

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БАШКОРТОСТАН
ГБПОУ СТЕРЛИТАМАКСКИЙ МЕЖОТРАСЛЕВОЙ КОЛЛЕДЖ**

УТВЕРЖДАЮ

Зам. директора по учебно-
производственной работе

_____ Исангулов Р.Р.

«_____» _____ 201__ г.

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ
ПО КУРСОВОМУ ПРОЕКТИРОВАНИЮ
МДК 01.02. ПРИКЛАДНОЕ ПРОГРАММИРОВАНИЕ
ПО СПЕЦИАЛЬНОСТИ 09.02.03
ПРОГРАММИРОВАНИЕ В КОМПЬЮТЕРНЫХ СИСТЕМАХ
3 КУРС**

Методические рекомендации по выполнению курсового проекта разработаны преподавателем спец.дисциплин Т.Г.Минибаевой.

Методические указания обсуждены и одобрены на заседании МЦК «Экономических, математических и общих естественнонаучных дисциплин» от _____ 201__ г., протокол № ____.

СОДЕРЖАНИЕ

1.	Общие положения	4
2.	Содержание разделов курсовой работы (проекта)	6
3.	Примерная тематика курсовых проектов	10
	Приложение А. Титульный лист	38
	Приложение В. Задание на разработку работы (проекта)	39

1. Общие положения

Курсовая работа по программированию в среде Delphi выполняется студентами третьего курса в шестом семестре.

Целью работы является выработка у студентов практических навыков по проектированию программ, их отладке и документированию.

Выполнение курсовой работы начинается с разработки технического задания (ТЗ) и завершается составлением отчета, в котором должно содержаться описание всей проделанной работы.

Данные цели проявляются через следующие конкретные задачи курсовой работы:

- систематизация, закрепление, углубление и расширение теоретических знаний, полученных при изучении МДК 01.02 Прикладное программирование, а также приобретение практических навыков решения комплексных задач;

- привитие навыков самостоятельной работы по подбору литературы, работы с научной литературой и иными информационными источниками;

- умение самостоятельно систематизировать и излагать знания, полученные в процессе самостоятельного изучения литературы;

- привитие навыков научно-исследовательской работы, использование анализа и самостоятельных выводов.

В результате выполнения курсовой работы студент должен *научиться*:

- создавать программу в среде Delphi 7.0 в соответствии с основными этапами ее разработки;

- строить схему алгоритма работы программы в соответствии с требованиями ГОСТ 19.701-90;

- грамотно тестировать программу;

- анализировать результаты работы программы и делать выводы.

Курсовой проект является самостоятельной работой студента, позволяет оценить качество знаний и отражает приобретенные студентом практические навыки. Также позволяет расширить объем знаний студентов в

области программирования и создать реальную основу использования своих знаний для решения задач с помощью компьютера по другим дисциплинам и в своей дальнейшей практической деятельности.

Тема назначается руководителем курсового проекта и утверждается на заседании предметной цикловой комиссии.

Перед студентом ставится задача разработать приложение с целью решения конкретной задачи. Результатом решения является:

- А) исполняемый файл программы;
- Б) пояснительная записка.

Для решения поставленной задачи студенту необходимо предварительно ознакомиться с литературой, посвященной теме задания. При этом следует обратить внимание на средства, используемые для решения аналогичных задач или для решения каких-либо ключевых моментов задачи.

Сформулированные в настоящих указаниях задания на курсовой проект представляют студенту простор для творчества. В текстах задач умышленно опущены некоторые детали и необходимые требования. После ознакомления с литературой студент должен оценить возможности языка программирования и вычислительной техники, на которой предлагается реализовать решение. Результатом этой работы должна быть точная формулировка задачи со всеми ограничениями и требованиями.

При решении задачи необходимо придерживаться техники пошаговой детализации, использовать стандартные структуры, не забывая при этом о развитии программного окружения программиста, расширяя возможности языка за счет включения новых процедур и функций.

При разработке алгоритма необходимо предусмотреть средства, проверки и тестирования программы, удобство работы пользователя, возможные модификации.

При написании программы не следует забывать о хорошем стиле программирования, о таких понятиях, как читаемость, эффективность, надежность. Необходимо искать наиболее простые и естественные приемы.

2. Содержание разделов курсовой работы (проекта)

Все этапы разработки программы отражаются в пояснительной записке. Пояснительная записка состоит из следующих разделов:

Титульный лист

Содержание

Задание на разработку курсовой работы (проекта)

Введение

Постановочная часть

1.1. Постановка задачи.

1.2. Описание алгоритма.

2. Проектная часть

2.1. Написание программы

2.2. Описание программы.

ЗАКЛЮЧЕНИЕ.

Список использованной литературы.

ПРИЛОЖЕНИЕ. Листинг программы.

ОГЛАВЛЕНИЕ. Оглавление составляется в соответствии с содержанием пояснительной записки и должно отражать все разделы курсовой работы. После написания пояснительной записки в оглавлении проставляются страницы.

ВВЕДЕНИЕ. В разделе "Введение" указывается тема и цель курсового проекта. А также указывают назначение и краткую характеристику области применения программного курсового проекта.

1. Постановочная часть.

1.1. Постановка задачи. Решение задачи начинается с ее постановки. Дается точное описание исходных данных, условий задачи и целей ее решения. На этом этапе условия задачи, записанные в форме различных словесных описаний, необходимо выразить на формальном языке математики.

Обычно математическая модель - это набор уравнений, неравенств и ограничений, приближенно описывающих задачу. При построении математической модели отбрасываются некоторые свойства реальной задачи, мало влияющие на решение.

В этом разделе могут быть описаны основные приемы программирования и типы данных, используемые при решении аналогичных задач. Например, если в задаче используются динамические структуры, то перечисляются виды динамических структур данных и основные процедуры по работе с динамическими структурами. Если задача заключается в формировании базы данных и дальнейшей работе с базой, то приводится описание используемых типов данных (характеристика данных записного типа) и приемы работы с файлами.

Далее описываются возможные пути решения задачи с указанием их достоинств и недостатков. Выбирается и обосновывается метод решения задачи. Описываются ограничения, накладываемые на исходные данные, необходимая разрядность и точность представления исходных данных и результатов решения. Указываются возможные пределы изменения входных параметров задачи.

1.2. Описание алгоритма. В разделе дается обобщенное словесное описание алгоритма решения поставленной задачи, излагаются основные требования к алгоритму и пути их реализации. Приводится схема алгоритма, состоящая из укрупненных модулей. Дается пояснение назначения и состава каждого модуля. Обобщенный алгоритм обычно использует обозначения и термины исходной задачи.

На следующем этапе каждый модуль детализируется. Выделяются укрупненные команды, реализуемые по вспомогательным алгоритмам. Тот же подход применяется при разработке вспомогательных алгоритмов:

2. Проектная часть.

2.1. Написание программы

Современные программы разрабатываются для функционирования в

среде Windows. Приложение для Windows студенты разрабатывают в среде визуального программирования.

Визуальное программирование строится на тесном взаимодействии двух процессов:

- процесс конструирования Windows-окна;
- процесс написания кода, придающего элементам этого окна и программе в целом необходимую функциональность.

Проект Windows-окна должен быть представлен в виде графической схемы, на которой расположены все визуальные и невидимые компоненты, разрабатываемого интерфейса.

Компоненты на схеме должны быть пронумерованы. После схемы приводится расшифровка изображенных на схеме компонентов: название и имя компонента; назначение в программе; события, на которые данный компонент откликается.

Для каждого компонента должны быть указаны свойства, измененные при проектировании окна.

Программная реализация разработанных алгоритмов содержит обработчики событий. На этапе разработки рабочего проекта необходимая степень детализации алгоритмов обычно выбирается такой, чтобы предписания разработанных алгоритмов могли записываться на языке программирования, выбранном для составления текста программы. При детализации алгоритма необходимо перейти к обозначениям, принятым для разработки программ на алгоритмическом языке. При этом имена следует выбирать таким образом, чтобы они отражали сущность используемых параметров.

Кодирование должно быть простым. Изощренное программирование может обойтись слишком дорого при отладке и модификации программы. Необычное кодирование (например, использование скрытых возможностей машины) часто препятствует отладке программы и затрудняет ее использование другими программистами. Программа должна быть по

возможности универсальной. Универсальные программы обеспечивают независимость программы от конкретного набора данных. Например, универсальная программа использует в качестве параметров переменные, а не константы, обрабатывает вырожденные случаи и т. д. Универсальность программы экономит время по дальнейшей работе, над ней и обеспечивает широкие возможности по использованию. Разрабатывая такие программы, можно предвидеть будущие изменения в спецификациях этой программы.

Входные форматы должны быть разработаны с учетом максимального удобства для пользователя и минимальной возможности ошибок. Порядок переменных и форматы данных, привычные для пользователя, помогут избежать ошибок и облегчат использование программ.

При написании программы следует применять операторы, позволяющие использовать основные алгоритмические структуры.

При написании программ не следует забывать о хорошем стиле программирования. После заголовка процедуры или функции записывается комментарий, содержащий поясняющий текст, а именно: назначение подпрограммы; перечень и назначение формальных параметров, их тип. Комментариями должны быть снабжены и основные смысловые блоки программы или подпрограммы. Для облегчения чтения текста программы отдельные операторы программы записываются с отступом.

Примечание. Не следует представлять в виде схем алгоритмов линейные вычислительные процессы. В детализированном виде представляются алгоритмы, имеющие сложную структуру. Сложной структурой будем считать алгоритмы, содержащие более одного разветвления или более одного цикла.

2.2. Описание программы. Этот раздел должен содержать общие сведения, функциональное назначение и описание логической структуры программы. Также в этот раздел можно включить руководство оператора.

В подразделе "Общие сведения" должны быть указаны: обозначение и наименование программы; программное обеспечение, необходимое для

функционирования программы; языки программирования, на которых написана программа.

В подразделе "Функциональное назначение" должны быть указаны классы решаемых задач и (или) назначение программы и сведения о функциональных ограничениях на применение.

В подразделе "Описание логической структуры" должны быть указаны используемые методы; структура программы с описанием функций составных частей и связи между ними; связи программы с другими программами. Описание логической структуры программы выполняют с учетом текста программы на исходном языке.

ЗАКЛЮЧЕНИЕ. В заключении необходимо показать, каким образом решены поставленные задачи курсового проекта и какова степень достижения ее цели. Заключение должно содержать выводы и предложения, обоснованные студентом в процессе выполнения.

В **список использованной литературы** следует включать не только цитируемые и упомянутые произведения, но и те, с которыми студент ознакомился в процессе подготовки работы.

Источники нумеруются для того, чтобы на них можно было ссылаться из текста пояснительной записки. Например:

1. Фаронов В. В. Delphi 7. Учебный курс. -М.: Издательство "Нолидж", 2005.

2. Дарахвелидзе П., Марков Е. Программирование в Delphi 7. - СПб.: БХВ-СПб., 2008.

3. Примерная тематика курсовых проектов

Варианты курсовой работы распределяются следующим образом: первая подгруппа – нечетные задания, по порядку следования списка в журнале , вторая – четные по порядку, следования списка в журнале

1. Создайте программу введения базы данных "Сотрудники" научного учреждения "Прогресс" База данных состоит из двух файлов. Файл №1 - список сотрудников: фамилия, код должности, подразделение. Файл № 2 - справочник: наименование должности, зарплата за_один час работы. Количество отработанных часов для каждого сотрудника вводятся с клавиатуры компьютера. Ведение базы данных включает в себя следующие пункты: ввод и редактирование данных, вывод расчетной ведомости для каждого подразделения и расчет суммы выплаты заработной платы по институту в целом. Расчетная ведомость имеет вид:

№	Фамилия	Начислено	Подходный налог	Отчисления в пенсионный фонд	К выплате
---	---------	-----------	-----------------	------------------------------	-----------

2. Создайте программу ведения базы данных торговой фирмы. Программа включает в себя: формирование и корректирование файлов данных; расчет комиссионного вознаграждений сотрудников фирмы. Файл данных о продавце включает его имя и фамилию, табельный номер, дату поступления на работу. Торговая фирма выплачивает продавцам комиссионное вознаграждение в размере 5%, если товара продано на, сумму менее 1000 долл. в день, и 6%, если выручка составляет 1000 долл./день и выше. Продавцы, проработавшие в фирме более 10 лет, получают комиссионные на 1% больше.

Сумма выручки за день для каждого продавца вводится с клавиатуры ЭВМ. Организуйте вывод общих итогов по сумме выручки и сумм комиссионного вознаграждения за месяц.

3. Разработать приложение "Помощник экзаменатора". Экзаменационные вопросы и ответы к ним хранятся в файлах на диске. Каждый вопрос имеет балл сложности. Необходимо подобрать пять вопросов

из разных разделов курса, имеющих в сумме балл сложности N, и вывести их на экран. Предусмотреть тренировочный режим, работы, когда возможен вывод ответов на представленные вопросы. Доступ к тренировочному режиму работы предоставляется по паролю.

4. Разработать программу ведения базы данных риэлторской фирмы. Данные о продаже квартир хранятся в двух файлах - основном и справочном. Основной файл содержит сведения о сделках и имеет следующие поля: дата сделки, покупатель, идентификационный номер продаваемой квартиры, цена. Файл-справочник содержит следующую информацию об уже проданных квартирах: общую площадь квартиры, число комнат.

Номер записи в файле-справочнике соответствует идентификационному номеру квартиры. Предусмотреть ввод/корректирование исходной информации и вывод ведомости о сумме продаж, совершенных в каком-либо году или в каком-либо месяце. Ведомость о сумме продаж имеет вид:

Месяц	Сумма продаж	По числу комнат в квартире		
		1	2	3
Январь				
Итого:				

5. Напишите программу, которая генерирует или считывает шахматную позицию и определяет, не находится ли один из королей под шахом и не является ли шах матом. В программе предусмотреть два варианта ввода исходных данных: шахматная позиция генерируется с помощью датчиков случайных чисел и шахматная позиция вводится с клавиатуры ЭВМ.

6. Поле шахматной доски задается парой натуральных чисел: первое указывает номер вертикали при счете слева направо, второе - номер Горизонта пи при счете снизу вверх. Расстановка фигур задается таким

образом, что в начале указываются поля, на которых стоят перечисленные белые фигуры, затем - поля, на которых стоят перечисленные черные фигуры.

- На доске стоят два ферзя. Указать поля, на которые может пойти белый ферзь так, чтобы не попасть под удар черного ферзя.

- У белых на доске остался только король, у черных - король, конь, слон. Охарактеризовать положение белых с помощью слов: мат, шах, пат, обыкновенная позиция.

- Получить m расстановок 8 ферзей на шахматной доске, при которых ни один из ферзей не угрожает другому.

7. Напишите программу составления кроссвордов. Исходными данными является конфигурация 6 на 6 (некоторое расположение пустых и заполненных квадратов) и список слов, состоящих из шести или менее букв. Результатом должно быть расположение этих слов, образующее общепринятый кроссворд, или сообщение о том, что такая конфигурация невозможна.

8. Разработать программу, моделирующую игру. Игра имеет следующие правила. Перед Вами большое число ящиков с деньгами. Сумма денег в каждом ящике - случайная величина, равномерно распределенная на отрезке $[0, 1]$. Вы выбираете ящик, открываете его и или берете деньги из ящика, или отказываетесь от них. Если Вы берете деньги, игра кончается. В противном случае Вы можете выбрать другой ящик. Эта процедура повторяется максимум до пяти ящиков (деньги из/пятого ящика должны быть взяты, если он открыт).

9. Разработайте программу, моделирующую игру. Два игрока, "нечетный" и "четный", по очереди ставят единицы и нули в незанятые позиции поля N на N . Каждый из игроков может ставить 1 или 0 в произвольную свободную позицию, тем самым, занимая ее. Игра продолжается до заполнения всех позиций. После этого суммируются числа вдоль каждой строки, каждого столбца и главных диагоналей. Число ODD

нечетных сумм сравнивается с числом EVEN четных сумм. Если $ODD > EVEN$, выигрывает "нечетный"; если $EVEN > ODD$, выигрывает "четный"; если $ODD = EVEN$, результат считается ничейным. Если одним из игроков является ЭВМ, то постройте для нее выигрышную стратегию.

10. Разработать программу, моделирующую игру "Кости". Играющий называет любое число в диапазоне от 2 до 12 и ставку, которую он делает в этот ход. Программа с помощью датчика случайных чисел дважды выбирает числа от 1 до 6 ("бросает кубик", на гранях которого цифры от 1 до 6). Если сумма выпавших цифр меньше 7 и играющий задумал число меньше 7, он выигрывает сделанную ставку. Если сумма выпавших цифр больше 7 и играющий задумал число больше 7, он также выигрывает сделанную ставку. Если играющий угадал сумму цифр, он получает в четыре раза больше очков, чем сделанная ставка. Ставка проиграна, если не имеет место ни одна из описанных ситуаций. В начальный момент у играющего 100 очков. В программе должно присутствовать графическое изображение поверхности кубика при каждом ходе игрока.

11. Разработать программу, моделирующую игру "Морской бой". На поле 10 на 10 позиций стоят невидимые вражеские корабли: 4 корабля по 1 клетке, 3 корабля по 2 клетки, 2 корабля по 3 клетки, 1 корабль в 4 клетки. Необходимо поразить каждую из клеток кораблей. Два игрока вводят позиции кораблей в виде цифр (1, 2, 3, 4) в соответствующие элементы матрицы, тем самым определяя конфигурацию и положение кораблей. Игроки по очереди "наносит удары" по кораблям противника. Если позиция корабля указана верно, то она помечается крестиком на поле. Предусмотреть вариант игры, когда одним из играющих является компьютер.

12. Разработать программу, моделирующую игру "Сбей самолет". По экрану летят вражеские самолеты. Цель играющего — сбить их. Пусковая установка находится в нижней строке экрана. Пусковую установку можно перемещать по строке вперед и назад.

13. Разработать программу, моделирующую игру "Скачки". В игре

участвуют 10 наездников; за каждый тур игры каждый из них продвигается вперед на расстояние от 1 до 5 км случайным образом. Длина дистанции — 50 км. Всего проводится 5 заездов, победителю каждого заезда начисляется 5 очков. Победителем считается наездник, набравший наибольшее количество очков во всех заездах. Перед началом заездов участник игры выбирает номер наездника, с которым он будет идентифицироваться во время игры. Количество участников игры не превышает 10. В каждом туре с вероятностью 0.1 каждый наездник может упасть, т.е. продвинуться за этот тур на ноль км. Передвижение наездников отобразить графически на экране. Предусмотреть возможность случайного распределения номеров наездников.

14. Игра "Морской бой" с компьютером.

15. *Разработка собственного Web - браузера.* Разработать программное средство, реализующее стандартные функции работы с Web-страницами. Реализовать следующие возможности обозревателя:

- выпадающий список, содержащий ранее просмотренные в данном сеансе адреса URL;

- кнопки Назад и Вперед, перемещающие пользователя по списку ранее просмотренных страниц;

- панель состояния, отображающую сведения о загружаемом документе;

- загрузку в первый момент выполнения домашней страницы;

- возможность в любой момент перейти к домашней странице;

- настройку адреса домашней страницы, загружаемой в начале сеанса;

- возможность в любой момент перейти к странице поиска и, возможность настройки адреса этой страницы;

- возможность обновить текущую страницу Web;

- возможность остановить операцию загрузки или просмотра;

16. БД Ресторана

Таблицы:

1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон,

Паспортные данные, Код должности)[10 записей].

2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].

3) Склад (Код ингредиента, Наименование ингредиента, Дата выпуска, Объём, Срок годности, Стоимость, Поставщик)[10 записей].

4) Меню (Код блюда, Наименование блюда, Код ингредиента 1, Объём ингредиента 1, Код ингредиента 2, Объём ингредиента 2, Код ингредиента 3, Объём ингредиента 3, Стоимость, Время приготовления)[10 записей].

5) Заказ (Дата, Время, ФИО заказчика, Телефон, Код блюда 1, Код блюда 2, Код блюда 3, Стоимость, Отметка о выполнении, Код сотрудника)[10 записей].

Запросы:

1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по полю "Код должности").

2) Меню (Связывает таблицы "Меню" и "Склад" по полям "Код ингредиента", "Код ингредиента 1", "Код ингредиента 2" и "Код ингредиента 3").

3) Заказ (Связывает таблицы "Заказ", "Меню" и "Сотрудники" по полям "Код блюда", "Код блюда 1", "Код блюда 2", "Код блюда 3" и "Код сотрудника").

Фильтры:

1) Фильтры для отображения сотрудников отдельных должностей (На основе запроса "Отдел кадров").

2) Фильтры отображения ингредиентов определённых поставщиков (На основе таблицы "Склад").

3) Фильтры выполненных и невыполненных заказов (На основе запроса "Заказы").

17. БД Банка

Таблицы:

- 1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности)[10 записей].
- 2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].
- 3) Вклады (Код вклада, Наименование вклада, Минимальный срок вклада, Минимальная сумма вклада, Код валюты, Процентная ставка, Дополнительные условия)[5 записей].
- 4) Валюта (Код валюты, Наименование, Обменный курс)[3 записи].
- 5) Вкладчики (ФИО вкладчика, Адрес, Телефон, Паспортные данные, Дата вклада, Дата возврата, Код вклада, Сумма вклада, Сумма возврата, Отметка о возврате вклада, Код сотрудника)[10 записей].

Запросы:

- 1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по полю "Код должности").
- 2) Вклады (Связывает таблицы "Вклады" и "Валюта" по полю "Код валюты").
- 3) Вкладчики (Связывает таблицы "Вкладчики", "Вклады" и "Сотрудники" по полям "Код вклада" и "Код сотрудника").

Фильтры:

- 1) Фильтры для отображения сотрудников отдельных должностей (На основе запроса "Отдел кадров").
- 2) Фильтры для отображения вкладов отдельных валют (На основе запроса "Вклады").
- 3) Фильтры для отображения вкладчиков с отдельными вкладами (На основе запроса "Вкладчики").
- 4) Фильтры для отображения возвращённых и невозвращённых вкладов

(На основе запроса "Вкладчики").

18. БД Больницы.

Таблицы:

1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности)[10 записей].

2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].

3) Лекарства (Код лекарства, Наименование, Показания, Противопоказания, Упаковка, Стоимость)[5 записей].

4) Болезни (Код болезни, Наименование, Симптомы, Продолжительность, Последствия, Код лекарства 1, Код лекарства 2, Код лекарства 3)[10 записей].

5) Пациенты (ФИО пациента, Возраст, Пол, Адрес, Телефон, Дата обращения, Код болезни, Код сотрудника, Результат лечения)[10 записей].

Запросы:

1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по полю "Код должности").

2) Болезни (Связывает таблицы "Болезни" и "Лекарства" по полю "Код лекарства", "Код лекарства 1", "Код лекарства 2" и "Код лекарства 3").

3) Пациенты (Связывает таблицы "Пациенты", "Болезни" и "Сотрудники" по полям "Код болезни" и "Код сотрудника").

Фильтры:

1) Фильтры для отображения сотрудников отдельных должностей (На основе запроса "Отдел кадров").

2) Фильтры для отображения болезней с одинаковыми симптомами (На основе запроса "Болезни").

3) Фильтры для отображения пациентов с одинаковыми болезнями (На

основе запроса "Пациенты").

19. БД Гостиницы.

Таблицы:

1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности)[10 записей].

2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].

3) Номера (Код номера, Наименование, Вместимость, Описание, Стоимость, Код сотрудника)[5 записей].

4) Услуги (Код услуги, Наименование, Описание, Стоимость)[5 записей].

5) Клиенты (ФИО, Паспортные данные, Дата заселения, Дата выезда, Код номера, Код услуги 1, Код услуги 2, Код услуги 3, Стоимость, Код сотрудника)

Запросы:

1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по полю "Код должности").

2) Номера (Связывает таблицы "Сотрудники" и "Номера" по полю "Код сотрудника").

3) Клиенты (Связывает таблицы "Клиенты", "Номера", "Услуги" и "Сотрудники" по полям "Код номера", "Код услуги", "Код услуги 1", "Код услуги 2", "Код услуги 3" и "Код сотрудника").

Фильтры:

1) Фильтры для отображения сотрудников отдельных должностей (На основе запроса "Отдел кадров").

2) Фильтры для отображения клиентов проживающих в разных номерах (На основе запроса "Клиенты").

3) Вывести номера различной вместимости (На основе запроса

“Номера”).

20. БД Аэропорта.

Таблицы:

1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности)[10 записей].

2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].

3) Самолёты (Код самолёта, Марка, Вместимость, Грузоподъемность, Код типа, Технические характеристики, Дата выпуска, Налётано часов, Дата последнего ремонта, Код сотрудника)[5 записей].

4) Типы самолётов (Код типа, Наименование, Назначение, Ограничения).

5) Экипажи (Код экипажа, Налётано часов, Код сотрудника 1, Код сотрудника 2, Код сотрудника 3)[5 записей].

6) Рейсы (Код рейса, Дата, Время, Откуда, Куда, Код экипажа, Код самолёта, Время полёта)[5 записей].

7) Билеты (ФИО пассажира, Паспортные данные, Место, Код рейса, Цена)

Запросы:

1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по полю "Код должности").

2) Самолёты (Связывает таблицы “Самолёты”, “Типы самолётов” и “Сотрудники” по полям “Код типа” и “Код сотрудника”)

3) Экипажи (Связывает таблицы “Экипажи” и “Сотрудники” по полям “Код сотрудника” “Код сотрудника 1”, “Код сотрудника 2” и “Код сотрудника 3”)

4) Рейсы (Связывает таблицы “Рейсы”, “Самолёты” и “Экипажи” по полям “Код экипажа” и “Код самолёта”)

5) Билеты (Связывает таблицы “Билеты” и “Рейсы” по полю “Код рейса”)

Фильтры:

1) Фильтры для отображения сотрудников отдельных должностей (На основе запроса "Отдел кадров").

2) Фильтры для отображения самолётов разных типов (На основе запроса "Самолёты").

3) Фильтры для отображения билетов отдельных рейсов (На основе запроса "Билеты").

21. БД Библиотеки.

Таблицы:

1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности)[10 записей].

2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].

3) Издательства (Код издательства, Наименование, Город, Адрес)[5 записей].

4) Жанры (Код жанра, Наименование, Описание) [5 записей].

5) Книги (Код книги, Наименование, Автор, Код издательства, Год издания, Код жанра) [10 записей].

6) Читатели (Код читателя, ФИО, Дата рождения, Пол, Адрес, Телефон, Паспортные данные) [10 записей].

7) Выданные книги (Код книги, Код читателя, Дата выдачи, Дата возврата, Отметка о возврате, Код сотрудника) [10 записей].

Запросы:

1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по

полю "Код должности").

2) Каталог (Связывает таблицы "Книги", "Издательства" и "Жанры" по полям "Код издательства" и "Код жанра").

3) Книги на руках (Связывает таблицы "Выданные книги", "Книги", "Читатели" и "Сотрудники" по полям "Код книги", "Код читателя" и "Код сотрудника")

Фильтры:

1) Фильтры для отображения сотрудников отдельных должностей (На основе запроса "Отдел кадров").

2) Фильтры для отображения книг отдельных издательств (На основе запроса "Каталог").

3) Фильтры для отображения книг отдельных авторов (На основе запроса "Каталог").

4) Фильтры для отображения книг отдельных годов издания (На основе запроса "Каталог").

5) Фильтры для отображения сданных и не сданных книг (На основе запроса " Книги на руках ").

6) Фильтры для отображения книг на руках отдельных читателей (На основе запроса " Книги на руках ").

22. БД Радиостанции.

Таблицы:

1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности)[10 записей].

2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].

3) Исполнители (Код исполнителя, Наименование, Описание)[5 записей].

4) Жанры (Код жанра, Наименование, Описание)[5 записей].

5) Записи (Код записи, Наименование, Код исполнителя, Альбом, Год, Код жанра, Дата записи, Длительность, Рейтинг)[10 записей].

6) График работы (Дата, Код сотрудника, Время 1, Код записи 1, Время 2, Код записи 2, Время 3, Код записи 3)[10 записей].

Запросы:

1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по полю "Код должности").

2) Музыкальный архив (Связывает таблицы "Записи", "Исполнители" и "Жанры" по полям "Код исполнителя" и "Код жанра").

3) Сетка вещания (Связывает таблицы "График работы", "Сотрудники" и "Записи" по полям "Код сотрудника", "Код записи", "Код записи 1", "Код записи 2" и "Код записи 3").

Фильтры:

1) Фильтры для отображения сотрудников отдельных должностей (На основе запроса "Отдел кадров").

2) Фильтры для отображения записей отдельных исполнителей (На основе запроса "Музыкальный архив").

3) Фильтры для отображения записей отдельных жанров (На основе запроса "Музыкальный архив").

4) Фильтры сетки вещания по отдельным датам (На основе запроса "Сетка вещания").

5) Фильтры сетки вещания по отдельным сотрудникам (На основе запроса "Сетка вещания").

23. БД Туристического агентства.

Таблицы:

1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности)[10 записей].

2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].

3) Виды отдыха (Код вида, Наименование, Описание, Ограничения)[5 записей].

4) Отели (Код отеля, Наименование, Страна, Город, Адрес, Телефон, Количество звёзд, Контактное лицо)[10 записей].

5) Дополнительные услуги (Код услуги, Наименование, Описание, Цена) [5 записей].

6) Клиенты (Код клиента, ФИО, Дата рождения, Пол, Адрес, Телефон, Паспортные данные)[5 записей].

7) Путёвки (Дата начала, Дата окончания, Продолжительность, Код отеля, Код вида, Код услуги 1, Код услуги 2, Код услуги 3, Код клиента, Код сотрудника, Отметка о бронировании, Отметка об оплате)[10 записей].

Запросы:

1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по полю "Код должности").

2) Список путёвок (Связывает таблицы "Путёвки", "Отели", "Виды отдыха", "Дополнительные услуги", "Клиенты" и "Сотрудники" по полям "Код отеля", "Код вида", "Код услуги", "Код услуги 1", "Код услуги 2", "Код услуги 3", "Код клиента" и "Код сотрудника").

Фильтры:

1) Фильтры для отображения сотрудников отдельных должностей (На основе запроса "Отдел кадров").

2) Фильтры для отображения путёвок по отдельным видам отдыха (На основе запроса "Список путёвок").

3) Фильтры для отображения путёвок в отдельные страны (На основе запроса "Список путёвок").

4) Фильтры для отображения путёвок в отдельные отели (На основе

запроса “Список путёвок”).

5) Фильтры для отображения забронированных и не забронированных путёвок (На основе запроса “Список путёвок”).

6) Фильтры для отображения оплаченных и не оплаченных путёвок (На основе запроса “Список путёвок”).

7) Фильтры для отображения заказанных и не заказанных путёвок (На основе запроса “Список путёвок”).

24. БД Страховой компании.

Таблицы:

1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности)[10 записей].

2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].

3) Риски (Код риска, Наименование, Описание, Средняя вероятность)[5 записей].

4) Виды полисов (Код вида полиса, Наименование, Описание, Условия, Код риска 1, Код риска 2, Код риска 3)[5 записей].

5) Группы клиентов (Код группы, Наименование, Описание)[5 записей].

6) Клиенты (Код клиента, ФИО, Дата рождения, Пол, Адрес, Телефон, Паспортные данные, Код группы)[10 записей].

7) Полисы (Номер полиса, Дата начала, Дата окончания, Стоимость, Сумма выплаты, Код вида полиса, Отметка о выплате, Отметка об окончании, Код клиента, Код сотрудника)[10 записей].

Запросы:

1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по полю "Код должности").

2) Риски полисов (Связывает таблицы “Виды полисов” и “Риски” по

полям “Код риска”, “Код риска 1”, “Код риска 2”, “Код риска 3”).

3) Список клиентов (Связывает таблицы “Клиенты” и “Группы клиентов” по полю “Код группы”).

4) Список полисов (Связывает таблицы “Полисы”, “Виды полисов”, “Клиенты” и “Сотрудники” по полям “Код вида полиса”, “Код клиента” и “Код сотрудника”).

Фильтры:

1) Фильтры для отображения сотрудников отдельных должностей (На основе запроса "Отдел кадров").

2) Фильтры для отображения клиентов отдельных групп (На основе запроса “Список клиентов”).

3) Фильтры для отображения полисов отдельных видов (На основе запроса “Список полисов”).

4) Фильтры для отображения полисов по которым производились выплаты и по которым не производились выплаты (На основе запроса “Список полисов”).

5) Фильтры для оконченных и неоконченных полисов (На основе запроса “Список полисов”).

25. БД Сервис центра.

Таблицы:

1) Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности)[10 записей].

2) Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования)[5 записей].

3) Запчасти (Код запчасти, Наименование, Функции, Цена)[5 записей].

4) Ремонтируемые модели (Код модели, Наименование, Тип, Производитель, Технические характеристики, Особенности)[5 записей].

5) Виды неисправностей (Код вида, Код модели, Описание, Симптомы,

Методы ремонта, Код запчасти 1, Код запчасти 2, Код запчасти 3, Цена работы)[5 записей].

6) Обслуживаемые магазины (Код магазина, Наименование, Адрес, Телефон)[5 записей].

7) Заказы (Дата заказа, Дата возврата, ФИО заказчика, Серийный номер, Код вида неисправности, Код магазина, Отметка о гарантии, Срок гарантии ремонта, Цена, Код сотрудника)[10 записей].

Запросы:

1) Отдел кадров (Связывает таблицы "Сотрудники" и "Должности" по полю "Код должности").

2) Список неисправностей (Связывает таблицы "Виды неисправностей", "Ремонтируемые модели" и "Запчасти" по полям "Код модели", "Код запчасти", "Код запчасти 1", "Код запчасти 2", "Код запчасти 3").

3) Список заказов (Связывает таблицы "Заказы", "Виды неисправностей", "Обслуживаемые магазины" и "Сотрудники" по полям "Код вида неисправности", "Код магазина" и "Код сотрудника").

Фильтры:

1) Фильтры для отображения сотрудников отдельных должностей (На основе запроса "Отдел кадров").

2) Фильтры для отображения видов неисправностей отдельных моделей (На основе запроса "Список неисправностей").

3) Фильтры для отображения видов неисправностей отдельных типов устройств (На основе запроса "Список неисправностей").

4) Фильтры для отображения видов неисправностей моделей отдельных производителей (На основе запроса "Список неисправностей").

5) Фильтры для отображения заказов отдельных магазинов (На основе запроса "Список заказов").

6) Фильтры для отображения заказов отдельных неисправностей (На основе запроса “Список заказов”).

7) Фильтры для отображения гарантийных и не гарантийных заказов (На основе запроса “Список заказов”).

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БАШКОРТОСТАН
ГБОУ СПО «СТЕРЛИТАМАКСКИЙ СЕЛЬСКОХОЗЯЙСТВЕННЫЙ
ТЕХНИКУМ»

УТВЕРЖДАЮ

Зам. директора по УПР

_____ Исангулов Р.Р..

« ____ » _____ 2014 г.

**Методические рекомендации по выполнению и защите
курсовой работы (проекта)**

Рассмотрены и одобрены на
заседании МЦК

Протокол № _____ от _____ 2014 г.

1. Общие указания по выполнению курсового проекта (работы)

Выполнение курсового проекта (работы) занимает важное место в подготовке высококвалифицированных специалистов, поскольку ее написание способствует глубокому изучению МДК 01.02. Прикладное программирование, включенных в процесс обучения.

Курсовой проект (работа) должен содержать иллюстративные материалы (схемы, чертежи, диаграммы, рисунки и т.п.). Курсовые проекты (работы), выполненные только на основе литературных источников, без привлечения практических материалов, не будут приниматься.

По объему курсовая работа (проект) должна быть не менее 20-25 страниц печатного текста.

Практика написания студентами курсовых работ (проектов) показывает, что весь процесс подготовки, выполнения и защиты состоит из ряда последовательных этапов:

- выбор темы и согласование ее с преподавателем;
- подбор литературы;
- изучение требований к оформлению работы;
- изучение подобранной литературы;
- написание работы;
- завершение работы и представление ее на кафедру;
- разработка тезисов доклада для защиты;
- защита работы.

Перечисленные этапы не равнозначны по своей сложности и по количеству затрачиваемого времени на выполнение каждого этапа.

2. Требования к оформлению курсового проекта (работы)

Курсовой проект (работа) оформляется на стандартных листах формата А4. Сокращение слов, кроме общепринятых, не допускается. Текст излагается от третьего лица или в форме безличных предложений без использования личных местоимений.

Слова «СОДЕРЖАНИЕ», «ВВЕДЕНИЕ», «ЗАКЛЮЧЕНИЕ», «СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ» записывают по центру прописными буквами, включают в содержание курсовой работы. Данные заголовки не нумеруют.

К оформлению работы предъявляются следующие требования:

- 1) текст размещается по ширине листа;
- 2) поля: слева 30 мм, справа, снизу и сверху – 10 мм;
- 3) Межстрочный интервал не более 1,5 (допускается использование междустрочного интервала 1,0 в таблицах и рисунках)
- 4) Шрифт Times New Roman, размер шрифта – 14 пунктов.
- 5) номер страницы проставляется внизу, справа, шрифт 10 пунктов;
- 6) абзац в тексте следует начинать с отступа, равного 1,25 см;
- 7) каждая глава начинается с новой страницы;
- 8) все заголовки глав и параграфов должны быть выделены полужирным шрифтом и без точки;
- 9) сокращения слов в таблицах и рисунках не допускается (разрешается в таблицах и рисунках необходимые надписи делать более мелким шрифтом, чем в текстовой части).

Порядок нумерации работы следующий. На первой странице располагается титульный лист (номер страницы не ставится). Затем постранично (последовательно) размещаются задание на разработку курсового проекта (работы), содержание (образец содержания в приложении А). Нумерация страниц работы начинается с пятого листа (начало введения) и ведется постранично до конца, включая список литературы и приложения.

Все нумерации (страниц, глав, параграфов, рисунков и т.д.) выполняют

только арабскими цифрами. При необходимости в оглавление и, соответственно, в основную часть вводят рубрикации типа 1.1. или 2., 2.2., 2.2.1. и т.п.

Требования к оформлению заголовков

Основная часть курсовой работы (проекта) состоит из разделов и подразделов (при необходимости). Разделы должны иметь порядковые номера в пределах всей курсовой работы, обозначенные арабскими цифрами.

Главы нумеруются цифрами 1, 2 и т.д.

Номера разделов включают номер главы и номер раздела в главе, которые разделяются точкой 1.1, 1.2, 2.1, 2.2 и т.д.

Название глав, разделов должно располагаться с красной строки.

Заголовки разделов и подразделов следует записывать посередине текста, с прописной буквы, не подчеркивая. Переносы слов в заголовках не допускаются. Если заголовок состоит из двух предложений, их разделяют точкой.

Каждая глава начинается с новой страницы.

Требования к оформлению списков

В случае необходимости создания списка его пункты нумеруются арабскими цифрами с круглой скобкой, например 1), 2), 3) и т.д., а подпункты буквами кириллицы с круглой скобкой, например, а), б), в) и т.д..

После скобки текст также начинается с маленькой буквы, а пункты и подпункты между собой разделяются точкой с запятой, при этом красная строка по подпункту должна быть смещена на 1,0 ... 1,25 см относительно красной строки пункта.

Требования к оформлению формул

Формулы, содержащиеся в курсовой работе (проекте), располагают на отдельных строках, нумеруют сквозной нумерацией арабскими цифрами, которые записывают на уровне формулы справа в круглых скобках. Одну

формулу обозначают (1).

Непосредственно под формулой приводится расшифровка символов и числовых коэффициентов, если они не были пояснены ранее в тексте. Первая строка расшифровки начинается словом «где» без двоеточия после него. Выше и ниже каждой формулы должно быть оставлено не менее одной свободной строки.

Формулы рекомендуется оформлять в Редакторе формул.

На все формулы должны быть ссылки, которые оформляются следующим образом: «... рассчитывается по формуле (5)» или «см. формулу (4)».

Требования к оформлению таблиц и рисунков

Таблицы и рисунки в тексте располагаются непосредственно после первой ссылки на них. Они должны отделяться от текста одной пустой строкой и сверху, и снизу; допускается их расположение на отдельной странице, следующей после страницы, где делается первая ссылка на данную таблицу или рисунок.

Все таблицы должны иметь порядковый номер и название, отражающее содержание. Слово "таблица" и ее порядковый номер (без знака №) пишутся в правом верхнем углу; ниже, по центру должно быть название таблицы.

Первая ссылка на таблицы оформляется следующим образом: «...приведена в табл. 1», - последующие ссылки «см. табл. 1».

Название рисунка включает слово «Рисунок», его номер, а через точку его непосредственное название (например, Рисунок 1. Динамика основных показателей деятельности предприятия, Рисунок 2. Структура рынка).

Если ссылка делается в первый раз, то это оформляется следующим образом: «На рис. 5 приведена структура рынка легковых автомобилей», - в последующих случаях «см. рис. 5»

Нумерация таблиц и рисунков может быть сквозной или соотноситься с номером главы и параграфа. Например, если таблица или рисунок

включены в текст первого параграфа второй главы, нумерация следующая: Таблица 2.1.1., рис. 2.1.1. Последняя цифра означает порядковый номер таблицы (или рисунка) в данном параграфе.

В таблицах можно использовать одинарный интервал, а размер шрифта сократить до 12 – 10. Названия таблиц и рисунков следует выделять жирным шрифтом.

Каждому документу, таблице, схеме, представленным в приложении, присваивается соответствующий номер, обозначаемый арабскими цифрами (например: Приложение 1, Приложение 2 и т.д.).

Таким образом, таблицы в приложениях (если таковые имеются) не нумеруются, и ссылки на них делаются так: "по данным таблицы, представленной в Приложении 4 ..." и т.д.

Графики, диаграммы, схемы и т.д. в тексте работы называются рисунками.

Оформление приложений

Иллюстрации, таблицы, текст вспомогательного характера допускается давать в виде приложений. Приложения оформляют как продолжение курсовой работы (проекта) на последующих ее листах.

Приложения располагают в порядке ссылок на них в тексте курсового проекта.

Каждое приложение следует начинать с новой страницы (листа) с указанием наверху справа слова «Приложение», после которого следует буква кириллицы, например: «Приложение А».

Приложение должно иметь заголовок, который располагается симметрично относительно страницы с прописной буквы отдельной строкой.

В рамках каждого приложения нумерация формул, рисунков, таблиц начинается с номера 1, при этом перед номером указывается номер приложения, а через точку ставится сам номер, например «Таблица А.1», «Рисунок В.1» и т.д.

Все приложения должны быть перечислены в содержании курсовой

работы (проекте) с указанием их буквы и заголовков.

В тексте курсовой работы на все приложения должны быть ссылки, например «... в приложении А».

Оформление списка использованной литературы

В конце работы необходимо привести список использованной литературы.

Список используемой литературы оформляется на отдельном листе обычным шрифтом. В заголовке списка заглавными буквами по центру пишется «СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ». Далее через пустую строку размещается список.

Литература размещается в алфавитном порядке и нумеруется.

Примеры:

1. Дедов В.К. Надежность: Учебное пособие. – М.: Экономика, 2000. – 120 с.
2. Дедов В.К., Тараканова А.С. Надежность: Учебное пособие. – М.: Экономика, 2000. – 120 с.
3. Надежность. Учебное пособие/ В.К. Дедов, А.С. Тараканова, К.Л. Иванов и др.; Под ред. Л.Н. Осотова. – М.: Экономика, 2000. – 120с.
4. Березкин К.К. Современные подходы к формированию ценовой политики на продукцию промышленных предприятий// Вопросы экономики. – 2000. - № 3. – С. 15-28
5. О конкуренции и ограничении монополистической деятельности на товарных рынках: Закон РК от 22.03.2009 № 948-1//Бюллетень нормативных актов. - 2009. - № 2-3.

3. Сдача и защита курсовых работ (проектов)

Курсовые работы (проекты) защищаются и проводятся по графику, утвержденному заместителем директора по учебно -производственной работе.

Курсовые работы (проекты), представленные с нарушением предусмотренных сроков без уважительных причин, к допуску не подлежат. Причины нарушения сроков (болезнь, семейные обстоятельства и др.) должны быть подтверждены соответствующими документами и заявлением студента.

При защите курсовой работы (проекта) необходимо обосновать актуальность темы, сформулировать цель и задачи ее написания, изложить основные выводы и рекомендации, обосновать их эффективность. Студент может воспользоваться заранее подготовленными тезисами доклада, а также своей курсовой работой.

Курсовая работа (проект) оценивается по пятибалльной системе. Оценка “отлично” по МДК 01.02. Прикладное программирование, по которой выполнена курсовая работа (проект) выставляется только при условии успешной сдачи курсовой работы (проекта) на оценку не ниже оценки “хорошо”.

При оценке учитываются качество выполнения работы, ее научно-теоретический уровень и связь с практикой, степень самостоятельности в изложении материала, логика, язык и стиль изложения, оформление работы, проделанная студентом работа по устранению недостатков, выступление студента на защите, ответы на вопросы по теме.

Обучающемуся, получившему неудовлетворительную оценку по курсовой работе (проекту):

- не разрешается допуск к сдаче итогового контроля знаний по МДК 01.02. Прикладное программирование.

Защита курсовой работы (проекта) принимается комиссией из числа преподавателей по решению соответствующей предметной (цикловой)

КОМИССИИ.

Законченные и принятые у обучающихся курсовые работы (проекты) хранятся в архиве техникума сроком один год. По истечению данного срока все курсовые работы (проекты) списываются по акту и уничтожаются.

Лучшие курсовые работы (проекты), представляющие учебно-методическую ценность хранятся в соответствующих предметных (цикловых) комиссиях и используются в качестве учебных пособий.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БАШКОРТОСТАН
НАИМЕНОВАНИЕ УЧЕБНОГО ЗАВЕДЕНИЯ

Специальность: 09.02.03 Программирование в компьютерных системах

Курсовая работа

по МДК 01.02. ПРИКЛАДНОЕ ПРОГРАММИРОВАНИЕ

На тему XXXXXXXXXXXX

Выполнила студентка 423 группы

ФАМИЛИЯ И.О.

Руководитель:

ФАМИЛИЯ И.О.

Работа защищена

«__» _____ 20__

Оценка : _____

Подпись: _____

ЗАДАНИЕ НА РАЗРАБОТКУ КУРСОВОЙ РАБОТЫ (ПРОЕКТА)

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БАШКОРТОСТАН

НАИМЕНОВАНИЕ УЧЕБНОГО ЗАВЕДЕНИЯ

ЗАДАНИЕ

на разработку курсовой работы по МДК 01.02. ПРИКЛАДНОЕ ПРОГРАММИРОВАНИЕ

(ФИО полностью) _____ студенту 323 группы
IV курса Специальность 09.02.03 Программирование в компьютерных системах

Тема курсовой работы: (наименование полностью)

Дата выдачи задания _____

(число, месяц, год) (подпись и фамилия студента)

(подпись и фамилия руководителя)

Дата защиты курсовой работы _____

(число, месяц, год) (подпись и фамилия студента)

(подпись и фамилия руководителя)