельность учащихся, осуществляемая под руководством преподавателя, направленная на решение творческой, исследовательской, личностно или социально значимой проблемы и на получение конкретного результата (в данном случае выполнению практической работы). Этот метод позволяет:

- ✓ Научить учащихся самостоятельному критическому мышлению.
- ✓ Размышлять, опираясь на знания фактов, закономерностей науки, делать собственные выводы.
- ✓ Принимать самостоятельные аргументированные решения.
- ✓ Научить работать в команде, выполняя разные задачи.

При проведении лабораторных работ учащиеся работают в группах по 2-3 человека, создаваемых на различной основе и на различный срок. Это форма обучения позволят эффективно проводить при освоении новых программных средств, при работе над проектами, при недостаточном количестве компьютеров и т.д. Эта форма отражает реальное разделение труда в коллективе учащихся (программистов), работающих над одной задачей. При обучении в составе группы внутри нее возникает интенсивный обмен информацией, а усвоение знаний и умений происходит результативнее при общении учащихся с более подготовленными товарищами.

Используется также метод соревнования, направленный на формирования умений и навыков путем здорового соперничества в определенных видах коллективной деятельности.

По моему мнению, только при таком подходе общение с компьютером увеличивает потребность в приобретении знаний, способствует развитию творческих способностей каждой личности, формированию умений и навыков свободно ориентироваться в мире стремительно развивающихся информационных технологий.

Итак, учащиеся делятся на рабочие группы, каждая из которых получает идентичные задание для выполнения практической работы. Каждая из групп будет работать в своей программе для моделирования электронных схем без их сборки. Используемые программы: Electronic Workbench и Micro-Cap.

Electronic Workbench (EWB) - мощная программа для моделирования процессов и расчета электронных устройств на аналоговых и цифровых элементах. Большой выбор виртуальных генераторов, тестеров, осциллографов. Совместима с программами РСВдизайна и САD. Программа EWB позволяет проверить и проанализировать работу электронных схем без их сборки.

Місго-Сар - профессиональная программа аналогового, цифрового и смешанного моделирования и анализа цепей электронных устройств средней степени сложности[1]. Программа имеет интуитивно понятный интерфейс, нетребовательность к вычислительным ресурсам персонального компьютера и большой спектр возможностей. Алгоритм работы включает в себя создание электрической цепи в графическом редакторе, задание параметров анализа и изучение полученных данных. Программа самостоятельно составляет

уравнения цепи и проводит моментальный расчёт. Любое изменение схемы или параметров элементов приводит к автоматическому обновлению результатов. Графический редактор опирается на библиотеки электронных компонентов, которые можно пополнять на основе экспериментальных или справочных данных с помощью встроенного модуля Shape Editor. Все номиналы и параметры элементов могут быть как неизменными, так и зависящими от температуры, времени, частоты, состояния схемы, параметров других компонентов. Анимированные детали (светодиоды, реле, семисегментные индикаторы и некоторые другие элементы) изменяют состояние в соответствии с поступающими на них сигналами. Моделирование включает в себя целый набор различных анализов: переходных процессов, передаточных характеристик по постоянному току, малосигнальных частотных характеристик, чувствительностей по постоянному току, нелинейных искажений, метода Монте-Карло и многих других. Полная поддержка SPICE-моделей позволяет применять проекты из других программ (DesignLab, OrCAD, P-CAD).

После выполнения практической работы производиться анализ полученных результатов, их сравнение между группами.

Текст работы Ход работы:

Цель: изучение принципов построения и функционирования, а также экспериментальное исследование асинхронного и синхронного RS-триггеров в различных программах моделирования.

Оборудование: ПК с учебной программой Місгосар 7,0, ПК с учебной программой ElectronicsWorkbenchv. 5.12.

Время выполнения – 2 часа (урока), на втором занятии учащиеся 1 группы выполняют задание 2 группы и наоборот соответственно.

План работы:

1.Изучение теоретического материала - 5 мин.

2.Исследование работы асинхронного RS-триггера (в различных программах моделирования) - 15 мин.

3.Исследование работы синхронного RS-триггера (в различных программах моделирования) – 15 мин.

4. Сравнение полученных результатов -7 мин.

5.Контрольные вопросы – 3 мин.

1. Теоретическая часть [3, стр. 152-159] (просмотр презентации «RS-триггеры»)

2.Практическая часть

Порядок выполнения работы:

2.1 Исследование работы асинхронного RS-триггера в Місго Сар 7(задание для 1 группы)

2.1. Запустить программу моделирования Місго Сар 7[2].

Собрать схему асинхронного RS-триггер на элементах Пирса (ИЛИ-НЕ)


Рис. 2.1.1 Схема асинхронного RS-триггер на элементах Пирса.

Запустить моделирование Analisis. Transient. Ran, получить временную диаграмму[2, стр. 57]



Рис. 2.1.2. Временная диаграмма (ВД)

Обратите внимание: Для элементов ИЛИ-НЕ активный входной сигнал логическая 1.

R - установка выхода Q в состояние 0

S - установка выхода Q в состояние 1

Задание: по ВД составить таблицу переходов:

Таблица 1. Таблица переходов асинхронного RS-триггер на элементах Пирса.

Вход		Выход		
R	S	Q	nQ	
0	0	Заполнить	Заполнить	
0	1	Заполнить	Заполнить	
1	0	Заполнить	Заполнить	
1	1	Заполнить	Заполнить	

Собрать схему асинхронного RS-триггер на элементах Шеффера (И-НЕ)



Рис. 2.1.3 Схема асинхронного RS-триггер на элементах Шеффера Запустить моделирование Analisis.... Transient.... Ran, получить временную диаграмму (ВД) и заполнить таблицу переходов:

Вход		Выход		
nR	nS	Q	nQ	
0	0	Заполнить	Заполнить	
0	1	Заполнить	Заполнить	
1	0	Заполнить	Заполнить	
1	1	Заполнить	Заполнить	

Таблица 2. Таблица переходов асинхронного RS-триггер на элементах Шеффера

Обратите внимание: Для элементов И-НЕ активный входной сигнал логического 0.

2.2 Исследование работы асинхронного RS-триггера в ElectronicsWorkbenchv. 5.12 (задание для 2 группы)

Программная реализация асинхронного - триггера в пакете ElectronicsWorkbenchv. 5.12(EWB):

После запуска программы перед пользователем появляется окно (рис. 2.2.4)



Рис. 2.2.4 – Окно программы

1 - группа пассивных элементов;

- 2 группа активных элементов;
- 3 группа цифровых устройств;
- 4 рабочее окно;
- 5 группа логических элементов;
- 6 группа индикаторов;
- 7 группа источников сигналов.

Выполнение работы начинается с перемещения необходимых элементов в рабочее окно

(4). Для перемещения источников сигнала и «земли» необходимо нажать кнопку (2).



1 – источник питания;

2 – «земля».

Рис. 2.2.4. – Активные окна источников питания и «земли».



1 –световой индикатор.

Рис. 2.2.5. – Активные окна индикаторов.



1 – асинхронный -триггер.

Рис. 2.2.6. – Активные окна цифровых устройств.

Нажав (и не опуская) левую кнопку мыши, переносим элемент в рабочее окно, после чего опускаем кнопку мыши. Действуя аналогичным образом, перетаскиваем в рабочее окно следующие элементы: четыре световых индикатора из группы индикаторов (6), источник питания и «землю» из группы активных элементов (2); два ключа из группы пассивных элементов (1); триггер из группы цифровых устройств (3).

Закончив подготовительную работу, необходимо приступить к сборке схемы. Соединения с помощью проводников производятся следующим образом: курсор мыши помещаем на контакт одного из соединяемых элементов, затем нажав левую кнопку мыши, подводим курсор к контакту другого элемента и отпускаем левую кнопку.

Включение и отключение питания схемы осуществляется путем нажатия переключателя, расположенного в верхнем правом углу экрана.

Задание: собрать схему для исследования работы асинхронного RS – триггера, приведенную на рис. 2.2.7.



Рис. 2.2.7. – Принципиальная схема для исследования работы асинхронного RS – триггера.

Запустить моделирование, заполнить таблицу переходов (табл.3), по таблице переходов получить временную диаграмму (ВД).

При исследовании получили следующие значения, которые показаны в таблице 2.2. Таблица 3 - Состояния асинхронного RS-триггера

Входы		Выходы		
S	R	Q	nQ	
0	0	Заполнить	Заполнить	
0	1	Заполнить	Заполнить	
1	0	Заполнить	Заполнить	
1	1	Х		

Из полученных результатов должно быть видно полное соответствие с результатами из теории, что должно подтвердить правильность исследуемой схемы.

3.1. Исследование работы синхронного RS-триггера в Місго Сар 7(задание для 1 группы)

Собрать схему синхронного RS-триггер на элементах Шеффера (И-НЕ), приведенную на рис.3.1.1.



Рис.3.1.1.

Синхронный (тактируемый) RS-триггер на элементах И-НЕ.

Запустить моделирование Analisis.... Transient.... Ran.

Составить по ВД (рис. 3.1.2) и заполнить таблицу переходов (табл.4)



Рис. 3.1.2. Временная диаграмма синхронного RS-триггера на элементах И-НЕ. Таблица 4. – Таблица переходов синхронного RS-триггера на элементах И-НЕ

Вход		Выход		
R	S	С	Q	nQ
0	0	0	Заполнить	Заполнить
0	0	1	Заполнить	Заполнить
0	1	0	Заполнить	Заполнить
0	1	1	Заполнить	Заполнить

1	0	0	Заполнить Заполнить	
1	0	1	Заполнить Заполнить	
1	1	0	Заполнить Заполнит	
1	1	1	Заполнить	Заполнить

3.2. Исследование работы синхронного RS-триггера в 7(задание для 2 группы)

Программная реализация синхронного - триггера в пакете Electronics Workbenchv. 5.12. Подготовка к работе и сборка схемы для исследования синхронного - триггера аналогична сборке схемы для исследования асинхронного - триггера, только асинхронный триггер нужно заменить на синхронный. Синхронный - триггер соберем из логических элементов И, И-НЕ из группы логических элементов (5).

Задание: собрать схему для исследования работы синхронного RS – триггера, приведенную на рис. 3.2.1.



Рис. 3.2.1 – Принципиальная схема для исследования работы синхронного RS- триггера Запустить моделирование, заполнить таблицу переходов (табл.3), по таблице переходов получить временную диаграмму (ВД).

Входы			Вы	іходы
S	R	С	Q	nQ
0	0	0	Заполнить	Заполнить
1	0	1	Заполнить	Заполнить
0	1	1	Заполнить	Заполнить
1	1	1		X

Таблица 2.2.	Таблица	переходов	синхронного	RS-T	оиггера
--------------	---------	-----------	-------------	------	---------

4.Сравнение полученных результатов

На этом этапе урока учащимся предлагается устно сравнить полученные результаты (таблицы переходов) и сделать выводы о проделанной работе.

5. Контрольные вопросы:

1. Объясните принцип работы RS-триггера с прямым управлением.

- 2.Объясните принцип работы RS-триггера с инверсным управленим.
- 3. Объясните принцип работы синхронного RS-триггера.

Содержание отчета:

1. Схемы асинхронного и синхронного RS-триггера.

2. Временные диаграммы триггеров.

3. Таблицы переходов.

4. Ответы на контрольные вопросы.

Использованная литература:

1. Амелина М.А., Амелин С.А. Программа схемотехнического моделирования Місго-Сар Версии 9,10.

2. Разевиг В.Д. «Система схемотехнического моделирования Micro-Cap V» - М.: 1997г.

3. Потехин В.А. Схемотехника цифровых устройств: учебное пособие – Томск: В-Спектр, 2012.

РАЗВИТИЕ ТВОРЧЕСКИХ ВОЗМОЖНОСТЕЙ ПРИ ВЫПОЛНЕНИИ ЛАБОРАТОРНЫХ И ПРАКТИЧЕСКИХ РАБОТЫ

Овсейчук	Татьяна
Васильевна	
Г	
П	
0	
\boldsymbol{Y}	
F	

Енакиевский металлургический

У Льюиса Кэрролла в «Алисе в Зазеркалье» ЧернеаниКурродение поворатель для того, чтобы оставаться на месте, надо бежать со всех ног*специи сциятивно стещий*ы и повытаться вперед, нужно бежать вдвое быстрее. И студенты бегуквалификационной категории

Они тратят уйму времени на написание конспектов лекций, не всегда совершенных, выполнение лабораторных работ, не всегда осмысленных. Многие интерес к учебе и стремятся лишь получить зачет

Очень хочется изменить настоящее, чтобы сделать лучше и интереснее будущее не всех, а каждого. Творчество, инженерное творчество Обучение творчеству можно начать с выполнения лабораторных работ по дисциплине «Информационные технологии» по специальности 09.02.03 Программирование в компьютерных системах.

При изучении темы «Создание функции пользователя на языке Visual Basic в среде Excel» Студент выполняет практическую и лабораторную работу.

Практическая работа выполняется на уроке как средство закрепления полученных теоретических знаний.

ПРАКТИЧЕСКОЕ РАБОТА СОЗДАНИЕ ФУНКЦИИ ПОЛЬЗОВАТЕЛЯ НА ЯЗЫКЕ VISUAL BASIC В СРЕДЕ EXCEL

Для того чтобы попасть в редактор VBA, выберите команду

1. Разработчик→ Visual Basic или нажмите комбинацию клавиш Alt + F11.

2. Для того чтобы добавить стандартный модуль. впроект, выберите команду Insert, Module.

3. перейти к написанию функции пользователя. В общем случае, функция пользователя имеет следующий вид:

Function ИмяФункции (СписокПараметров)

Инструкции

End Function Создать следующую функцию

> Function aaa(f) aaa = 2 * f End Function

Проверить ее работу в EXCEL

Лабораторная работа выполняется на следующем занятии и должна служить не столько средством закрепления определённых навыков и знаний, но выполнять собственную роль как средства развития мышления студента.

ЛАБОРАТОРНАЯ РАБОТА СОЗДАНИЕ ФУНКЦИИ ПОЛЬЗОВАТЕЛЯ НА ЯЗЫКЕ VISUAL BASIC В СРЕДЕ EXCEL

- 1. В табличном редакторе Excel записать условие своего варианта как объект Word.
- 2. Исходные данные записываются в ячейке (или ячейках) Excel. Создать не менее двух функций пользователя для получения результата. Вариант студент выбирает самостоятельно. Оценка выставляется в зависимости от набранных баллов. Нельзя выполнять более двух вариантов одной степени сложности.

Степень	Номера	Максимальное количество
сложности	вариантов	бал лов за вариант
1	1,2,3,4	3
2	5,6,7,8,9	4
3	10,11,12	6
4	13,14,15,16	8
5	17	12

Общее количество	оценка
баллов	
6-7	3
8-12	4
13-16	5

Номера вариантов

- 1. Дано целое число в диапазоне 1–7. Вывести строку название дня недели, соответствующее данному числу (1 «понедельник», 2 «вторник» и т. д.).
- Дано целое число К. Вывести строку-описание оценки, соответствующей числу К (1 «плохо», 2 «неудовлетворительно», 3 «удовлетворительно», 4 «хорошо», 5 «отлично»). Если К не лежит в диапазоне 1–5, то вывести строку «ошибка».
- Дан номер месяца целое число в диапазоне 1–12 (1 январь, 2 февраль и т. д.). Вывести название соответствующего времени года («зима», «весна», «лето», «осень»).
- 4°. Дан номер месяца целое число в диапазоне 1–12 (1 январь, 2 февраль и т. д.). Определить количество дней в этом месяце для невисокосного года.
- 5. Арифметические действия над числами пронумерованы следующим образом: 1 сложение, 2 вычитание, 3 умножение, 4 деление. Дан номер действия N (целое число в диапазоне 1–4) и вещественные числа A и B (B не равно 0). Выполнить над числами указанное действие и вывести результат.
- Единицы длины пронумерованы следующим образом: 1 дециметр, 2 — километр, 3 — метр, 4 — миллиметр, 5 — сантиметр. Дан номер единицы длины (целое число в диапазоне 1–5) и длина отрезка в этих единицах (вещественное число). Найти длину отрезка в метрах.
- Единицы массы пронумерованы следующим образом: 1 килограмм, 2 миллиграмм, 3 — грамм, 4 — тонна, 5 — центнер. Дан номер единицы массы (целое число в диапазоне 1–5) и масса тела в этих единицах (вещественное число). Найти массу тела в килограммах.
- 8. Даны два целых числа: *D* (день) и *M* (месяц), определяющие правильную дату невисокосного года. Вывести значения *D* и *M* для даты, предшествующей указанной.
- 9°. Даны два целых числа: *D* (день) и *M* (месяц), определяющие правильную дату невисокосного года. Вывести значения *D* и *M* для даты, следующей за указанной.
- 10. Элементы окружности пронумерованы следующим образом: 1 радиус R, 2 диаметр $D = 2 \cdot R$, 3 длина $L = 2 \cdot \Box \cdot R$, 4 площадь круга $S = \Box \cdot R^2$. Дан номер одного из этих

элементов и его значение. Вывести значения остальных элементов данной окружности (в том же порядке). В качестве значения 🗆 использовать 3.14.

- 11. Элементы равнобедренного прямоугольного треугольника пронумерованы следующим образом: 1 катет *a*, 2 гипотенуза $c = a\sqrt{2}$, 3 высота *h*, опущенная на гипотенузу (h = c/2), 4 площадь $S = c \cdot h/2$. Дан номер одного из этих элементов и его значение. Вывести значения остальных элементов данного треугольника (в том же порядке).
- 12. Элементы равностороннего треугольника пронумерованы следующим образом: 1 сторона *a*, 2 радиус R_1 вписанной окружности ($R_1 = a\sqrt{3}/6$), 3 радиус R_2 описанной окружности ($R_2 = 2 \cdot R_1$), 4 площадь $S = a^2\sqrt{3}/4$. Дан номер одного из этих элементов и его значение. Вывести значения остальных элементов данного треугольника (в том же порядке).
- Дано целое число в диапазоне 20–69, определяющее возраст (в годах). Вывести строкуописание указанного возраста, обеспечив правильное согласование числа со словом «год», например: 20 — «двадцать лет», 32 — «тридцать два года», 41 — «сорок один год».
- 14. Дано целое число в диапазоне 10–40, определяющее количество учебных заданий по некоторой теме. Вывести строку-описание указанного количества заданий, обеспечив правильное согласование числа со словами «учебное задание», например: 18 — «восемнадцать учебных заданий»,

23 — «двадцать три учебных задания», 31 — «тридцать одно учебное задание».

- 15. Дано целое число в диапазоне 100–999. Вывести строку-описание данного числа, например: 256 «двести пятьдесят шесть», 814 «восемьсот четырнадцать».
- 16. Дано целое число в диапазоне 1–100, определяющее сумму (в гривнях). Вывести строкуописание указанной суммы, обеспечив правильное согласование числа со словом «гривня», например: 20 — «двадцать гривен», 32 — «тридцать две гривни», 41 — «сорок одна гривня».
- 17. В восточном календаре принят 60-летний цикл, состоящий из 12-летних подциклов, обозначаемых названиями цвета: зеленый, красный, желтый, белый и черный. В каждом подцикле годы носят названия животных: крысы, коровы, тигра, зайца, дракона, змеи, лошади, овцы, обезьяны, курицы, собаки и свиньи. По номеру года определить его название, если 1984 год — начало цикла: «год зеленой крысы».

ХОД РАБОТЫ

1. Загрузите .Excel

2. Убедитесь, что окно Excel полностью не развернуто. На экране должно оставаться свободное место.

3. чтобы попасть в редактор VBA, выберите команду

Сервис→Макрос→Редактор Visual Basic или нажмите комбинацию клавиш <Alt>+<F11>.

4. Для того чтобы добавить стандартный модуль в проект, выберите команду Insert, Module.

- 5. Напишите функции пользователя в окне кода
- 6. Занесите исходные данные на лист Excel, используя функцию пользователя

проверьте результат

7. Занесите код функции пользователя в отчет.

В этой работе студенту предоставляется возможность самому оценитвать степень сложности задания, свои знания и возможности

Пример выполнения лабораторной работы .на рисунке 1

Рис.1

Лабораторная работа № Тема: Создание функции пользователя на языке Visual Basic в среде Excel для решения математических задач Цель: Использование средств табличного редактора Excel для создания функции пользователя на языке Visual Basic. Задание:1 Дано целое число от 1-7. Вывести строку-название дня недели, соответствующее данному числу (1-понедельник и т.д.) Код: Function ДниНедели(d) If d = 5 Then ДниНедели = "Патница" If d = 1 Then ДниНедели = "Понедельник" If d = 2 Then ДниНедели = "Вторник" If d = 6 Then ДниНедели = "Суббота" If d = 7 Then ДниНедели = "Воскресенье" . 32 If d = 3 Then ДниНедели = "Среда" If d < 1 And d > 7 Then ДниНедели = "Ошибка" If d = 4 Then ДниНедели = "Четверг" End Function Задание 2 Дано целое число К. Вывести строку-описание оценки, соответствующей числу К(1-"плохо" и т.д.) если К не в диапазоне от 1-5 - вывести "Ошибка" Код: Function Оценка(К) If K = 4 Then Оценка = "Хорошо" 70 If K = 1 Then Оценка = "Плохо" If K = 2 Then Оценка = "Неудовлетворительно" If K = 5 Then Оценка = "Отлично" If K < 1 And a > 5 Then Оценка = "Ошибка" If K = 3 Then Оценка = "Удовлетворительно" End Function Залание 13 Дано целоечисло в диапазоне от 20-69, определяющее возраст(в годах). Вывести строку-описание возраста, описание: 20-"Двадцать лет" и т.д. 20 доадцать на н.д. Function Boasac(первый, символ, второй _символ) If первый_символ = 2 Then a = "Двадцать" If первый_символ = 3 Then a = "Тридцать" If первый_символ = 5 Then a = "Сорок" If первый_символ = 5 Then a = "Шестьдесят" If первый_символ = 6 Then a = "Шестьдесят" If aropoil_cumson = 0, then a = the total if aropoil_cumson = 1 then = $q^{(0)}Q\mu H^{(0)}$ if aropoil_cumson = 2 then q = "Aea"if aropoil_cumson = 3 then q = "Tpu"if aropoil_cumson = 5 then q = "Inats"If второй_символ = 6 Then q = "Шесть" If второй_символ = 6 Then q = "Шесть" If второй_символ = 7 Then q = "Семь" If второй_символ = 8 Then q = "Восемь" If второй символ = 9 Then g = "Девять" If второй_символ = 2 Then P = "года" If второй_символ = 3 Then P = "года" If второй_символ = 4 Then P = "года" If второй символ = 1 Then P = "год" If второй символ = 5 Then P = "лет" If второй символ = 5 Then P = "лет" If второй символ = 6 Then P = "лет" If второй_символ = 7 Then P = "лет" If второй_символ = 8 Then P = "лет" If второй_символ = 9 Then P = "лет" f = a + j + d + j + P Возраст = f Bospact = f End Functio erddad dawlet аторой римация = 2 Thus and manifest and them ETERNA CHARDER # 2 THE ETODOM_CONVIRCED = SiTUAL () ETODOM_DOMNBOLIES THEORE STARST CONVERSE & Then I's STORON CHARGEST F CHARTE Выполнил: Салий И Проверила: Овсейчук Т.В

ЛАБОРАТОРНАЯ РАБОТА № 6 «ТАБЛИЧНЫЙ ПРОЦЕССОР ЕХСЕL. ПРИМЕНЕНИЕ ЛОГИЧЕСКИХ, МАТЕМАТИЧЕСКИХ И СТАТИСТИЧЕСКИХ ФУНКЦИЙ» ПО ДИСЦИПЛИНЕ «ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ПРОФЕССИОНАЛЬНОЙ ДЕЯТЕЛЬНОСТИ»

Чигринец Анна Викторовна ГПОУ «Шахтерский техникум кино и телевидения имени А.А.Ханжонкова», преподаватель дисциплины «Информационное обеспечение профессиональной деятельности», специалист высшей квалификационной категории

Тема занятияИзучение возможностей табличного процессора ExcelВид занятияпрактическоеТип занятиязанятие совершенствования знаний, умений и навыков

Формы обучения индивидуальная

Цель занятия: приобретение навыков работы со встроенными функциями текстового процессора Excel.

Формируемые компетенции

ОК 2. Организовывать собственную деятельность, определять методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.

ОК 4. Осуществлять поиск, анализ и оценку информации, необходимой для постановки и решения профессиональных задач, профессионального и личностного развития.

ОК 5. Использовать информационно-коммуникационные технологии для совершенствования профессиональной деятельности.

Оборудование. ПК, OC WINDOWS, табличный процессор EXEL.

Межпредметные связи Основы теории информации, Планирование и организация работы структурного подразделения, Метрология стандартизация и сертификация

Список литературы:

1. Михеева Е.В. Практикум по информационным технологиям в профессиональной деятельности, 5-е издание. М.: Издательский центр «Академия», 2006.

2. Практикум по информатике: Уч. Пособие/ А. В. Могилев, Н. И. Пак, Е. К. Хеннер; под ред. Е. К. Хеннера.-М.: ИЦ «Академия», 2005.

3. Симонович С. В., Евсеев Г. А.: Windows: лаборатория мастера: Практическое руководство по эффективным приемам работы с компьютером.- М.: АСТ-ПРЕСС КНИГА, 2002г.

4. Гарнаев А.Ю. Excel, VBA, Internet в экономике и финансах. – СПб.: БХВ – Петербург, 2001.

ИНСТРУКТИВНО-МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ВЫПОЛНЕНИЮ ЛАБОРАТОРНОЙ РАБОТЫ № 6 «ТАБЛИЧНЫЙ ПРОЦЕССОР ЕХСЕL. ПРИМЕНЕНИЕ ЛОГИЧЕСКИХ, МАТЕМАТИЧЕСКИХ И СТАТИСТИЧЕСКИХ ФУНКЦИЙ»

Теоретические сведения

Статистические функции:

Табличный процессор EXEL содержит много встроенных функций. Для того, чтобы вызвать одну из них необходимо выделить ячейку, в которой будет храниться результат, щелкнуть по значку f_x и в открывшемся диалоговом окне выбрать интересующую Вас функцию, затем следуя подсказкам мастера ввести данные.



Функция МАКС

Возвращает наибольше значение из набора значений.

<u>Синтаксис:</u> МАКС(число1;число2; ...)

Пример 1. Если ячейки А1:А5 содержат числа 10, 7, 9, 27 и 2, то:

МАКС(А1:А5) равняется 27

<u>Пример 2.</u> МАКС(А1:А5;30) равняется 30

Функция СРЗНАЧ

Возвращает среднее (арифметическое) своих аргументов.

<u>Синтаксис:</u> СРЗНАЧ(число1; число2; ...)

Пример. Если ячейки А1:А5 содержат числа 10, 7, 9, 27 и 2, то:

СРЗНАЧ(А1:А5) равняется 11

СРЗНАЧ(А1:А5; 5) равняется 10

Функция СУММЕСЛИ

Суммирует ячейки, специфицированные заданным критерием.

<u>Синтаксис</u>: СУММЕСЛИ(интервал; критерий; сумм_интервал)

<u>Интервал</u> это ячейки, в которых проверяется критерий.

Критерий это заданное условие.

<u>Сумм интервал</u> это ячейки, в которых производится суммирование согласно условию.

<u>Пример.</u> Пусть ячейки A1:A4 содержат следующие числа: 10, 20, 30, 40 соответственно. Пусть ячейки B1:B4 содержат следующие числа: 7, 14, 21, 28.

СУММЕСЛИ(А1:А4;">16";В1:В4) равняется 63.

Функция СЧЕТ

Подсчитывает количество чисел в списке аргументов. Функция СЧЁТ используется для получения количества числовых ячеек в интервалах или массивах ячеек.

<u>Синтаксис:</u> СЧЁТ(значение1; значение2; ...)

<u>Значение1, значение2,</u> ... - это от 1 до 30 аргументов, которые могут содержать или ссылаться на данные различных типов, но в подсчете участвуют только числа.

Функция СЧЕТЕСЛИ

Подсчитывает количество ячеек внутри интервала, удовлетворяющих заданному критерию.

<u>Синтаксис:</u> СЧЁТЕСЛИ(интервал;критерий)

<u>Интервал</u> – заданный ряд данных.

<u>Критерий</u> – проверяемый критерий в заданном интервале (число, выражение, текст). <u>Пример 1.</u> Пусть ячейки АЗ:А6 содержат "отличник", "хорошист", "троечник", "хорошист" соответственно.

СЧЁТЕСЛИ(АЗ:А6; "хорошист") равняется 2.

Функция РАНГ Возвращает ранг числа в списке чисел. Ранг числа — это его величина относительно других значений в списке. (Если отсортировать список, то ранг числа будет его позицией.)

<u>Синтаксис:</u> РАНГ(число;ссылка;[порядок])

<u>Число</u> Число, для которого определяется ранг.

<u>Ссылка</u> Массив или ссылка на список чисел. Нечисловые значения в ссылке игнорируются.

<u>Порядок</u> Необязательный. Число, определяющее способ упорядочения.

Если значение аргумента "порядок" равно 0 или опущено, ранг числа определяется в Microsoft Excel так, как если бы ссылка была списком, отсортированным в порядке убывания.

Если значение аргумента "порядок" — любое число, кроме нуля, то ранг числа определяется в Microsoft Excel так, как если бы ссылка была списком, отсортированным в порядке возрастания.

<u>Логические функции:</u>

Функция ЕСЛИ

Возвращает одно значение, если заданное условие при вычислении дает значение ИСТИНА, и другое значение, если ЛОЖЬ.

Функция ЕСЛИ используется для условной проверки значений и формул и имеет следующий <u>синтаксис</u>: ЕСЛИ(лог_выражение; значение_истина; значение_ложь) Эта функция позволяет осуществить выбор варианта вычислений. Если значение логического выражения истинно, то выбирается <u>значение_истина</u>; иначе- значение_ложь. В качестве логического выражения могут применяться в сложных случаях функции «И» и «ИЛИ». Также может вставляться вложенная функция ЕСЛИ.

<u>Пример 1</u>. ЕСЛИ(А10=1;СУММ(В5:В15);"нет")

В этом примере, если значение ячейки A10 = 1, то *лог_выражение* имеет значение ИСТИНА и вычисляется сумма для ячеек B5:B15. В противном случае *лог_выражение* имеет значение ЛОЖЬ и возвращается текст ("нет").

Использование функций И – ИЛИ:

ЕСЛИ(И(A1<2;A2>5); A2;A3), где A1 содержит число 5, A2 - число 10, A3 - число 8. Ответ равен 8.

ЕСЛИ(ИЛИ(A1<2;A2>5); A2;A3), то ответ 10.

Задание.

1. Оформите заглавную строку таблицы, содержащей следующие столбцы: №, Фамилия студента, Математика, Физика, История, Русский язык, Средний балл, Минимальная оценка, Максимальная оценка, Наличие стипендии, Рейтинг студента.

2. Заполните первую строку таблицы следующими данными:

1 Иванов 4 3 4 3.

3. В ячейку средний балл внесите статистическую функцию СРЗНАЧ- которая посчитает среднее значение баллов по четырем предметам студента Иванова.

4. Аналогично внесите формулы в столбцы Минимальная оценка и Максимальная оценка. (Статистические функции МИН и МАКС)

5. В столбец Наличие стипендии введите логическую функцию ЕСЛИ, которая в зависимости от минимального балла студента будет выдавать слова – «есть», если минимальный балл больше либо равен 4, и «нет» в других случаях.

6. Оформите рамками таблицу на 20 студентов. Продублируйте формулы в столбцах: Средний балл, Минимальная оценка, Максимальная оценка, Наличие стипендии.

7. При помощи условного форматирования, выделите все оценки ниже трех - красным цветом.

8. В столбец Рейтинг студента вставьте формулу Ранг – по среднему баллу студента определит его место в группе. Не забудьте об относительных и абсолютных адресах для копирования формулы.

9. Заполните первый 6 строк таблицы произвольными данными. Проверьте, как работают Ваши формулы.

10. Под двадцатью оформленными для студентов строками, сделайте итоговую строку, в которой в столбцах: Математика, Физика, История, Русский язык – подсчитайте средний балл по этим предметам.

11. В этой же строке, в столбце Наличие стипендии расположите формулу, которая подсчитает количество студентов имеющих стипендию.

12. Под таблицей в ячейке B25 – поместите текст «всего студентов», в ячейке H25 – поместите формулу, которая в зависимости от количества набранных в таблице фамилий будет подсчитывать количество студентов.

13. Вбейте данные еще по двум студентам. Проверьте работу формул.

14. В ячейке В27 – поместите текст «Процент, студентов получающих стипендию», в ячейке С27 – формулу, подсчета процента студентов получающих стипендию (количество студентов, получающих стипендию разделить на общее количество студентов в группе), выполните формат этой ячейки в процентах.

ЛАБОРАТОРНАЯ РАБОТА № 16 «РАБОТА С ФОРМАМИ» ПО ДИСЦИПЛИНЕ «WEB-ПРОГРАММИРОВАНИЕ»

Скрипий Алексей Александрович ГПОУ «Шахтерский техникум кино и телевидения имени А.А.Ханжонкова», преподаватель спецдисциплин, специалист первой квалификационной категории

Тема занятия Работа с формами

Вид занятия практическое

Тип занятия занятие совершенствования знаний, умений и навыков **Формы обучения** индивидуальная

Цели занятия:

Дидактическая: закрепить, углубить, систематизировать знания студентов по теме «Язык программирования PHP»; получить практические навыки *Методическая:* способствовать формированию практических умений в использовании методов передачи данных из формы на сервер и их обработки средствами языка PHP.

Воспитательная: вырабатывать умение мыслить самостоятельно, воспитывать внимательность, дисциплинированность.

Формируемые компетенции:

 администрировать сетевые ресурсы в информационных системах в (ПК 2.2.);
 обеспечивать сбор данных для анализа использования и функционирования программно-технических средств компьютерных сетей. (ПК 2.3.);

- устанавливать, настраивать, эксплуатировать и обслуживать технические и программно-аппаратные средства компьютерных сетей. (ПК 3.1.).

Оборудование. демонстрационный монитор, ПК, ПК для студентов по количеству присутствующих, локальный сервер, программное обеспечение OpenServer. Межпредметные связи _____ обеспечивающие – математика, информатика, программирование; обеспечиваемые – программное обеспечение компьютерных сетей.

Список литературы:

1. Фрейн Б. HTML5 и CSS3.Разработка сайтов для любых браузеров и устройств /Б.Фрейн - СПб.: «Питер», 2013. - 304 с.

2. Дмитриева М. JavaScript. Быстрый старт. / М.Дмитриева - СПб.: БХВ Санкт-Перебург, 2002. – 335 с.

3. Веллинг Л., Разработка веб-приложений с помощью PHP и MySQL. / Люк Веллинг, Лаура Томсон - М.: Вильямс, 2010. – 848с.

4. Прохоренок Н. А. HTML, JavaScript, PHP и MySQL. Джентльменский набор Webмастера 3-е изд., перераб. и доп. / Н. А.Прохоренок — СПб.: БХВ-Петербург, 2010. - 912 с.:

Лабораторная работа № 16 Работа с формами

Цель работы: Создать форму для оформления заказа средствами HTML, JavaScript, PHP.

Приборы и оборудование:

- Персональный компьютер;
- OC Windows;
- Локальный сервер;
- Текстовый редактор NotePad++;
- OpenServer.

Задание:

Создать форму заказа товара следующего вида:

Выберите и закажите наш товар!!!

Выберите интересующие Вас товары:

Шахматы	2
Шашки	
Нарды	
	4

Язык интерфейса:

Выберите из списка 🔻

Baш e-mail:^{*}

Пароль:*

Отправить

Дополнительно:

- 🗹 Подписаться на новости сайта
- Получать персональные сообщения на e-mail
- Подписаться на рекламу от наших спонсоров

В форме должны присутствовать два скрытых поля

- uID (User ID) содержащее произвольное шестизначное число;
- queryDate Содержащее текущую дату и время.

Также необходимо:

• реализовать проверку на наличие заполнения обязательных полей и выводить сообщение, если не все они заполнены;

Форма должна отправлять данные в файл getData2.php, в котором должна производиться проверка полученных данных на корректность и выводится строка об успешном получении заказа.

Пример желаемого вывода:

Уважаемый заказчик! Введённый Вами пароль совпал с тем, что хранится в нашей базе данных. Ваш заказ успешно зарегистрирован в системе в 29.11.2016 20:49. Как только он будет обработан оператором, на адрес Вашей электронной почты будет выслано извещение! Подробности заказа: Дата: 29.11.2016 20:49 Вы выбрали 1 товар: Шахматы

Язык интерфейса: Выберите из списка

Дополнительно вы: Подписались на новости сайта Подписались на рекламу от наших спонсоров

Ход работы:

- 1. Создайте документ index.php со стандартными тегами <html>, <head>, <body>
- 2. Используя теги <h1>, <i>, <h4>,
 создайте заголовок формы заказа;
- **3.** Создайте форму для заполнения заказа с помощью тега *<form name="forma" action = "getData2.php" method = "get" onSubmit=validate_form()>*, где :
- name="forma" имя формы,
- *action = "getData2.php" имя файла в который будут передаваться данные;*
- *method* = "get" метод, с помощью которого будут передаваться данные (может быть GET или POST);
- onSubmit=validate_form() имя JavaScript, которое будет вызываться при отправке формы для проверки на наличие заполнения обязательных полей.
 Не забываем закрыть тег в конце документа!!!!
 - 4. Создайте Список множественного выбора, который позволяет в выпадающем меню выбрать несколько элементов для выбора заказываемых товаров:

<select type="select" multiple id="goodsList" name="goodsList[]" > <option value="Шахматы" selected="selected" > Шахматы</option> <option value="Шашки" > Шашки</option> <option value="Hapды" > Hapды</option> </select> Выбор нескольких элементов производится при удержании кнопок **Ctrl + Shift.**

5. Создайте обычный список для выбора языка:

<select name="lang"><option>Выберите из списка</option> <option>Английский</option> <option>Русский</option> </select></br>

6. Создайте обязательные поля для заполнения E-mail адреса и пароля:

<h4>Baw e-mail:^{*}</h4> <input type="text" maxlength="20" name="someInputName">
 <h4>Пароль:^{*}</h4> <input type="password" maxlength="8" name="pass">

7. Создайте кнопку «Отправить»:

<input type="submit" value="Отправить" validate_form()><hr>

8. Создайте дополнительные поля для оформления подписок с помощью группы флажков:

```
<h4>Дополнительно:</h4>
```

<input type="checkbox" name="addChksArr[]" value="su1" checked >Подписаться на новости caŭma

<input type="checkbox" name="addChksArr[]" value="su2" >Получать персональные сообщения на e-mail

<input type="checkbox" name="addChksArr[]" value="su3" checked >Подписаться на рекламу от наших спонсоров

9. Создайте дополнительные скрытые поля для передачи номера ID клиента и времени отправления формы:

```
<input type="hidden" name="uID" value="215745" /><br />
<input type="hidden" name="queryDate" value="<?php date("d.m.Y G:i")?>" /><br />
В первом поле передается переменная 215745, а во втором поле передается дата и время.
Эти поля в форме не отображаются.
```

10. Не забудьте сохранить файл и закрыть тег <form>.

11. Напишите на JavaScript функцию проверки обязательных для заполнения полей (адрес и пароль):

```
<script type="text/javascript">
<!--
function validate form ()
{
       valid = true;
    if ( document.forma.pass.value == "" )
    {
         alert ( "Пожалуйста заполните поле 'Ваш пароль'.");
        valid = false;
    }
    if ( document.forma.someInputName.value == "" )
    {
         alert ( "Пожалуйста заполните поле 'Ваш e-mail'.");
         valid = false:
    }
    return valid;
}
//-->
```

Поместите эту функцию в тег <head>. Функция вызывается в пункте 3.

- **12.** Создайте новый файл под названием *getData2.php* в этой-же директории. Если файл будет находиться в другой папке, то необходимо указывать относительный путь к этой папке. Данные в этот файл передаются в пункте 3.
- 13. С помощью функций РНР получите данные из формы и передайте их в переменные РНР:

```
<?php
$sel = $_GET['goodsList'];
$lang = $_GET["lang"];
$email = $_GET['someInputName'];
$pass = $_GET['pass'];?>
```

14. С помощью функции count получаем количество выбранных товаров:

```
<?$nsel = count($sel);?>
```

15. Пишем функцию для проверки отмеченных в конце формы полей для оформления подписок:

```
<?function IsChecked($chkname,$value)
{
    if(!empty($_GET[$chkname]))
    {
        foreach($_GET[$chkname] as $chkval)
        {
            if($chkval == $value)
            {
               return true;
            }
        }
        return false;
    }
}</pre>
```

16. Приступаем к созданию формы вывода результатов заказа:

```
<i>Уважаемый заказчик!<br>
```

```
Введённый Вами пароль совпал с тем, что хранится в нашей базе данных.<br>
Ваш заказ успешно зарегистрирован в системе в <?echo date("d.m.Y G:i");?>.<br> Как только<br/>он будет обработан оператором, на адрес Вашей электронной почты будет выслано<br/>извещение!<br>
Подробности заказа:<br>
Дата: <?echo date("d.m.Y G:i");?><br>
Функция <?echo date("d.m.Y G:i");?> возвращает дату и время заказа<br/>
17. Выводим в сообщение количество и список выбранных товаров:<br/>
<?echo("<p>Вы выбрали $nsel товар:<br/>
($i=0; $i < $nsel; $i++)
```

```
{
  echo($sel[$i] . " <br>");
}
echo("");?>
```

\$nsel – количество выбранных товаров;

\$sel[\$i] – массив, содержащий список выбранных товаров.

18. Выводим язык:

Язык интерфейса: <?echo \$lang?>

19. Выводим поля, отмеченные для оформления подписок:

Дополнительно вы:

<?if(IsChecked('addChksArr','su1'))

echo "Подписались на новости сайта
";

if(IsChecked('addChksArr','su2'))

echo "Дали разрешение получать персональные сообщения по e-mail
";

if(IsChecked('addChksArr','su3'))

echo "Подписались на рекламу от наших спонсоров
";?>

</i>

20. Сохраняем файлы и проверяем их на работоспособность!!!!

Контрольные вопросы:

- 1. Какие методы передачи данных наиболее часто используются?
- 2. Какую функцию выполняет скрипт в данной программе и можно-ли его заменить с помощью функции PHP?
- 3. Для чего используется функция count?
- 4. С помощью какой функции мы можем просмотреть все элементы массива в РНР?

Отчет должен содержать:

- 1. Листинг двух файлов index.php и getData2.php.
- 2. В конце файла getData2.php необходимо добавить ответы на контрольные вопросы в формате HTML.
- 3. Файлы index.php и getData2.php поместить в свою папку на локальном сервере.

ЛАБОРАТОРНАЯ РАБОТА № 10 «ИСПОЛЬЗОВАНИЕ ЛИНЕЙНОГО БЛОЧНОГО КОДИРОВАНИЯ» ПО ДИСЦИПЛИНЕ «ОСНОВЫ ТЕОРИИ ИНФОРМАЦИИ»

Чубарьян Андрей Сергеевич ГПОУ «Шахтерский техникум кино и телевидения имени А.А.Ханжонкова», преподаватель дисциплины «Основы теории информации», специалист второй квалификационной категории

Тема занятия Использование линейного блочного кодирования

Вид занятия лабораторная работа

Тип занятия контроль и корекция знаний, умений и навыков

Формы обучения индивидуальная

Цель занятия: приобретение практических навыков использования линейного блочного кодирования на примере кода Хэмминга.

Формируемые компетенции

ПК 1.1. Обеспечивать защиту информации в сети с использованием программноаппаратных средств.

Оборудование. Персональный компьютер, операционная система Windows XP или более поздняя версия, прикладная программа «hemming».

Межпредметные связи: Информатика, Технология физического уромня передачи данных, Безопасность функционирования информационных систем, Математический аппарат построения компьютерных сетей, Элементы математической логики.

Список литературы:

1. Литвинская О.С. Основы теории передачи информации: учебное пособие / О.С. Литвинская, Н.И. Чернышёв. – М. : КНОРУС, 2010. – 168 с.

2. Хохлов Г.И. Основы теории информации: учебное пособие для студентов высших учебных заведений / Г.И. Хохлов. – М.: Издательский центр «Академия», 2008. – 176с.

ИНСТРУКТИВНО-МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ВЫПОЛНЕНИЮ ЛАБОРАТОРНОЙ РАБОТЫ №10

«ИСПОЛЬЗОВАНИЕ ЛИНЕЙНОГО БЛОЧНОГО КОДИРОВАНИЯ» Теоретические сведения

В реальных условиях приём двоичных символов всегда происходит с ошибками, когда вместо символа "1" принимается символ "0" и наоборот. Ошибки могут возникать изза помех, действующих в канале связи (особенно помех импульсного характера), изменения за время передачи характеристик канала (например, замирания), снижения уровня передачи, нестабильности амплитудно- и фазочастотных характеристик канала и т.п.

Общепринятым критерием оценки качества передачи в дискретных каналах является нормированная на знак или символ допустимая вероятность ошибки для данного вида сообщений. Так, допустимая вероятность ошибки при телеграфной связи может составлять 10^{-3} (на знак), а при передаче данных – не более 10^{-6} (на символ). Для обеспечения таких значений вероятностей одного улучшения только качественных показателей канала связи может оказаться недостаточным. Поэтому основной мерой является применение специальных методов повышения качества приёма передаваемой информации. Эти методы можно разбить на две группы.

К первой группе относятся методы увеличения помехоустойчивости приёма единичных элементов (символов) дискретной информации, связанные с выбором уровня сигнала, отношения сигнал-помеха (энергетические характеристики), ширины полосы канала, методов приёма и т.д.

Ко второй группе относятся методы обнаружения и исправления ошибок, основанные на искусственном введении избыточности в передаваемое сообщение.

Помехоустойчивые (корректирующие) коды делятся на блочные и непрерывные.

Блочными называются коды, в которых информационный поток символов разбивается на отрезки и каждый из них преобразуется в определённую последовательность (блок) кодовых символов. В блочных кодах кодирование при передаче (формирование проверочных элементов) и декодирование при приёме (обнаружение и исправление ошибок) выполняются в пределах каждой кодовой комбинации (блока) в отдельности по соответствующим алгоритмам.

Непрерывные или рекуррентные коды образуют последовательность символов, не разделяемую на отдельные кодовые комбинации. Кодирование и декодирование непрерывно совершаются над последовательностью элементов без деления их на блоки. Формирование проверочных символов ведётся по рекуррентным (возвратным) правилам, поэтому непрерывные коды часто называют рекуррентными или цепными.

В простейшем цепном коде каждый проверочный элемент формируется путём сложения по модулю 2 соседних или отстоящих друг от друга на определённое число позиций информационных элементов. В канал связи передаётся последовательность импульсов, в которой за каждым информационным следует проверочный. Подобную чередующуюся последовательность разрядов имеет, например, корреляционный манчестерский код.

К непрерывным кодам относятся и свёрточные коды, в которых каждый информационный символ, поступающий на вход кодирующего устройства, вызывает появление на его выходе ряда проверочных элементов, образованных суммированием по модулю 2 данного символа и " k-1 " предыдущих информационных символов. Рекуррентные коды позволяют исправлять групповые ошибки (" пачки ") в каналах связи.

Блочные коды делятся на равномерные и неравномерные. В равномерных кодах, в отличие от неравномерных, все кодовые комбинации содержат одинаковое число n - символов (разрядов) с постоянной длительностью т0 импульсов символов кода. Равномерные коды в основном и применяются в системах связи, так как это упрощает технику передачи и приёма.

94

Классическими примерами неравномерного кода являются код Морзе, широко применяемый в телеграфии, и код Хафмена, применяемый для компрессии информации (факсимильная связь, ЭВМ).

Систематические коды образуют наиболее обширную группу разделимых кодов. Особенностью этих кодов является то, что проверочные (корректирующие) символы образуются с помощью линейных операций над информационными. Кроме того, любая разрешённая кодовая комбинация может быть получена в результате линейной операции над набором линейно независимых кодовых комбинаций. В частности, суммирование по модулю 2 двух и более разрешённых комбинаций также дает разрешённую кодовую комбинацию. Поскольку теоретической основой получения таких комбинаций является математический аппарат линейной алгебры, то коды и называют линейными, а учитывая, что проверочные символы формируются по определённой системе (правилам), блочные равномерные разделимые линейные коды получили название систематических. Использование аппарата линейной алгебры, в которой важное значение имеет понятие "группа", породило и другое название этих кодов - групповые. Эти коды получили наибольшее применение в системах передачи дискретной информации.

Несистематические (нелинейные) коды указанными выше свойствами не обладают и применяются значительно реже в специальных случаях. Эти коды обычно используются в несимметричных каналах связи, в которых вероятность перехода 1 к 0 значительно больше вероятности перехода 0 к 1 или наоборот. В таких каналах очень маловероятно, чтобы в одном блоке были переходы обоих видов, и поэтому почти все ошибки приводят к изменению веса блока, и, следовательно, обнаруживаются.

Наиболее известны среди систематических кодов коды Хемминга, которые исторически были найдены раньше многих других кодов и сыграли большую роль в развитии теории корректирующих кодов. В этих кодах используется принцип проверки на чётность определённого ряда информационных символов. Проверочная группа из г символов формируется поэлементно по соответствующему алгоритму. Коды Хемминга, имеющие dmin = 3, позволяют исправить одну ошибку.

Циклические коды также относятся к классу линейных систематических кодов и обладают всеми их свойствами. Коды названы циклическими потому, что циклический сдвиг любой разрешённой кодовой комбинации также является разрешённой комбинацией. Теория построения циклических кодов базируется на разделах высшей алгебры, изучающей свойства двоичных многочленов. Особую роль в этой теории играют так называемые неприводимые многочлены, т.е. полиномы, которые не могут быть представлены в виде произведения многочленов низших степеней. В связи с этим циклические коды относят к разновидности полиномиальных кодов.

Построение кодов Хемминга базируется на принципе проверки на чётность веса W (числа единичных символов) в информационной группе кодового блока.

В таком коде к кодовым комбинациям без избыточного первичного двоичного k разрядного кода добавляется один дополнительный разряд (символ проверки на чётность, называемый проверочным, или контрольным). Если число символов "1" исходной кодовой комбинации чётное, то в дополнительном разряде формируют контрольный символ 0, а если число символов "1" нечётное, то в дополнительном разряде формируют символ 1. В результате общее число символов "1" в любой передаваемой кодовой комбинации всегда будет чётным.

Таким образом, правило формирования проверочного символа сводится к следующему:

 $r1 = i1 \bigoplus i2 \bigoplus \dots \bigoplus ik$,

где i - соответствующий информационный символ (0 или 1), k - общее их число, а под операцией "⊕" здесь и далее понимается сложение по mod2. Очевидно, что добавление дополнительного разряда увеличивает общее число возможных комбинаций вдвое по сравнению с числом комбинаций исходного первичного кода, а условие чётности разделяет все комбинации на разрешённые и неразрешённые. Код с проверкой на чётность позволяет обнаруживать одиночную ошибку при приёме кодовой комбинации, так как такая ошибка нарушает условие чётности, переводя разрешённую комбинацию в запрещённую.

Критерием правильности принятой комбинации является равенство нулю результата S суммирования по mod 2 всех n символов кода, включая проверочный символ r1. При наличии одиночной ошибки S принимает значение 1:

 $S = r1 \bigoplus i1 \bigoplus i2 \bigoplus ... \bigoplus ik = 0$ - ошибки нет,

= 1 - однократная ошибка.

Этот код является (k +1, k) - кодом, или (n, n -1) - кодом. Минимальное расстояние кода равно двум (d min = 2), и, следовательно, никакие ошибки не могут быть исправлены. Простой код с проверкой на чётность может использоваться только для обнаружения (но не исправления) однократных ошибок.

Увеличивая число дополнительных проверочных разрядов и формируя по определённым правилам проверочные символы г, равные 0 или 1, можно усилить корректирующие свойства кода так, чтобы он позволял не только обнаруживать, но и исправлять ошибки. На этом и основано построение кодов Хемминга.

Для каждого числа проверочных символов r = 3,4,5... существует классический код Хемминга. При других значениях числа информационных символов k получаются так называемые усечённые (укороченные) коды Хемминга.

Для примера рассмотрим классический код Хемминга (7,4), который можно сформировать и описать с помощью кодера, представленного на рис.1. В простейшем варианте при заданных четырёх (k=4) информационных символах (i1, i2, i3, i4) будем полагать, что они сгруппированы в начале кодового слова, хотя это и не обязательно. Дополним эти информационные символы тремя проверочными символами (r = 3), задавая их следующими равенствами проверки на чётность, которые определяются соответствующими алгоритмами:

r1 = i1 \oplus i2 \oplus i3; r2 = i2 \oplus i3 \oplus i4; r3 = i1 \oplus i2 \oplus i4, где знак \oplus означает сложение по модулю 2. В декодере в режиме исправления ошибок строится последовательность: s1 = r1' \oplus i1' \oplus i2' \oplus i3'; s2 = r2' \oplus i2' \oplus i3' \oplus i4'; $s3 = r3' \oplus i1' \oplus i2' \oplus i4'.$

Трёхсимвольная последовательность (s1, s2, s3) называется синдромом. В данном случае синдром S = (s1, s2, s3) представляет собой сочетание результатов проверки на чётность соответствующих символов кодовой группы и характеризует определённую конфигурацию ошибок (шумовой вектор).

к = 4				r = 3			
İ1	i ₂	İ3	i4	r 1	r ₂	r ₃	
0	0	0	0	0	0	0	j.
0	0	0	1	0	1	1	
0	0	1	0	1	1	0	
0	0	1	1	1	0	1	
0	1	0	0	1	1	1	
0	1	0	1	1	0	0	
0	1	1	0	0	0	1	
0	1	1	1	0	1	0	
1	0	0	0	1	0	1	
1	0	0	1	1	1	0	
1	0	1	0	0	1	1	
1	0	1	1	0	0	0	
1	1	0	0	0	1	0	
1	1	0	1	0	0	1	
1	1	1	0	1	0	0	
1	1	1	0	1	1	1	

Кодовые слова (7,4) - кода Хемминга.

При числе проверочных символов r = 3 имеется восемь возможных синдромов (23 = 8). Нулевой синдром (000) указывает на то, что ошибки при приёме отсутствуют или не обнаружены. Всякому ненулевому синдрому соответствует определённая конфигурация ошибок, которая и исправляется. Классические коды Хемминга (3.20) имеют число синдромов, точно равное их необходимому числу, позволяют исправить все однократные ошибки в любом информативном и проверочном символах и включают один нулевой синдром. Такие коды называются плотноупакованными.

Таким образом, код (7,4) позволяет исправить все одиночные ошибки. Простая проверка показывает, что каждая из ошибок имеет свой единственный синдром. При этом возможно создание такого цифрового корректора ошибок (дешифратора синдрома), который по соответствующему синдрому исправляет соответствующий символ в принятой кодовой группе. После внесения исправления проверочные символы гі можно на выход декодера не выводить. Две или более ошибки превышают возможности корректирующего кода Хемминга, и декодер будет ошибаться. Это означает, что он будет вносить неправильные исправления и выдавать искажённые информационные символы.

Задание на работу и содержание отчета

1. Запустите программное обеспечение «hemming.exe». В результате появится главное окно программы с тремя кнопками:

K	Сод Хэмм	инга
Дем	онстрация раб	оты кода
	Задание 1	
	Задание 2	
[ЗАКРЫТН	b

2. Нажмите на кнопку «Демонстрация работы кода» - запуститься соответствующее окно программы с примером работы кода Хэмминга для 8 информационных символов:

		0000	000									
12 1	11	10	9	8	7	6	5	4	3	2	1	L
<u></u>	#	×2	#	<u>8</u> _0	#	<u> 2</u> 2	#	<u></u>	#	<u>21-1</u> 2	<u>s_</u> :	k1
	#	#			#	#			#			k2
#					#	#	#					k4
#	#	#	#									<u>k8</u>

3. Произведите пошаговое рассмотрение работы кода Хэмминга с занесением в отчёт действий кода.

4. После выполнения демонстрационной части закройте это окно (кнопка с «шагами» обеспечит эту возможность).

5. Перейдите к пункту «Задание № 1»:

🛞 Задание № 1	
ЗАДАНИЕ № 1	
Кодовая последовательность: 101101011110 Бит с ошибкой: Исходное число: ПРОВЕРИТЬ	

В первом задании необходимо определить для заданной кодовой последовательности (полученной, используя код Хэмминга (8,4)) номер бита, который является ошибочным. Затем произвести корректировку и найти исходное число (осуществить перевод из двоичной системы счисления в десятичную). Эти данные ввести в соответствующие поля программы и нажать на кнопку «ПРОВЕРИТЬ» и в случае правильного решения можно закрыть данное окно. В случае неправильного ввода – повторите попытку. Все данные и расчеты занести в отчёт.

6. Перейдите к пункту «Задание № 2»:

🛞 Задание № 2						
ЗАДАНИЕ № 2						
Исходное число:	187					
Кодовая последовательность: 101111011110						
0 1 1 1 1 0 1	1 1 1 0					
Исправленный код	000000000000000000000000000000000000000					
Полученок число: ВЫПОЛНИТЬ						

В этом задании вам необходимо произвести исследование возможности кода Хэмминга. Для начала нажмите на кнопку «ВЫПОЛНИТЬ» и убедитесь в том, что программа сообщит об отсутствии ошибок. Затем произведите внесение ошибок в полях и посмотрите на результат. На основе данного исследования заполните следующую таблицу:

№	Исходное число	Кодовая последовательность	Количество и номера ошибок	Исправленный код	Полученное число
1	2	3	4	5	6

Для заполнения таблицы введите 1, 2, 3, 4 ошибки.

7. Закройте окно программы «Задание № 2». Закройте главное окно программы.

8. Подготовьте отчёт.