

МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»

Кафедра «Автоматизированные системы управления»

ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

*Методические рекомендации к курсовому проектированию
для студентов специальности*

*1 - 53 01 02 «Автоматизированные системы обработки
информации» дневной и заочной форм обучения*



Могилев 2019

УДК 004.4
ББК 32.973
О 75

Рекомендовано к изданию
учебно-методическим отделом
Белорусско-Российского университета

Одобрено кафедрой «Автоматизированные системы управления»
«12» июня 2019 г., протокол № 12

Составитель ст. преподаватель А. И. Кашпар

Рецензент Ю. С. Романович

Методические рекомендации к курсовому проектированию предназначены для студентов специальности 1-53 01 02 «Автоматизированные системы обработки информации» дневной и заочной форм обучения.

Учебно-методическое издание

ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

Ответственный за выпуск	А. И. Якимов
Редактор	А. Т. Червинская
Компьютерная верстка	Е. В. Ковалевская

Подписано в печать . Формат 60×84/16. Бумага офсетная. Гарнитура Таймс.
Печать трафаретная. Усл. печ. л. . Уч.-изд. л. . Тираж 31 экз. Заказ №

Издатель и полиграфическое исполнение:
Межгосударственное образовательное учреждение высшего образования
«Белорусско-Российский университет».

Свидетельство о государственной регистрации издателя,
изготовителя, распространителя печатных изданий
№ 1/156 от 07.03.2019.

Пр-т Мира, 43, 212022, Могилёв.

© Белорусско-Российский
университет, 2019



Содержание

Введение.....	4
1 Общая формулировка задания и требования к программе.....	5
2 Содержание пояснительной записки	6
2.1 Перечень основных разделов	6
2.2 Титульный лист.....	6
2.3 Задание на курсовую работу.....	6
2.4 Содержание.....	7
2.5 Введение.....	7
2.6 Техническое задание.....	7
2.7 Описание программы.....	7
2.8 Руководство оператора.....	11
2.9 Тестирование.....	14
2.10 Заключение.....	14
2.11 Список использованных источников.....	14
2.12 Приложения.....	14
3 Оформление пояснительной записки	15
3.1 Общие требования.....	15
3.2 Нумерация страниц.....	16
3.3 Построение текста.....	16
3.4 Рисунки.....	16
3.5 Таблицы.....	17
3.6 Формулы.....	18
3.7 Ссылки.....	18
3.8 Приложения	18
3.9 Список использованных источников.....	19
Список литературы.....	20
Приложение А. Образец оформления титульного листа.....	21
Приложение Б. Образец оформления листа задания.....	22
Приложение В. ГОСТ 19.402–2000 Описание программы.....	23
Приложение Г. ГОСТ 19.505–79 Руководство оператора.....	29
Приложение Д. Вариативная часть	30

Введение

Целью курсовой работы является закрепление основ и углубление знаний приемов программирования на языке C#, получение практических навыков на всех этапах создания программного продукта: от постановки задачи до практической реализации, сопровождающейся документацией и инструкциями по его использованию. При выполнении курсовой работы студентам рекомендуется обратить внимание на качество создаваемых программ и оформления документации.

На проверку преподавателю предоставляются оформленная и скрепленная пояснительная записка и программа на любом электронном носителе (исходные и исполняемый файлы).

Также можно предоставить для проверки материалы курсовой работы в электронном виде. Проверка курсовых работ в электронном виде прекращается за две недели до начала экзаменационной сессии.



1 Общая формулировка задания и требования к программе

В курсовой работе требуется разработать приложение Windows Forms в среде программирования Microsoft Visual Studio.

Язык программирования – C#.

Структура проекта – многофайловая.

Программа должна иметь заставку с указанием темы проекта, фамилии исполнителя и управляться при помощи системы меню.

Парадигма программирования – процедурная (по согласованию с преподавателем допускается реализация программы в рамках объектно-ориентированной парадигмы программирования).

Способ организации данных – структуры (struct) (либо поля соответствующих классов).

Способ хранения данных – файлы (по согласованию с преподавателем допускается подключение баз данных).

Каждая логически завершенная задача программы должна быть реализована в виде метода.

Разрабатываемый интерфейс должен быть понятным и защищенным от случайных ошибок.

Разрабатываемая программа должна содержать не менее двух форм и иметь раздел «Помощь».

Ввод исходных данных должен осуществляться либо с клавиатуры, либо из файла (по выбору пользователя).

Результаты работы программы должны сохраняться в текстовый файл по запросу пользователя.

Функциональные требования к программе.

Первым этапом работы программы является авторизация – предоставление прав доступа. В рамках этого этапа необходимо считать данные из файла с учетными записями пользователей следующего вида:

- login;
- password;
- role (данное поле служит для разделения в правах администраторов и пользователей).

После ввода пользователем своих персональных данных (логина и пароля) и сверки со считанной из файла информацией необходимо предусмотреть возможность входа в качестве администратора (в этом случае, например, role = 1) или в качестве пользователя (в этом случае, например, role = 0).

Если файл с учетными записями пользователей не существует, то необходимо программно создать его и записать учетные данные администратора. Регистрацию новых пользователей при входе в систему выполняет администратор в режиме работы с учетными записями пользователей.



Вторым этапом работы программы является собственно работа с данными, которая становится доступной только после прохождения авторизации. Данные хранятся в отдельном файле и имеют вид, описанный подробно в каждом варианте к курсовой работе.

Для работы с данными должны быть предусмотрены два функциональных модуля: модуль администратора и модуль пользователя.

Модуль администратора включает следующие подмодули (с указанием функциональных возможностей).

- 1 Управление учетными записями пользователей:
 - просмотр всех учетных записей;
 - добавление новой учетной записи;
 - редактирование учетной записи;
 - удаление учетной записи.
- 2 Работа с файлом данных:
 - создание файла;
 - открытие существующего файла;
 - удаление файла.
- 3 Работа с данными:
 - режим редактирования:
 - а) просмотр всех данных;
 - б) добавление новой записи;
 - в) удаление записи;
 - г) редактирование записи;
 - режим обработки данных:
 - а) выполнение индивидуального задания;
 - б) поиск данных (как минимум по трем различным параметрам);
 - в) сортировка (как минимум по трем различным параметрам).

Модуль пользователя включает подмодуль работы с данными со следующими функциональными возможностями:

- просмотр всех данных;
- выполнение индивидуального задания;
- поиск данных (как минимум по трем различным параметрам);
- сортировка (как минимум по трем различным параметрам).

Для реализации перечисленных модулей/подмодулей необходимо создавать меню с соответствующими пунктами.

Предусмотреть:

- обработку исключительных ситуаций:
 - а) введенные пользователем данные не соответствуют формату поля (*например, символы в числовом поле*);
 - б) введенные пользователем данные некорректны (*например, отрицательная цена товара*);
 - в) файл с данными для чтения не существует;
 - г) ничего не найдено по результатам поиска;
 - д) номер удаляемой записи выходит за пределы массива/вектора;



- е) логин новой учетной записи уже существует;
 - возможность возврата назад (навигация);
 - запрос на выполнение необратимых действий, а именно подтверждение удаления вида «Вы действительно хотите удалить файл (запись)?»;
 - обратную связь с пользователем, например, вывод сообщения об успешности создания файла/удаления записи/ и т. д.

Требования к программной реализации курсового проекта.

1 Все переменные и константы должны иметь осмысленные имена в рамках тематики варианта к курсовой работе.

2 Код не должен содержать неименованных числовых констант, неименованных строковых констант (например, имен файлов и др.). Подобного рода информацию следует выносить в глобальные переменные с атрибутом const. По правилам хорошего стиля программирования тексты всех информационных сообщений, выводимых пользователю в ответ на его действия, также оформляются как константы.

3 Код не должен дублироваться – для этого существуют методы.

4 Один метод решает только одну задачу (например, не допускается в одном методе считывать данные из файла и выводить их на консоль – это два разных метода). При этом внутри метода возможен вызов других методов.

5 Выполнение операций чтения/записи в файл должно быть сведено к минимуму (т. е. после однократной выгрузки данных из файла в массив/вектор дальнейшая работа ведется с этим массивом/вектором, а не происходит многократное считывание данных из файла в каждом методе).

6 Следует избегать длинных методов и глубокой вложенности: текст функции должен уместиться на один экран, а вложенность блоков и операторов должна быть не более трёх.

7 Выносите код логически независимых модулей в отдельные .cs файлы.

2 Содержание пояснительной записки

2.1 Перечень основных разделов

Пояснительная записка должна включать:

- титульный лист;
- задание на курсовую работу;
- содержание;
- введение;
- основную часть;
- заключение;
- список использованных источников;
- приложения.

Основная часть пояснительной записки должна иметь следующую структуру и не может меняться.



- 1 Техническое задание.
- 2 Описание программы.
- 3 Руководство оператора.
- 4 Тестирование.

2.2 Титульный лист

Титульный лист – это первая страница записки (но номер страницы на титульном листе не ставится). Пример его оформления приведен в приложении А. Перенос слов на титульном листе не допускается. Точки в конце заголовков не ставятся.

2.3 Задание на курсовую работу

Задание помещается после титульного листа, не нумеруется. Пример его оформления приведен в приложении Б.

Студенту необходимо в задании указать следующие сведения:

- срок сдачи законченной работы;
- фамилию, имя, отчество студента;
- группу;
- исходные данные;
- фамилию, имя, отчество руководителя курсовой работы;
- поставить подпись и указать дату получения задания.

2.4 Содержание

Содержание помещается в записке после задания. Оно состоит из последовательно перечисленных наименований структурных единиц записки (см. подразд. 2.1), начиная с введения, с указанием соответствующего номера страницы.

Слово «Содержание» записывается в виде заголовка симметрично тексту с прописной буквы. Наименования, включенные в содержание, записываются строчными буквами, первая буква – прописная (с абзаца).

2.5 Введение

Введение – это небольшой обзор по теме решаемой задачи. В нем нужно кратко отметить то, что необходимо сделать, и то, что должно быть получено в итоге. В общем случае во введении следует:

- охарактеризовать проблему, к которой относится тема работы;
- указать цель выполнения работы;
- изложить задачи, которые необходимо решить в процессе выполнения работы;
- изложить ожидаемые результаты;
- кратко описать содержание пояснительной записки.



2.6 Техническое задание

В данном разделе требуется привести развернутое описание требований к программе, разработать структуру меню.

Требования к программе включают:

- полный текст варианта задания;
- исходные данные для курсовой работы из первого раздела (функциональные требования к курсовому проекту (основной или альтернативный вариант));
- функциональные требования к конкретному курсовому проекту (рекомендуется взять за основу материал из первого раздела и расширить его для своей темы, например, прописать индивидуальное задание, разновидности поиска и сортировки, конкретизировать возможные исключительные ситуации);
- требования к программной реализации. В требовании не стоит указывать, что не реализуется в действительности. Для надежности после завершения работы над программой необходимо проверить соответствие программы тому, что описано в требованиях.

Разработка структуры меню подразумевает его графическое представление (пример на рисунке 2.1).



Рисунок 2.1 – Структура меню

Графическая среда может быть любой; в качестве рекомендации можно использовать online-редакторы, например:

http://www.newart.ru/htm/flash/risovalka_42.php

2.7 Описание программы

Данный раздел выполняется согласно ГОСТ 19.402 – 2000 Описание программы (приложение В).

Общие сведения – полное наименование программы, обозначение, ее возможные применения, а также программное обеспечение, необходимое для функционирования программы, и языки программирования, на которых она написана. Например:

Программа «Автоматизированное рабочее место разработчика САУ» предназначена для... реализована на... Программа поддерживает...

Программа написана на языке... с использованием компилятора...

Функциональное назначение – назначение программы и общее описание ее функционирования, основные характеристики, сведения об ограничениях, накладываемых на область применения программы. Например:

Программа предназначена для решения задач... Программа представляет собой ядро автоматизированного рабочего места...

Пользователь имеет возможность..., осуществить..., запустить..., проанализировать..., получить результаты анализа и обработки..., построить... и т.п.

Описание логической структуры – используемые методы, алгоритмы программы, описание ее структуры и логики (с привязкой к тексту программы на исходном языке) и составных частей, их методов и связей между ними, а также связи программы с другими программами. Содержание этого раздела должно быть конкретным и опираться на текст программы.

Выбор метода решения предполагает определение теоретической возможности решения задачи и нахождение формального правила его получения. Данный этап плохо формализуется, что связано с чрезвычайно широким многообразием задач и методов их решения.

Метод решения может быть представлен:

- в виде системы формул (безусловной или условной);
- в виде словесного изложения последовательности действий;
- в виде их комбинаций.

Описание метода может содержать ограничения на исходные данные, накладываемые методом.

Схемы алгоритмов, программ, данных и систем, приводимые в этом разделе, должны использовать метод пошаговой детализации. Причем уровень детализации должен быть таким, чтобы различные части программы и



взаимосвязи между ними были понятны в целом. Другими словами, сначала нужно описать обобщенную схему алгоритма, разбив программу на логические блоки, а затем подробно рассмотреть схемы алгоритмов работы каждого из этих блоков. Условные обозначения и правила выполнения схем определяются ГОСТ 19.701–90.

При использовании событийного программирования необходимо разработать и описать граф состояний интерфейса, на основе которого затем проектируются обработчики задействованных событий.

Разработка перечня пользовательских функций программы подразумевает перечисление и краткие комментарии прототипов функций, необходимых для реализации программы. Прототипы функций рекомендуется разбить на тематические группы в соответствии с модульной структурой программы. *В случае использования объектно-ориентированного программирования приводятся методы для классов.* Код функций (методов) не приводится, т. к. на этом этапе он еще не существует.

Пример

Структуры разработанных диалоговых форм представлены на рисунке 2.2.

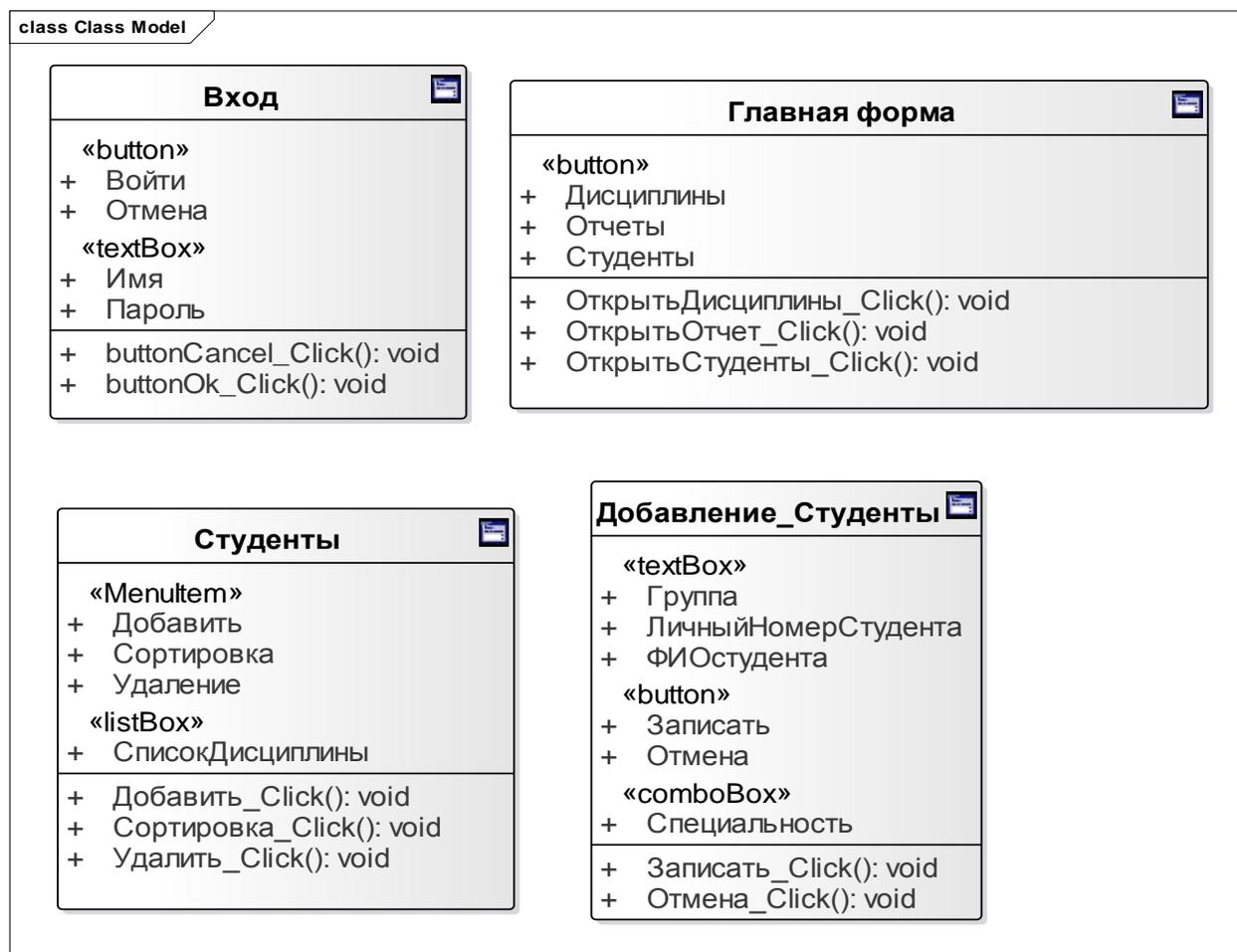


Рисунок 2.2 – Диаграмма классов диалоговых форм

Далее необходимо описать все методы программного модуля в виде таблицы 2.1.

Таблица 2.1 – Описание методов

Имя формы	Имя элемента управления	Метод (обработчик событий)	Краткое описание
Вход	buttonOk	buttonOk_Click()	Метод, вызываемый при нажатии кнопки «Войти»
Студенты		SortF()	Метод для сортировки массива структур по фамилии
...			

Используемые технические средства – типы ЭВМ и устройств, применяемых при работе программы. Например:

Программа эксплуатируется на персональном компьютере (ПК) типа IBM PC/AT. Для работы в диалоговом режиме используется экран дисплея, клавиатура и манипулятор типа «мышь». Для поддержки графического режима необходим адаптер SVGA. Входные данные хранятся на гибком и/или жестком дисках. Программа работает под управлением ОС...

Вызов и загрузка – способ вызова программы с соответствующего носителя данных и входные точки в программу, а также имя загрузочного модуля, сведения об использовании оперативной памяти, объем программы и описание всей процедуры вызова и загрузки системы. Например:

Загрузка программы осуществляется набором в командной строке DOS имени загрузочного модуля – SBM80N.EXE с возможным указанием имени файла данных.

Входные данные – характер, организация и предварительная подготовка входных данных, их формат, описание и способ кодирования.

Описание структуры входных данных включает указание типов данных, диапазонов их значений, форматов представления и т. д.

В этом же разделе должно приводиться перечисление аномалий во входных данных (значения, для которых нельзя применять реализуемый метод решения) и реакций программы на них.

Рекомендуется оформлять данный раздел в виде таблицы 2.2.

Пример

Входными данными для программы является текстовый файл, описывающий расширенную матрицу инцидентий графа исследуемой системы. Этот файл имеет следующую структуру...

Используемый метод определения... не может быть использован при..., поэтому эти значения являются недопустимыми, а программа реагирует на них выводом сообщения...

Таблица 2.2 - Входные данные

Название	Обозначение	Тип данных	Диапазон значений
Коэффициент трения	mu	double	0...∞
Угол	alfa	double	Ограничений нет
Время начальное	tn	double	0...∞

Выходные данные – характер и организация выходных данных, их формат, описание и способ кодирования.

Описание структуры выходных данных включает указание типов данных, диапазонов их значений, форматов представления и т. д.

Пример

Выходными данными являются:

- выводимая на экран графическая и текстовая информация (результаты анализа системы);*
- файлы в одном из графических форматов – копии изображения построенных характеристик (АЧХ, ФЧХ и т.д.);*
- текстовые файлы – отчеты о проведенных исследованиях в формате...;*
- диагностика состояния системы и сообщения обо всех возникших ошибках.*

В зависимости от особенностей программы допускается введение дополнительных разделов.

Содержание разделов допускается иллюстрировать пояснительными примерами, таблицами, схемами, графиками, примерами и т. п.

2.8 Руководство оператора

Данный раздел выполняется согласно ГОСТ 19.505–79 *Руководство оператора* (приложение Г).

В разделе «Назначение программы» должны быть указаны сведения о назначении программы и информация, достаточная для понимания функций программы и ее эксплуатации. Например,

Программа обеспечивает возможность Программа должна эксплуатироваться Пользователями программы должны являться

Пользователи программы должны обладать навыками

Программа обеспечивает возможность выполнения следующих функций:

- ...;*
- ... и т.д.*



В разделе «Условия выполнения программы» должны быть указаны условия, необходимые для выполнения программы (минимальный и (или) максимальный состав аппаратурных и программных средств и т. п.).

Например,

Климатические условия эксплуатации, при которых должны обеспечиваться заданные характеристики, должны удовлетворять следующим требованиям: температура от ... до..., влажность воздуха, и т. д.

В состав технических средств должен входить IBM-совместимый персональный компьютер (ПЭВМ) или Mac, включающий в себя:

- процессор не хуже...;
- объем оперативной памяти не менее...;
- жесткий диск объемом не менее...;
- звуковая карта не хуже;
- сетевая карта такая-то;
- и так далее.

Состав и параметры периферийных устройств:

- сканер такой-то;
- принтер такой-то;
- и т. д.

Принтер такой-то необходим из-за того, что ... (например, другие принтеры не обладают такими-то характеристиками).

Сканер такой-то необходим из-за того, что ... (например, у других сканеров маловато разрешение или узковата область сканирования).

Системные программные средства, используемые программой, должны быть представлены ... (например, лицензионной локализованной версией операционной системы). Допускается использование пакета обновления такого-то.

Также необходимо наличие таких-то драйверов (например, принтера, сканера). Чтобы программа полноценно выполняла свои функции, требуется наличие такого-то прикладного программного обеспечения.

В разделе «Выполнение программы» должна быть указана последовательность действий оператора, обеспечивающих загрузку, запуск, выполнение и завершение программы, приведено описание всех функций, формата и возможных вариантов команд, с помощью которых оператор осуществляет загрузки и управляет выполнением программы, а также ответы программы на эти команды.

В настоящем документе исключено прямое обращение к пользователю. Должны отсутствовать слова «откройте», «нажмите», «укажите» и пр. Следует применять штампы «следует открыть», «следует нажать» и им подобные.



Пример

Загрузка и запуск программы осуществляется ... способами. В случае успешного запуска программы на рабочем столе будет отображено Главное окно программы (рисунок 2.3).

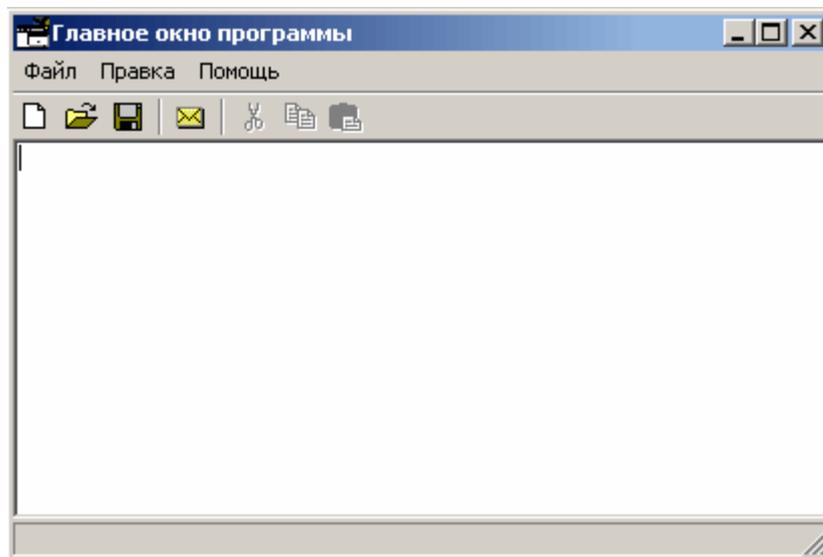


Рисунок 2.3 – Вид формы «Главное окно программы»

Выполнение функции создания нового (безымянного) файла возможно любым из перечисленных ниже способов:

- последовательным выбором пунктов меню *Файл – Создать*;
- нажатием кнопки .

В случае успешного выполнения указанной функции на рабочем столе будет отображено окно, представленное на рисунке Программа готова к вводу и редактированию текста.

...

Завершение работы программы возможно любым из перечисленных ниже способов:

- последовательным выбором пунктов меню *Файл – Выход* (см. рисунок такой-то);
- нажатием кнопки .

В разделе «Сообщения оператору» должны быть приведены тексты сообщений, выдаваемых в ходе выполнения программы, описание их содержания и соответствующие действия оператора (действия оператора в случае сбоя, возможности повторного запуска программы и т. п.).

Поскольку программа не консольная (с интерфейсом командной строки), а с графическим пользовательским интерфейсом, классических текстовых сообщений нет. Сообщения об ошибках отображаются в виде окон на рабочем столе.

Пример

При попытке сохранения файла с именем уже существующего файла на рабочем столе программы будет отображено сообщение об ошибке, представленное на рисунке 2.4.

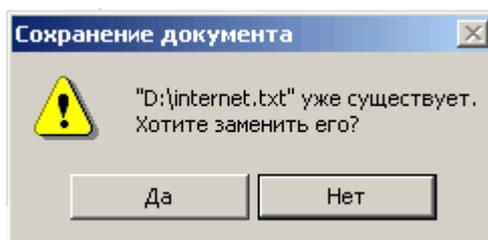


Рисунок 2.4 – Вид сообщения об ошибке

Для сохранения файла с именем уже существующего файла следует нажать кнопку Да.

Для сохранения файла с именем, отличным от имени существующего файла, следует:

2.9 Тестирование

Для тестирования программного продукта нужно самостоятельно подготовить данные, подобрав конкретные значения таким образом, чтобы можно было убедиться в правильности работы программы. Тесты должны быть подготовлены для каждой возможной ситуации (стандартной, при отсутствии решения и т. д.).

Например:

Для тестирования работы программы были подготовлены тесты для каждой ситуации и каждой возможности, для каждой границы областей допустимых значений всех входных данных, областей изменения данных, для всех недопустимых условий.

В результате реализации тестирования было зафиксировано, что все условные переходы выполняются в каждом направлении, не происходит “заикливания” в модуле при граничных значениях индексов циклов, также как и не обнаружено сбоев в работе модуля при невыполнении тела какого-либо из циклов.

Данные по результатам тестирования занесены в «Журнал тестирования программного продукта» (таблица 2.3).

Таблица 2.3 – Журнал тестирования программного продукта

Объект	Команда пользователя	Ожидаемая реакция	Отметка об отклике системы
О программе	Нажатие	Появление окна с информацией о программе	Работает корректно
Кнопка «Добавление»	Нажатие	Открытие формы «Добавление нового студента»	Данные добавлены в список

2.10 Заключение

В заключении приводятся выводы по результатам выполненной работы (какие вопросы были рассмотрены, что было изучено, что было разработано), указываются ее достоинства и недостатки, даются предложения по использованию, указываются перспективы дальнейшего развития.

2.11 Список использованных источников

В списке перечисляются книги, статьи, источники из Интернета, которые были использованы при выполнении работы. Информацию о правилах оформления этого списка смотрите в п. 3.9, а также в ГОСТ 7.1–2003 Библиографическая запись. Библиографическое описание. Общие требования и правила.

Список использованных источников должен содержать не менее пяти наименований.

2.12 Приложения

Приложения содержат материалы вспомогательного характера: алгоритмы, тексты программ, результаты тестирования, большие таблицы и т. д. В записке по данной курсовой работе обязательным является наличие приложения с текстом программы.

3 Оформление пояснительной записки

3.1 Общие требования

Пояснительная записка к курсовой работе оформляется в соответствии с правилами оформления текстовых документов, изложенными в ГОСТ 2.105–95 *Общие требования к текстовым документам*, и с правилами оформления курсовых работ, изложенными в данном методическом указании. Список использованных источников оформляется в соответствии с правилами, изложенными в ГОСТ 7.1–2003.



Записка должна быть напечатана на одной стороне листа белой бумаги формата А4. Рекомендуется: шрифт – Times New Roman, размер шрифта – 14, межстрочный интервал – одинарный.

Отступ от рамки – не менее 5 мм.

Цвет шрифта – черный, при необходимости вписывать также надо чернилами, пастой или тушью черного цвета. Опечатки не зачеркивать, а заклеивать, подчищать или закрашивать белой краской.

В тексте после знаков препинания обязательно ставится пробел. Нельзя сокращать слова (кроме сокращений, установленных правилами орфографии). Например, пишется целиком «то есть», «так как». Нельзя употреблять специальные знаки типа «=», «%», «+» и др. без цифр.

Записка оформляется на листах с рамкой.

Раздел «Содержание» печатается на листе с основной подписью для заглавного листа (пример заполнения представлен на рисунке 3.1), остальные листы курсовой работы выполняются на листах с основной подписью для последующих листов (пример заполнения представлен на рисунке 3.2).

					КР.1-53 01 02. № зачетной книжки. № по списку. 81-01			
Изм.	Лист	№ докум.	Подп.	Дата				
Разраб.	ФИО студента				Разработка программного модуля Курсовая работа	Лит.	Лист	Листов
Пров.	ФИО преподав.						2	21
Н.контр.						БРУ гр. № группы		
Утв.								

Рисунок 3.1 – Основная надпись для заглавного листа

					КР.1-53 01 02. № зачетной книжки. № по списку. 81-01		Лист
Изм.	Лист	№ Докум.	Подп.	Дата			4

Рисунок 3.2 – Основная надпись для последующих листов

3.2 Нумерация страниц

Нумерация страниц – сквозная, по всем страницам записки, включая приложения. Первая страница – это титульный лист, но на нем номер не ставится. Номер проставляется арабской цифрой без точки в нижней части листа справа.

3.3 Построение текста

Рассмотрим разбивку и нумерацию структурных единиц записки. Структурные единицы пояснительной записки приведены в подразд. 2.1. Каждая структурная единица записки начинается с нового листа. В основной части, кроме того, каждый раздел начинается с нового листа.

Структурные единицы записки «Содержание», «Введение», «Заключение», «Список использованных источников» не нумеруются. Соответствующие заголовки записываются обычным шрифтом с прописной буквы симметрично по тексту. Заголовки не подчеркиваются и точка в конце них не ставится.

Текст основной части делят на разделы (номер раздела состоит из одной цифры), разделы – на подразделы (номер подраздела состоит из двух цифр), подразделы – на пункты и т. д.

Разделы нумеруются арабскими цифрами без точки; подразделы – в пределах раздела. Номер состоит из номера раздела и подраздела, разделенных точкой. В конце номера подраздела точка не ставится. За номером раздела или подраздела следует его название, оно записывается обычным шрифтом с прописной буквы без точки в конце. Переносы слов в заголовках не допускаются. Если заголовок состоит из двух предложений, их разделяют точкой.

Каждый раздел начинается с нового листа, подразделы с нового листа не начинаются. Номер и название раздела или подраздела записываются с абзацного отступа. Расстояние между заголовком и текстом, заголовками раздела и подраздела – одна пустая строка.

3.4 Рисунки

Все иллюстрации (чертежи, схемы, графики, структурные схемы и др. (кроме таблиц)) называются рисунками. Они должны располагаться непосредственно после ссылки на них в тексте или на следующей странице. Рисунок располагается так, чтобы его удобно было смотреть без поворота листа или с поворотом по часовой стрелке.

Иллюстрации, за исключением иллюстраций приложений, следует нумеровать арабскими цифрами сквозной нумерацией. Если рисунок один, то он обозначается, например, «Рисунок 1». Слово «Рисунок» и его номер располагаются с абзацного отступа.

Допускается нумеровать иллюстрации в пределах разделов. В этом случае номер иллюстрации состоит из номера раздела и порядкового номера иллюстрации, разделенных точкой. Например, надпись «Рисунок 1.2» означает второй рисунок первого раздела.

Иллюстрации каждого приложения обозначают отдельной нумерацией арабскими цифрами, добавляя перед цифрой обозначение приложения. Например, «Рисунок А.3», что обозначает третий рисунок в приложении А.

Иллюстрации, при необходимости, могут иметь наименование и пояснительные данные (подрисуночный текст). Порядок расположения следующий: вначале – сам рисунок, затем – подрисуночный текст, далее – слово «Рисунок», номер рисунка и наименование в виде, например, «Рисунок 1 – Структурная схема».



3.5 Таблицы

В тексте пояснительной записки следует помещать итоговые и наиболее важные таблицы; таблицы вспомогательного и справочного характера – в приложениях.

Таблицы нумеруются и обозначаются по тем же правилам, что и иллюстрации (см. п. 3.4), только вместо слова «Рисунок» пишется слово «Таблица» и располагается это слово и название таблицы без абзацного отступа над таблицей. Примеры обозначений: «Таблица 1», «Таблица В.1», «Таблица 3.1 – Список модулей».

Таблицы ограничиваются одинарными горизонтальными и вертикальными линиями. Использование диагональных линий не допускается.

Если таблица выходит за формат листа, то ее делят на части. Части таблицы можно помещать на одном листе одна над другой или рядом или переносить на другие листы. При переносе части таблицы на другой лист заголовков помещают только над первой частью, то есть слово «Таблица» с названием указывают только над первой частью, а над другими частями пишут слова «Продолжение таблицы» с указанием номера таблицы. Головку таблицы при переносе части таблицы повторяют.

Заголовки строк (горизонтали) и граф (вертикали) пишут с прописной буквы, а подзаголовки граф – со строчной, если они составляют одно предложение с заголовком, или с прописной буквы, если они имеют самостоятельное значение.

3.6 Формулы

Формулы располагаются в тексте в отдельных строках, по центру строки. Нумеруются они по тем же правилам, что и иллюстрации (см. п. 3.4). Номера формул записываются на уровне формулы в круглых скобках справа, в конце строки. Примеры обозначения номеров формул: при сквозной нумерации – (1), при нумерации в пределах раздела – (3.1), при нумерации в приложении – (В.1).

Пояснения символов и числовых коэффициентов, входящих в формулу, если они не пояснены ранее в тексте, должны быть приведены непосредственно под формулой. Пояснение каждого символа следует давать с новой строки в той последовательности, в которой символы приведены в формуле. Первая строка пояснения должна начинаться со слова «где» без двоеточия после него.

3.7 Ссылки

Примеры ссылок в тексте пояснительной записки:

- 1) на иллюстрацию – «... в соответствии с рисунком 1.2 ...»;
- 2) на формулу - «... в формуле (2.1) ...»;
- 3) на приложение - «... (приложение Б) ...»;
- 4) на таблицу - «... в таблице 1.2 ...».



Повторные ссылки следует давать с сокращенным словом «смотри», например: (см. рисунок 1.2), (см. приложение Б).

На материалы, взятые из литературы или других источников, должны быть даны ссылки с указанием номера источника по списку использованных источников (см. п. 3.9). Номер ссылки проставляется арабскими цифрами в квадратных скобках, например: [1], [1,2,5].

3.8 Приложения

Каждое приложение начинается с нового листа с указанием наверху посередине слова «Приложение» с первой прописной буквы и его обозначения. Приложения обозначаются заглавными буквами русского алфавита, начиная с А, за исключением букв Ё, З, Й, О, Ч, Ъ, Ы, Ь. Например, «приложение А», «приложение Б» и т. д.

После строки, содержащей слово «Приложение», то есть на второй строке, записывается симметрично по тексту с первой прописной буквы заголовок приложения. Например:

Приложение А
(обязательное)
Текст программы

Затем через одну пустую строчку следует текст приложения. Текст каждого приложения может быть разделен на разделы, подразделы и т. д., которые нумеруются арабскими цифрами в пределах приложения по аналогии с разделами и подразделами основной части пояснительной записки. Например, «П.А.2» – это обозначение второго раздела приложения А.

3.9 Список использованных источников

Источники – это книги, учебники, диссертации, статьи из журналов, статьи из Интернета и т. д., использованные при выполнении курсовой работы. Источники в списке располагаются в порядке ссылок в тексте записки или по алфавиту, нумеруются арабскими цифрами без точки и печатаются с абзацного отступа, при этом дается библиографическое описание каждого источника в соответствии с ГОСТ 7.1, ГОСТ 7.12.

Общий шаблон описания книги, у которой не более трех авторов: ФИО_автора, название книги, точка, тире, город, двоеточие, издательство, запятая, год издания, точка, тире, количество страниц, буква «с», точка.

Название города дается целиком, допустимы только сокращения «М.» (Москва) и «СПб.» (Санкт-Петербург); название издательства – без кавычек. Если у книги один, два или три автора, то вначале указывается фамилия, потом – инициалы.



Например,

Один, два или три автора:

Шуп, Т. Решение инженерных задач на ЭВМ: практическое руководство: пер. с англ. / Т. Шуп. – М.: Мир, 1982. – 238 с.

Четыре и более автора:

Зубчатые передачи: справочник / Е. Г. Гинзбург [и др.]; под ред. Н. Т. Халебского. – Л.: Машиностроение, 1980. – 416 с.

Книги под общей редакцией:

Шейнин, А. М. Эксплуатация дорожных машин / А. М. Шейнин; под общ. ред. А. М. Шейнина. – М.: Транспорт, 1992. – 328 с.

Стандарты:

ГОСТ 19.701 – 90. Схемы алгоритмов, программ данных и систем. – М.: Изд-во стандартов, 2004. – 26 с.



Список литературы

- 1 **Биллинг, В. А.** Основы программирования на С# / В. А. Биллинг. – Москва: Бином, 2012. – 488 с.
- 2 **Васильев, А. Н.** С#. Объектно-ориентированное программирование. Учебный курс / А. Н. Васильев. – Санкт-Петербург: Питер, 2012. – 320 с.: ил.
- 3 **Дейтел, Х.** Как программировать на Visual С# 2012 / Х. Дейтел. – Санкт-Петербург: Питер. 2014. – 864 с.
- 4 **Мартынов, Н.** С# для начинающих / Н. Мартынов. – Москва: КУДИЦ-ПРЕСС, 2007. – 272 с.
- 5 **Павловская, Т. А.** С#. Программирование на языке высокого уровня: учебник для вузов / Т. А. Павловская. – Санкт-Петербург: Питер, 2007. – 432 с.: ил.
- 6 **Подбельский, В. В.** Язык С#. Решение задач / В. В. Подбельский. – Москва: Инфра-М, 2014. – 296 с.
- 7 **Троелсен, Э.** Язык программирования С# 4.0 и платформа .NET 4.0: пер. с англ. / Э. Троелсен. – Москва: Вильямс, 2011. – 1392 с.
- 8 **Фленов, М.** Библия С# / М. Фленов. – Санкт-Петербург: БХВ-Петербург, 2013. — 560 с.
- 9 **Шилдт, Г.** С#. Учебный курс / Г. Шилдт. – Санкт-Петербург: Питер, 2003. – 512 с.: ил.
- 10 **Шилдт, Г.** С# 4.0. Полное руководство : пер. с англ. / Г. Шилдт. – Москва: Вильямс, 2011. – 1056 с.
- 11 **Visual С# 2010.** Полный курс / К. Уотсон [и др.]. – Москва: Вильямс, 2011. – 955 с.



Приложение А (обязательное)

Образец оформления титульного листа

МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»

Кафедра «Автоматизированные системы управления»

РАЗРАБОТКА ПРОГРАММЫ

Курсовая работа
по дисциплине «Основы алгоритмизации и программирования»

КР.1-53 01 02. №зачетной книжки. № по списку. ПЗ



Исполнитель _____ ФИО студента, группа
(подпись)

Руководитель _____ ФИО руководителя
(подпись)

Дата допуска к защите _____

Дата защиты _____

Оценка _____

Приложение Б (обязательное)

Образец оформления листа задания

МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»

Факультет Электротехнический

«Утверждаю»

Заведующий кафедрой АСУ _____
«__» _____ 2019 г.

З А Д А Н И Е

на курсовую работу по дисциплине «Основы алгоритмизации и
программирования»

студенту

ФИО студента

гр. № группы

1. Тема курсовой работы: Разработка программы...
2. Срок сдачи законченной работы: указать срок сдачи согласно учебному плану
3. Исходные данные к курсовой работе:

_____ текст задания по варианту _____

4. Перечень подлежащих разработке вопросов:

Введение

1 Техническое задание

2 Описание программы

3 Руководство оператора

4 Тестирование

Заключение

Список использованных источников

5. На проверку предоставляются пояснительная записка, исходные тексты программ и исполняемые файлы на электронном носителе.

Руководитель курсовой работы: _____ /ФИО руководителя /

Задание принял к исполнению _____ / ФИО студента /
подпись студента, дата



Приложение В (рекомендуемое)

ГОСТ 19.402–2000 *Описание программы*

1 Общие положения

1.1 Структуру и оформление документа устанавливают в соответствии с ГОСТ 19.105. Составление информационной части (аннотации и содержания) является обязательным.

1.2 Описание программы содержит следующие разделы:

- общие сведения;
- функциональное назначение;
- описание логической структуры;
- используемые технические средства;
- вызов и загрузка;
- входные данные;
- выходные данные.

В зависимости от особенностей программы допускается вводить дополнительные разделы.

Разделы документа располагают в порядке их описания в данном стандарте. Если информация для отражения в каком-нибудь разделе отсутствует, то вслед за заголовком этого раздела приводят справку: "Соответствующая информация для заполнения данного раздела отсутствует, по следующим причинам исключена".

2 Содержание разделов

2.1 В разделе «Общие сведения» указывают:

- обозначение и наименование программы;
- автора, дату создания, версию программы;
- программную среду и программное обеспечение, необходимые для функционирования программы, а также рекомендации по выбору версий и модификаций;
- языки программирования, на которых написана программа;
- новые возможности и отличие от предыдущих версий и модификаций.

2.2 В разделе «Функциональное назначение» указывают классы решаемых задач и (или) назначение программы и сведения о функциональных ограничениях на применение.

2.3 В разделе «Описание логической структуры» указывают:

- алгоритм программы;
- используемые методы и типовые проектные решения;



- структуру программы с описанием функций составных частей, их проблемной и машинной ориентации и связей между ними;
- связи программы с другими программами;
- используемые стандартные протоколы связи программы.

2.4 В разделе «Используемые технические средства» указывают типы устройств, которые используются при работе программы.

2.5 В разделе «Вызов и загрузка» указывают:

- способ вызова программы с соответствующего носителя данных;
- входные точки в программу.

Допускается указывать адреса загрузки, сведения об использовании оперативной и внешней памяти, объем программы.

2.6 В разделе «Входные данные» указывают:

- характер и организацию входных данных;
- описание предварительной подготовки входных данных;
- формат, описание и способ кодирования входных данных.

2.7 В разделе «Выходные данные» указывают:

- характер и организацию выходных данных;
- формат, описание и способ кодирования выходных данных;
- возможности управления форматом и полнотой получаемых выходных данных.

2.8 Допускается содержание разделов иллюстрировать пояснительными примерами, таблицами, схемами, графиками.

В приложении к описанию программы допускается приводить различные материалы, которые нецелесообразно включать в разделы описания.



Приложение Г (рекомендуемое)

ГОСТ 19.505–79 *Руководство оператора. Требования к содержанию и оформлению*

1 Общие положения

1.1 Структуру и оформление документа устанавливают в соответствии с ГОСТ 19.105–78.

Составление информационной части (аннотации и содержания) является обязательным.

1.2 Руководство оператора должно содержать следующие разделы:

- назначение программы;
- условия выполнения программы;
- выполнение программы;
- сообщения оператору.

В зависимости от особенностей документы допускается объединять в отдельные разделы или вводить новые.

2 Содержание разделов

2.1 В разделе «Назначение программы» должны быть указаны сведения о назначении программы и информация, достаточная для понимания функций программы и ее эксплуатации.

2.2 В разделе «Условия выполнения программы» должны быть указаны условия, необходимые для выполнения программы (минимальный и (или) максимальный состав аппаратурных и программных средств и т. п.).

2.3 В разделе «Выполнение программы» должна быть указана последовательность действий оператора, обеспечивающих загрузку, запуск, выполнение и завершение программы, приведено описание функций, формата и возможных вариантов команд, с помощью которых оператор осуществляет загрузки и управляет выполнением программы, а также ответы программы на эти команды.

2.4 В разделе «Сообщения оператору» должны быть приведены тексты сообщений, выдаваемых в ходе выполнения программы, описание их содержания и соответствующие действия оператора (действия оператора в случае сбоя, возможности повторного запуска программы и т. п.).

2.5 Допускается содержание разделов иллюстрировать поясняющими примерами, таблицами, схемами, графиками.

2.6 В приложения к руководству оператора допускается включать различные материалы, которые целесообразно включать в разделы руководства.

Приложение Д (рекомендуемое)

Вариативная часть

1 Разработка программы учета товаров на складе.

Программа предоставляет сведения о товарах, имеющихся на складе: наименование товара; количество единиц товара; цена единицы товара; дата поступления товара на склад; ФИО зарегистрировавшего товар.

Индивидуальное задание: вывести в алфавитном порядке список товаров, хранящихся более х месяцев, стоимость которых превышает у рублей (х, у вводятся с клавиатуры).

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

2 Разработка программы распределения мест в общежитии.

Для получения места в общежитии формируется список студентов, который включает: ФИО студента; номер группы (шесть цифр); средний балл; участие в общественной деятельности; доход на одного члена семьи. Общежитие в первую очередь предоставляется тем студентам, чьи доходы на члена семьи меньше двух минимальных зарплат, затем остальным в порядке уменьшения среднего балла (при равных баллах приоритет отдается тем, кто участвовал в общественной деятельности).

Индивидуальное задание: вывести список очередности предоставления места в общежитии при условии, что размер минимальной зарплаты вводится с клавиатуры.

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

3 Разработка программы расписания движения автобусов.

В справочной автовокзала хранится расписание движения автобусов. Для каждого рейса указаны: номер рейса; тип автобуса; пункт назначения; время отправления; время прибытия на конечный пункт.

Индивидуальное задание: вывести информацию о всех рейсах, которыми можно воспользоваться для прибытия в пункт назначения не позднее чем за 12 часов до заданного времени (интересующее время прибытия вводится с клавиатуры).

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал



пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

4 Разработка программы продажи автобусных билетов.

Автовокзал осуществляет продажу билетов на пригородные рейсы. Для каждого рейса указаны: номер рейса; тип автобуса; пункт назначения; дату отправления; время отправления; время прибытия на конечный пункт; стоимость одного билета; количество оставшихся для продажи билетов; количество проданных билетов.

Индивидуальное задание: обеспечить функциональную возможность покупки билетов на конкретный рейс в требуемом количестве (при этом количество оставшихся для продажи билетов и количество проданных билетов обновляется).

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

5 Разработка программы учета переговоров абонентов сотовой связи.

Оператор сотовой связи хранит информацию о разговорах своих абонентов: номер абонента; ФИО абонента; указание принадлежности вызова к исходящему или входящему; номер исходящего или входящего вызова; дата звонка; время звонка; продолжительность разговора; тариф одной минуты.

Индивидуальное задание: вывести по каждому абоненту за требуемый период времени: перечень входящих и исходящих вызовов, общее время входящих вызовов, общее время исходящих вызовов, общую сумму на исходящие вызовы (требуемый период времени вводится с клавиатуры).

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

6 Разработка программы расчета заработной платы сотрудников предприятия.

Сведения о сотрудниках предприятия содержат: ФИО сотрудника; табельный номер; год; месяц; количество проработанных часов за месяц; почасовой тариф. Рабочее время свыше 144 часов считается сверхурочным и оплачивается в двойном размере.

Индивидуальное задание: рассчитать размер заработной платы каждого сотрудника за вычетом подоходного налога, который составляет 12 % от суммы заработной платы. Определить объем выплат конкретному сотруднику за требуемый период времени (требуемый период времени вводится с клавиатуры).

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

7 Разработка программы учета книг в библиотеке.

Для книг, хранящихся в библиотеке, задаются: регистрационный номер книги; автор; название; год издания; издательство; количество страниц; номер читательского билета (шесть цифр) последнего читателя; отметка о нахождении книги у читателя или в библиотеке в текущий момент.

Индивидуальное задание: вывести список книг с фамилиями авторов в алфавитном порядке, изданных после заданного года (год вводится с клавиатуры). Вывести список книг, находящихся в текущий момент у читателей.

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

8 Разработка программы учета выпускаемой предприятием продукции.

Сведения о выпущенной продукции включают: дату; номер цеха; наименование продукции; количество выпущенных единиц; ФИО ответственного по цеху в данный день.

Индивидуальное задание: для заданного цеха необходимо вывести количество выпущенных изделий по каждому наименованию за требуемый период времени (требуемый период времени вводится с клавиатуры).

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

9 Разработка программы учета стажа сотрудников предприятия.

Информация о сотрудниках предприятия содержит: ФИО сотрудника; дату рождения; название отдела; должность; дату начала работы.

Индивидуальное задание: вывести список сотрудников пенсионного возраста. Вывести список сотрудников в порядке убывания стажа.

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).



10 Разработка программы учета выплат заработной платы сотрудникам предприятия.

Информация о сотрудниках предприятия содержит: ФИО сотрудника, название отдела, должность, размер заработной платы за месяц.

Индивидуальное задание: вычислить общую сумму выплат за месяц по каждому отделу, а также среднемесячный заработок сотрудников по каждому отделу. Вывести список сотрудников, у которых зарплата ниже введенной с клавиатуры.

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

11 Разработка программы учета сведений об абонентах сотовой связи.

Оператор сотовой связи хранит информацию о своих абонентах: ФИО абонента, номер телефона, год подключения, наименование текущего тарифного плана.

Индивидуальное задание: вывести список и подсчитать общее количество абонентов, подключенных с xxxx года (год вводится с клавиатуры).

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

12 Разработка программы ассортимента игрушек в магазине.

Сведения об ассортименте игрушек в магазине включают: название игрушки, цена, изготовитель, количество, минимальная рекомендуемая граница по возрасту.

Индивидуальное задание: вывести список игрушек, которые подходят детям в возрасте x лет (x вводится с клавиатуры).

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

13 Разработка программы ассортимента обуви в магазине.

Сведения об ассортименте обуви в магазине включают: артикул, наименование, цена, изготовитель, размер, количество пар. Артикул начинается с буквы Ж для женской обуви, М для мужской, Д для детской.

Индивидуальное задание: вывести список обуви артикула x и размера y (x, y вводятся с клавиатуры).

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал



пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

14 Разработка программы учета заказов сервисного центра.

В сервисном центре хранится информация обо всех заказах: наименование ремонтируемого изделия (телевизор и т.д.), марка изделия, ФИО владельца, телефон владельца, стоимость ремонта, дата приемки, дата выдачи, статус (выполнен или нет).

Индивидуальное задание: вывести список заказов, невыполненных на текущий момент (сначала – просроченные, затем – ожидающие выполнения в плановом порядке). Вывести общий доход от даты x до даты y (x, y вводятся с клавиатуры).

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

15 Разработка программы учета успеваемости студентов.

Сведения об успеваемости студентов содержат следующую информацию: номер группы (шесть цифр), ФИО студента, сведения о пяти зачетах (зачет/незачет), оценки по пяти экзаменам.

Индивидуальное задание: вывести всех студентов в порядке убывания количества задолженностей (количество задолженностей по каждому студенту необходимо указать); средний балл, полученный каждым студентом группы x (вводится с клавиатуры), и всей группой в целом.

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

16 Разработка программы учета сведений о музыкальном конкурсе.

Информация о конкурсе включает: ФИО участника; год рождения; название страны; наименование музыкального инструмента (гитара, фортепиано, скрипка, виолончель и др.); занятое место по результатам конкурса.

Индивидуальное задание: по каждому классу музыкальных инструментов вывести первые три места с указанием возраста победителей. Вывести список самых молодых (до 12 лет) победителей конкурса в порядке увеличения возраста.

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).



17 Разработка программы учета сведений о пациентах медицинского центра.

Сведения о пациентах медицинского центра содержат: ФИО пациента, пол, дату рождения, место проживания (город), контактный телефон, диагноз.

Индивидуальное задание: вывести иногородних пациентов. Вывести список пациентов старше x лет, у которых диагноз y (x , y вводятся с клавиатуры).

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

18 Разработка программы продажи железнодорожных билетов.

Железнодорожный вокзал осуществляет продажу билетов на поезда дальнего следования. Для каждого поезда указаны: номер поезда; пункт назначения; дату отправления; время отправления; время прибытия на конечный пункт; стоимость одного билета; количество оставшихся для продажи билетов; количество проданных билетов.

Индивидуальное задание: обеспечить функциональную возможность покупки билетов на конкретный рейс в требуемом количестве (при этом количество оставшихся для продажи билетов и количество проданных билетов обновляется). Вывести номер, время отправления и наличие билетов для поездов, прибывающих в город x в интервале от a до b (x , a , b вводятся с клавиатуры).

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

19 Разработка программы планирования факультативных учебных дисциплин для студентов.

Для формирования факультативных занятий необходимо обработать информацию следующего вида: ФИО студента, номер группы, средний балл успеваемости, пять возможных для факультативного посещения дисциплин. Выбираемая дисциплина помечается 1, не выбираемая – 0. Перечень предлагаемых дисциплин: математика, физика, программирование, английский язык, базы данных.

Индивидуальное задание: вывести список и общее количество студентов, желающих прослушать дисциплину x . Если число желающих больше 15, то отобразить 15 студентов с более высоким баллом успеваемости. Вывести предлагаемые дисциплины в порядке убывания популярности с указанием общего числа записавшихся на каждую из них.

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал



пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

20 Разработка программы учета сведений об игроках хоккейной команды.

Сведения об игроках хоккейной команды включают: ФИО игрока, дату рождения, количество сыгранных матчей, число заброшенных шайб, количество голевых передач, количество штрафных минут.

Индивидуальное задание: вывести 6 лучших игроков (голы+передачи) с указанием их результативности.

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

21 Разработка программы продажи авиабилетов.

Авиакомпания осуществляет продажу билетов на самолеты с указанием: номера рейса; типа самолета; пункта назначения; дату вылета; время вылета; время прилета; вместимости самолета; количества оставшихся билетов бизнес-класса; стоимости билета бизнес-класса; количества оставшихся билетов эконом-класса; стоимости билета эконом-класса.

Индивидуальное задание: обеспечить функциональную возможность покупки билетов на конкретный рейс в требуемом количестве (при этом количество оставшихся для продажи билетов обновляется). Если на интересующий рейс нет билетов требуемого класса, то при наличии билетов другого класса на этот рейс – вывести соответствующее информационное сообщение (например, «Билетов эконом-класса на данный рейс нет в наличии, но имеются билеты бизнес-класса в количестве 10.»).

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

22 Разработка программы учета автомобилей таксопарка.

Таксопарк содержит информацию об имеющихся автомобилях: вид автомобиля (такси, микроавтобус, лимузин), вместимость, расход топлива, стоимость автомобиля, количество автомобилей данного вида.

Индивидуальное задание: подсчитать общую стоимость таксопарка. Подсчитать общую стоимость автомобилей каждого вида. Подобрать автомобили, по вместимости соответствующие заданному с клавиатуры диапазону.

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

23 Разработка программы расчета стипендии.

Для расчета стипендии необходимо обработать информацию следующего вида: номер группы, ФИО студента, форма обучения (платная/бюджетная), зачеты по пяти предметам (зачет/незачет), оценки по 4 предметам, признак участия в общественной работе: 1 – активное участие, 0 – неучастие.

Индивидуальное задание: рассчитать стипендию для студентов-бюджетников. При этом отличники (отметки 9, 10) и общественники получают 50 % надбавку, а просто отличники – 25 %. Студенты со средним баллом, равным или ниже 5, стипендию не получают. Базовый размер стипендии вводится с клавиатуры.

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

24 Разработка программы начисления пособий по уходу за ребенком.

Сведения о детях сотрудниц компании содержат следующую информацию: ФИО сотрудницы, дата рождения ребенка, ФИО ребенка. Для каждого ребенка сотрудницы создается отдельная запись указанного вида.

Индивидуальное задание: для каждой сотрудницы вывести количество детей, а также рассчитать общую сумму полагающихся пособий. Пособие начисляется для детей в возрасте до трех лет. Для первого ребенка в семье базовое пособие составляет x (вводится с клавиатуры), для второго – $1,25*x$, для третьего и последующих – $1,5*x$.

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

25 Разработка программы планирования бюджета проекта.

В компании по разработке программного обеспечения для планирования работ по проекту необходимо обработать информацию следующего вида: наименование проекта, вид работ (работа над требованиями, разработка архитектуры, реализация, тестирование), ФИО сотрудника, предполагаемое количество часов, стоимость одного часа.

Индивидуальное задание: для каждого проекта вывести его итоговую стоимость, перечень видов работ с указанием общего количества задействованных специалистов и стоимости данного этапа.

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

26 Разработка программы расчета выплат по больничным листам.

Сведения о больничных сотрудников компании имеют следующий вид: ФИО сотрудника; год; месяц; количество дней, пропущенных по болезни; оплата за один день.

Индивидуальное задание: для месяца x года y вывести список сотрудников с указанием выплат по больничным листам для каждого из них. Вывести общую сумму выплат по больничным листам за интересующий месяц. X , y вводятся с клавиатуры.

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

27 Разработка программы подбора туристической путевки.

Туристическая компания содержит сведения о предлагаемых путевках: тип путевки (отдых, экскурсионный тур, лечение, шопинг, круиз), страна пребывания, вид транспорта, количество дней, питание (завтраки/все включено), стоимость.

Индивидуальное задание: вывести все путевки требуемого типа (вводится с клавиатуры) стоимостью менее x в порядке убывания стоимости.

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

28 Разработка программы учета продаж театральных билетов.

Сведения о продажах билетной кассы содержат следующую информацию: дата, наименование театра, наименование спектакля, количество проданных билетов.

Индивидуальное задание: вывести список самых популярных театров в порядке убывания в указанный месяц. Вывести список самых популярных спектаклей в порядке убывания в указанный месяц. Месяц вводится с клавиатуры.

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

29 Разработка программы учета продаж проездных билетов.

Сведения о продажах проездных билетов содержат следующую информацию: год, месяц, наименование транспорта, количество проданных проездных билетов, стоимость одного проездного билета на данный вид транспорта. Индивидуальное задание: определить общий доход от продажи проездных билетов за определенный месяц. Вывести список самых



востребованных видов транспорта в порядке убывания в указанный месяц. Месяц вводится с клавиатуры.

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

30 Разработка программы учета командировок сотрудников предприятия.

Сведения о командировках сотрудников компании содержат следующую информацию: ФИО сотрудника; год; месяц; длительность командировки в днях; город, в который осуществлялся выезд; сумма командировочных расходов на один день.

Индивидуальное задание: определить общие выплаты командировочных за указанный месяц. Вывести список наиболее часто посещаемых городов в порядке убывания за указанный период (с месяца x по месяц y).

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

31 Разработка программы учета обучающихся в детском центре.

Сведения об обучающихся в детском центре содержат следующую информацию: наименование секции, ФИО ребенка, дата рождения, ФИО родителя, контактный телефон, стоимость обучения в месяц, сумма задолженности по оплате (0 в случае отсутствия задолженности).

Индивидуальное задание: вывести список должников в порядке убывания размера долга. Вывести имеющуюся в центре информацию в порядке увеличения возраста детей.

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

32 Разработка программы учета сведений об игроках футбольной команды.

Сведения об игроках футбольной команды включают: ФИО игрока, дату рождения, количество сыгранных матчей, число забитых мячей, количество голевых передач, количество желтых карточек, количество красных карточек.

Индивидуальное задание: вывести 6 лучших игроков (голы+передачи) с указанием их результативности. Вывести всех игроков, имеющих в активе красные карточки.

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).



33 Разработка программы учета подписчиков на печатную продукцию.

Сведения о подписчиках на печатную продукцию издательского дома содержат следующую информацию: вид издания (газета, журнал), наименование издания, ФИО подписчика, адрес подписчика, длительность подписки (в месяцах), стоимость подписки, дата начала подписки.

Индивидуальное задание: подсчитать общий доход издательского дома за интересующий месяц. Вывести наиболее популярные издания за интересующий месяц.

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

34 Разработка программы расписания движения автобусов.

В справочной автовокзала хранится расписание движения автобусов. Для каждого рейса указаны: номер рейса; тип автобуса; пункт назначения; время отправления; время прибытия на конечный пункт.

Индивидуальное задание: вывести информацию о всех рейсах, которыми можно воспользоваться для прибытия в пункт назначения не позднее чем за 12 часов до заданного времени (интересующее время прибытия вводится с клавиатуры).

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

35 Разработка программы продажи автобусных билетов.

Автовокзал осуществляет продажу билетов на пригородные рейсы. Для каждого рейса указаны: номер рейса; тип автобуса; пункт назначения; дата отправления; время отправления; время прибытия на конечный пункт; стоимость одного билета; количество оставшихся для продажи билетов; количество проданных билетов.

Индивидуальное задание: обеспечить функциональную возможность покупки билетов на конкретный рейс в требуемом количестве (при этом количество оставшихся для продажи билетов и количество проданных билетов обновляется).

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

36 Разработка программы учета переговоров абонентов сотовой связи.

Оператор сотовой связи хранит информацию о разговорах своих абонентов: номер абонента; ФИО абонента; указание принадлежности вызова к



исходящему или входящему; номер исходящего или входящего вызова; дата звонка; время звонка; продолжительность разговора; тариф одной минуты.

Индивидуальное задание: вывести по каждому абоненту за требуемый период времени: перечень входящих и исходящих вызовов, общее время входящих вызовов, общее время исходящих вызовов, общую сумму на исходящие вызовы (требуемый период времени вводится с клавиатуры).

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

37 Разработка программы расчета заработной платы сотрудников предприятия.

Сведения о сотрудниках предприятия содержат: ФИО сотрудника; табельный номер; год; месяц; количество проработанных часов за месяц; почасовой тариф. Рабочее время свыше 144 часов считается сверхурочным и оплачивается в двойном размере.

Индивидуальное задание: рассчитать размер заработной платы каждого сотрудника за вычетом подоходного налога, который составляет 12 % от суммы заработной платы. Определить объем выплат конкретному сотруднику за требуемый период времени (требуемый период времени вводится с клавиатуры).

Общее для всех вариантов задание: реализовать авторизацию для входа в систему (без регистрации!), функционал администратора и функционал пользователя, как минимум три вида поиска, как минимум три вида сортировки (см. более подробно – в общих требованиях к курсовой работе).

