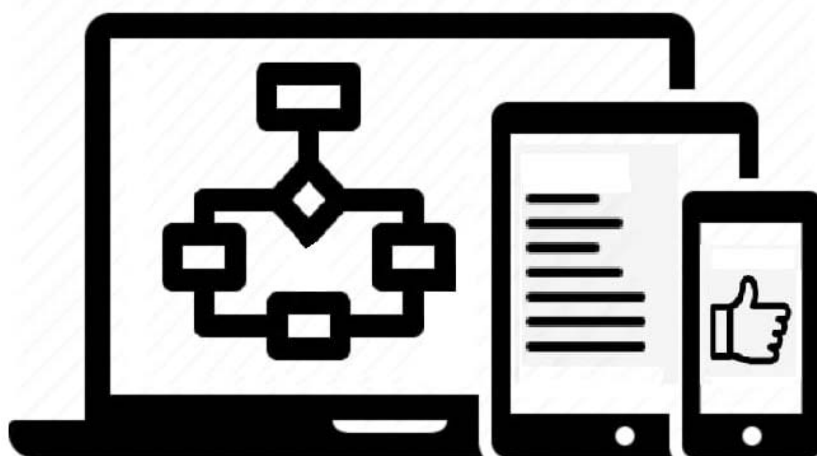


МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»

Кафедра «Программное обеспечение информационных технологий»

ПРОГРАММИРОВАНИЕ

*Методические рекомендации к курсовому проектированию
для студентов направления подготовки
09.03.01 «Информатика и вычислительная техника»
09.03.04 «Программная инженерия»
очной формы обучения*



Могилев 2022

УДК 621.01
ББК 36.4

Рекомендовано к изданию
учебно-методическим отделом
Белорусско-Российского университета

Одобрено кафедрой «Программное обеспечение информационных технологий» 04 марта 2022 г., протокол № 9

Составители: ст. преподаватель О. В. Сергиенко;
канд. тех. наук, ст. преп. Ю. В. Вайнилович;
канд. тех. наук, доц. Н. Н. Горбатенко

Рецензент канд. тех. наук, доц. Ковальчук В. М.

Методические рекомендации содержат требования к курсовому проектированию по дисциплине «Программирование».

Учебно-методическое издание

ПРОГРАММИРОВАНИЕ

Ответственный за выпуск
Технический редактор
Компьютерная верстка

В. В. Кутузов
Т. А. Рыжикова
М. М. Дударева

Подписано в печать 02.11.2022 . Формат 60x84/16. Бумага офсетная. Гарнитура Таймс.
Печать трафаретная. Усл. печ. л. 2,33. Уч.-изд. л. 2,38 . Тираж 21 экз. Заказ № 993.

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, г. Могилев.

© Белорусско-Российский
университет, 2022

Содержание

Введение.....	4
1 Общая формулировка задания.....	5
2 Минимальные функциональные требования к курсовой работе	5
3 Требования к программной реализации курсовой работы	7
4 Содержание пояснительной записки	8
5 Методические рекомендации к содержанию введения, основной части и заключения пояснительной записки.....	9
6 Оформление пояснительной записки.....	19
7 Порядок выполнения и защиты курсовой работы.....	21
Список литературы.....	23
Приложение А. Образец оформления титульного листа.....	24
Приложение Б. Пример описания предметной области.....	25
Приложение В. ГОСТ 19.402-2000 Описание программы	28
Приложение Г. ГОСТ 19.505-79 Руководство оператора. Требования к содержанию и оформлению.....	30
Приложение Д. Вариативная часть задания на курсовое проектирование.....	31

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.

Курсовая работа является одной из форм учебной деятельности, которая выполняется студентом самостоятельно под руководством преподавателя.

Методические рекомендации к курсовому проектированию по дисциплине «Программирование» содержат общие положения, порядок выполнения курсовой работы, структуру курсовой работы, требования к содержанию и оформлению курсовой работы, а также приложения, содержащие примеры оформления структурных элементов курсовой работы.

Методические рекомендации предназначены для оказания помощи студентам в организации работы по курсовому проектированию.

1 Общая формулировка задания

В курсовой работе требуется разработать консольное приложение 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.

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.

Предметная область, для которой будет разрабатываться приложение, выдается преподавателем. Возможны два варианта заданий:

– стандартное задание (приложение Г), за которое выставляется максимальная оценка «хорошо» при условии выполнения требований, оговоренных в п. 7;

– творческое задание, для его постановки студент совместно с преподавателем выбирает актуальную предметную область, предполагающую более тщательный самостоятельный анализ предметной области, проработку структур данных и архитектуры приложения. Студент может претендовать на оценку «отлично» при условии выполнения требований, оговоренных в п. 7.

2 Минимальные функциональные требования к курсовой работе

Разрабатываемое программное средство должно работать минимум для двух категорий пользователей: администратор и пользователь.

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.

Первым этапом работы программы является авторизация – предоставление прав доступа (не путать с регистрацией!). В рамках данного 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.

Обратите внимание: РЕГИСТРАЦИЯ НОВЫХ ПОЛЬЗОВАТЕЛЕЙ ПРИ ВХОДЕ В СИСТЕМУ НЕ ПРЕДУСМОТРЕНА! Данную задачу 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 файлами баз данных:

- создание файла;
- удаление файла.

3 Работа с данными:

а) режим редактирования:

- просмотр всех данных;
- добавление новой записи;
- удаление записи;
- редактирование записи;

б) режим обработки данных:

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

Модуль пользователя включает возможность выбора файла базы данных и работу с данными, предусмотренными п. 3 модуля администратора.

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

Следует предусмотреть:

- обработку исключительных ситуаций:
 - a) введенные пользователем данные не соответствуют формату поля (например, символы в числовом поле);
 - b) введенные пользователем данные некорректны (например, отрицательная цена товара);
 - c) файл с данными для чтения не существует;
 - d) ничего не найдено по результатам поиска;
 - e) номер удаляемой записи выходит за пределы массива/вектора;
 - f) 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.

3 Требования к программной реализации курсовой работы

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

2 Имена методов должны быть осмысленными и строится по принципу глагол + существительное. Если метод выполняет какую-либо проверку и возвращает результат типа `bool`, то ее название должно начинаться с глагола `is` (например, `isFileExist`, `isUnicLogin`).

3 Оператор `goto` под запретом.

4 Код не должен содержать неименованных числовых констант (так называемых «магических» чисел), 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.

5 Код необходимо комментировать (как минимум в части объявления структур, массивов/векторов, заголовков методов, нетривиальной логики).

6 Код не должен дублироваться – для этого существуют методы!

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

8 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 многократного считывания данных из файла в каждом методе).

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

10 Код логически независимых модулей необходимо выносить в отдельные файлы.

4 Содержание пояснительной записки

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 следующие разделы:

- техническое задание;
- описание программы;
- руководство оператора;
- тестирование.

Разделы могут разбиваться на подразделы.

Титульный лист – это первая страница записки (номер страницы на титульном листе не ставится). Пример его оформления приведен в приложении А.

Задание на курсовую работу помещается после титульного листа и не нумеруется. Задание выдается руководителем курсовой работы.

Содержание помещается в записке после задания. Оно состоит из последовательно перечисленных наименований структурных единиц записки начиная с введения с указанием соответствующего номера страницы.

В списке использованных источников перечисляются книги, статьи, источники из интернета, которые были использованы при выполнении работы.

Список использованных источников должен содержать не менее десяти наименований. На все источники должны быть даны ссылки в тексте работы.

Приложения содержат материалы вспомогательного характера: алгоритмы, тексты программ, результаты тестирования, большие таблицы и т. д.

5 Методические рекомендации к содержанию введения, основной части и заключения пояснительной записки

5.1 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.

5.1 Техническое задание

В данном разделе требуется привести развернутое описание предметной области, подробные требования к программе, структуру меню.

В описании предметной области необходимо привести подробное словесное описание объектов предметной области и реальных связей, которые присутствуют между описываемыми объектами. Желательно, чтобы данное описание позволяло корректно определить все взаимосвязи между объектами предметной области. Анализ предметной области должен содержать подробное описание информации об объектах предметной области, которая требуется для решения конкретных задач курсовой работы и которая должна храниться в файлах, формулировку конкретных задач, которые будут решаться с использованием разрабатываемого программного средства с кратким описанием алгоритмов их решения. Также следует описать выходные данные, которые должны генерироваться программой, и входные данные, которые служат хранению и обработке. Кроме того, необходимо указать какие пользователи могут работать с программой и какие задачи будет решать каждый пользователь.

Пример описания предметной области представлен в приложении Б.

Требования к программе включают:

- исходные данные для курсовой работы из раздела «Общие требования к курсовой работе» (выберите основной или альтернативный вариант);
- функциональные требования к конкретно вашей курсовой работе. Рекомендуется взять за основу материал из раздела «Минимальные функциональные требования к курсовой работе» и расширить его для своей темы, например, прописать индивидуальное задание, разновидности поиска и сортировки, конкретизировать возможные исключительные ситуации;
- требования к программной реализации.

Не пишите в требования то, что Вы точно не реализуете в действительности. Для надежности после завершения работы над программой вернитесь к требованиям и проверьте соответствие Вашей программы тому, что Вы заявляли.

Разработка структуры меню подразумевает его графическое представление (пример представлен на рисунке 5.1 (НЕ ДЛЯ КОПИРОВАНИЯ В ВАШУ ПОЯСНИТЕЛЬНУЮ ЗАПИСКУ)).



Рисунок 5.1 – Структура меню

Графическая среда может быть любой; в качестве рекомендации следует обратить внимание на online-редакторы, например, http://www.newart.ru/htm/flash/risovalka_42.php

5.2 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, языки программирования, на которых она написана.

Пример

*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.

Пример

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.

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 методов обработки данных подразумевает перечисление и краткие комментарии о назначении методов реализуемых в них алгоритмов. Методы 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.

Пример

Программное средство учета документации на реализуемый товар состоит из следующих функционально-логических модулей:

- модуль ведения справочников, который содержит необходимую информацию для настройки прикладной программы на реальную предметную область;
 - модуль выполнения расчетов, связанных с предоставлением услуг по доставке продукции (доставка, расчет реализации);
 - модуль оформления документации на реализуемый товар;
 - модуль регистрации организационной и финансовой документации на предоставленные услуги при работе с клиентами;
 - модуль ведения учета клиентов-дебиторов и наличия продукции на складе.
- Логическая схема программного средства приведена на рисунке 5.2.

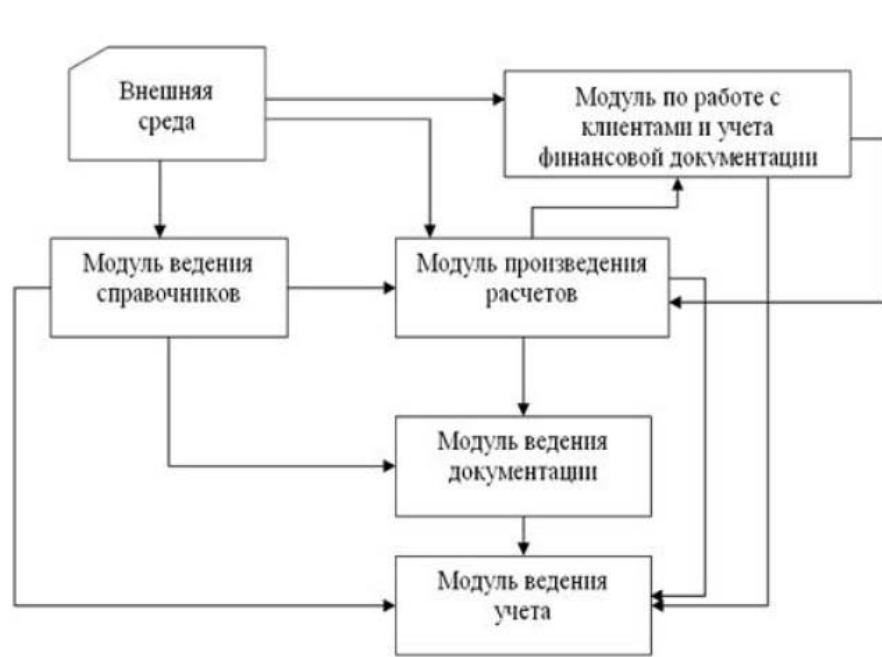


Рисунок 5.2 – Логическая схема программного средства

Модуль ведения справочников содержит следующие методы.

Метод AddTransport позволяет ..., реализует ... содержит ... Блок-схема метода приведена на рисунке ...

Метод ... позволяет ..., реализует ... содержит ... Метод состоит из следующих шагов:

- чтение данных из файла;*
- запрос критерия поиска;*
- валидация введенных пользователем данных;*
- если данные корректны, то осуществляется ... ;*
- если данные некорректны, то ... ;*
- если по запросу ничего не найдено, то ... ;*
- если по запросы найдена хотя бы одна запись, то*

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

Пример

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.

Пример

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. Описание входных данных рекомендуется оформлять в виде таблицы 5.1.

Также в данном разделе описываются способы организации данных:

– в качестве способа описания входных данных: приводится описание типов struct с указанием конкретных полей. В случае объектно-ориентированного программирования приводятся названия предполагаемых

классов и содержащихся в них полей. При работе с базой данных дополнительно приводится структура таблиц;

– в качестве способа объединения входных данных: указывается использование массивов/векторов/перечислений/..., а также их выбранная область видимости (локальные/глобальные).

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

При описании внешних файлов следует описать структуру файла и ограничения, которые на него накладываются.

Пример

Входными данными для программы являются текстовые файлы. Файл user.dat хранит данные о пользователях. Этот файл имеет следующую структуру. Текстовый файл db.dat хранит данные о ... и имеет следующую структуру ...

Таблица 5.1 – Входные данные

<i>Назначение</i>	<i>Обозначение</i>	<i>Тип данных</i>	<i>Диапазон значений</i>
<i>Фамилия сотрудника</i>	<i>surName</i>	<i>string</i>	<i>Ограничений нет</i>
<i>Стаж работы</i>	<i>workExperience</i>	<i>int</i>	<i>0...∞</i>
<i>Должность</i>	<i>Pst</i>	<i>string</i>	<i>Перечисление «Posts»</i>
<i>Критерий поиска сотрудников по стажу работы</i>	<i>experienceFind</i>	<i>string</i>	<i>Ограничений нет</i>

Способом хранения данных о сотрудниках выбрана структура. Она имеет следующие поля: ...

Перечисление Post предназначено для ... и хранит следующие значения ...

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 информация обо всех сотрудниках организации в виде таблицы со следующими полями ...;
- выводимая на экран информация о сотрудниках со стажем, меньшим, чем указано в критерии отбора в виде таблицы с полями ...;
- выводимая в файл информация о сотрудниках, отсортированная по фамилиям в алфавитном порядке. Файл имеет следующую структуру ...

В зависимости от особенностей программы допускается введение дополнительных разделов.

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

5.3 Руководство оператора

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

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

Пример

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.).

Пример

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. Не должны присутствовать слова «откройте», «нажмите», «укажите» и пр. Необходимо 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 5.3).

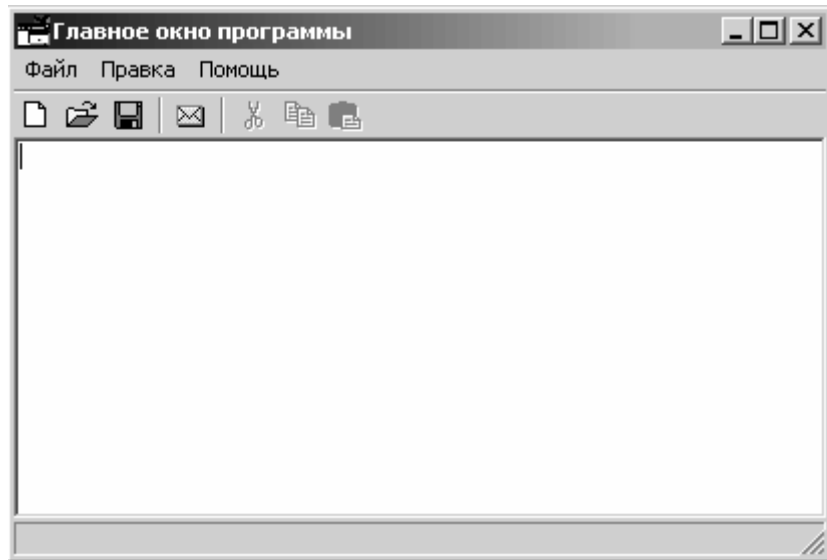



Рисунок 5.3 – Вид формы «Главное окно программы»


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.).

Пример

При попытке сохранения файла с именем уже существующего файла на рабочем столе программы будет отображено сообщение об ошибке, представленное на рисунке 5.4.

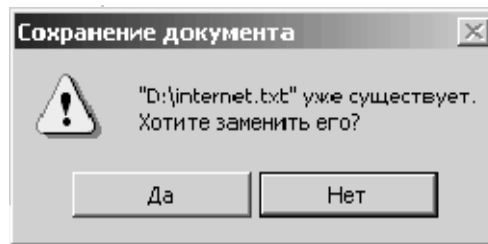


Рисунок 5.4 – Вид сообщения об ошибке

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

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

5.4 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.).

Пример

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, так же как и не обнаружено сбоев в работе модуля при невыполнении тела какого-либо из циклов.

Данные по результатам тестирования каждого метода разработанного программного средства заносятся в таблицу, подобную таблице 5.2.

Таблица 5.2 – Тест ... метода

Действие	Ожидаемый результат	Результат теста
1	2	3
Предусловие:		
Шаг 1		
Шаг 2		
Постусловие:		

Пример

Таблица 4.1 – Тест сортировки данных по фамилии сотрудника

Действие	Ожидаемый результат	Результат теста
1	2	3
Предусловие: на экране отображается главное меню программы		
Выбор пункта главного меню «Сортировка»	Переход в подменю «Сортировка»	Скрин соответствующего экрана
Выбор пункта меню «Сортировать по фамилии»	Вывод на экран списка сотрудников в алфавитном порядке	Скрин соответствующего экрана
Постусловие: осуществлен выход в главное меню, ожидание действий пользователя		

5.5 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.

6 Оформление пояснительной записки

Пояснительную записку выполняют в соответствии с ГОСТ Р 2.105–2019 *Единая система конструкторской документации. Общие требования к текстовым документам с применением ЭВМ.*

Для названий разделов используется шрифт Times New Roman, размер шрифта – 16 пт, начертание полужирное.

Для названий подразделов используется шрифт Times New Roman, размер шрифта – 14 пт, начертание полужирное.

Для основного текста используется шрифт Times New Roman, размер шрифта – 14 пт, межстрочный интервал – 1,5 см, отступ «красной строки» – 1,25 см.

Расстояние от внутренней рамки формата до границ текста:

- слева и справа – 5 мм;
- сверху и снизу – 10 мм.

Для текста в таблицах используется шрифт Times New Roman, размер шрифта – 12 пт, межстрочный интервал – 1 см.

Если наименование таблицы содержит более одной строки используется одинарный межстрочный интервал.

Рисунки размещают по центру строки без абзацного отступа.

Подписи к рисункам центрируют по левому краю с абзацного отступа. Если наименование рисунка содержит более одной строки, используется одинарный межстрочный интервал.

Остальные элементы оформления выполняются в соответствии с ГОСТ Р 2.105–2019.

Пояснительную записку нумеруют сквозной нумерацией, отсчет ведут с титульного листа.

Список использованных источников оформляется в соответствии с ГОСТ Р 7.0.100–2018 «Библиографическая запись. Библиографическое описание. Общие требования и правила составления».

Приложения должны быть перечислены в «Содержании» курсовой работы с указанием их заголовков и страниц. В тексте работы на все приложения должны быть даны ссылки. Приложения располагают в порядке ссылок на них в тексте курсовой работы.

Основные надписи выполняют в соответствии с ГОСТ 2.104–2006 *Единая система конструкторской документации. Основные надписи*.

Содержание оформляют на листе с основной подписью формы 2 (40 мм, ГОСТ 2.104—2006, рисунок 6.1), а последующие листы – с основной надписью формы 2а (15 мм, ГОСТ 2.104–2006, рисунок 6.2).

Приложения оформляются на листах без рамки, включаются в сквозную нумерацию, номер страницы проставляется вверху, посередине листа.

					КР XX.XX.XX.XXXXXXXXXX.XX ПЗ					
Изм.	Лист	№ докум.	Подп.	Дата	Тема курсовой работы Курсовая работа			Лит.	Лист	Листов
Разраб.	ФИО студента								2	
Пров.	ФИО преподав.									
Н.контр.								БРУ гр. № группы		
Утв.										

Рисунок 6.1 – Основная надпись для заглавного листа

					КР XX.XX.XX.XXXXXXXXXX.XX ПЗ					Лист
Изм.	Лист	№ Докум.	Подп.	Дата						4

Рисунок 6.2 – Основная надпись для последующих листов

Расшифровка шифра в основной надписи и на титульном листе приведена на рисунке 6.3.

КР XX.XX.XX.XXXXXXXXXX.XX ПЗ

⏟
⏟
⏟

шифр
номер зачетной
номер по

специальности
книжки
списку

Рисунок 6.3 – Расшифровка шифра

7 Порядок выполнения и защиты курсовой работы

Этапы выполнения курсовой работы приведены в таблице 7.1.

Таблица 7.1 – Этапы выполнения курсовой работы

Планируемая работа	Срок, неделя семестра
Подготовительный этап	
1 Согласование темы курсовой работы с преподавателем	1
2 Получение задания на курсовое проектирование	
Этап разработки технического задания	
3 Анализ предметной области	2–3
4 Разработка требований к программе	
5 Разработка схемы меню	4
Этап разработки программного средства	
6 Разработка структуры данных	5
7 Реализация методов обработки данных	6–12
8 Тестирование программного средства	13
Этап разработки программной документации	
9 Описание программы	14
10 Разработка руководства пользователя	15
Заключительный этап	
11 Оформление пояснительной записки	16–17
13 Защита курсовой работы	

В течение каждого этапа студент ОБЯЗАН не менее трех раз продемонстрировать преподавателю ход выполнения курсовой работы.

В конце каждого этапа студент ОБЯЗАН предоставить преподавателю законченную работу по данному этапу.

По результатам этапа анализа предметной области студент может написать тезисы доклада и принять участие в научной конференции.

Не позже чем за неделю до защиты курсовой работы на проверку руководителю сдается:

- пояснительная записка в печатном виде в папке-скоросшивателе с подписями студента;
- пояснительная записка в электронном виде в формате .doc;
- программное средство с исходными тестами.

Защита курсовой работы проводится в присутствии руководителя и преподавателей кафедры. На защите студент делает краткое сообщение о теме работы, целях и задачах работы (до 10 мин), затем демонстрирует разработанное программное средство и отвечает на вопросы.

Работа студента оценивается в соответствии со следующими критериями.

Оценка «отлично» ставится, если:

- работа сдана вовремя;
- в течение семестра студент выполнял требования по демонстрации преподавателю хода выполнения курсовой работы в соответствии с этапами, представленными в таблице 7.1;
- имеется публикация (тезис доклада, статья) на научной конференции по тематике курсовой работы (или имеется информация, что материал принят к опубликованию);
- программное средство содержит функционал сверх установленного минимума;
- пояснительная записка содержит все разделы в соответствии с заданием, выполнена в полном объеме, грамотно, отсутствуют заимствования из методических указаний и элементы чужой работы, скопированный и не переработанный текст из интернета. Материал изложен логически связно, последовательно, полно. Используются дополнительные источники информации. Пояснительная записка оформлена аккуратно, в соответствии с требованиями п. 6 методических рекомендаций;
- изложение доклада краткое, последовательное, логическое, язык грамотный, выразительный. Выводы аргументированы, доказательны, выделены отличительные черты проекта.

Оценка «хорошо» ставится, если:

- работа сдана с опозданием;
- в течение семестра студент нарушал требования по демонстрации преподавателю хода выполнения курсовой работы в соответствии с этапами, представленными в таблице 7.1;
- отсутствует публикация на научной конференции по тематике курсовой работы;
- реализованы минимальные требования к программному средству;
- пояснительная записка содержит все разделы в соответствии с заданием, грамотно, отсутствуют заимствования из методических указаний и элементы чужой работы, скопированный и непереработанный текст из интернета. Материал изложен логически связно, последовательно. Часть пояснений изложена кратко, формально. Пояснительная записка оформлена аккуратно, но имеются грамматические ошибки, несущественные отклонения от требований п. 6 методических рекомендаций;
- содержание и основная цель проекта раскрыты. Доклад изложен достаточно последовательно, грамотно, с выделением главных моментов. Отдельные принятые решения обоснованы недостаточно убедительно. В отдельных случаях имеется неправильное использование терминологии.

Оценка «удовлетворительно» ставится, если:

- работа сдана с опозданием;

– в течение семестра студент нарушал требования по демонстрации преподавателю хода выполнения курсовой работы в соответствии с этапами, представленными в таблице 7.1;

– отсутствует публикация на научной конференции по тематике курсовой работы;

– пояснительная записка содержит все разделы в соответствии с заданием, имеются заимствования из методических указаний, элементы чужой работы, скопированный и переработанный текст из интернета. Материал изложен логически связно, последовательно. Часть пояснений дана формально. Принятые решения при разработке программного продукта допустимы, но не обоснованы, нерациональны. Пояснительная записка оформлена аккуратно, но имеются грамматические ошибки, несущественные отклонения от требований п. 7 методических рекомендаций;

– содержание и основная цель проекта раскрыты частично. Есть нарушения и в логике, и в последовательности изложения доклада. Прослеживаются затруднения в определении главного и второстепенного при обобщении материала, в аргументации принятых в проекте решений. Допускается множественное неправильное использование терминологии.

Список литературы

1 **Павловская, Т. А.** С#. Программирование на языке высокого уровня: учебник для вузов / Т. А. Павловская. – Санкт-Петербург: Питер, 2019. – 432 с.

2 **Васильев, А. В.** Программирование на С# для начинающих. Основные сведения / А. В. Васильев. – Москва: Эксмо, 2018. – 592 с.

3 **Дорогов, В.Г.** Основы программирования на языке С: учебное пособие / В. Г. Дорогов, Е. Г. Дорогова ; под ред. проф. Л. Г. Гагариной. – Москва: ФОРУМ; ИНФРА-М, 2019. – 224 с.

**Приложение А
(обязательное)**

Образец оформления титульного листа

МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»

Кафедра «Программное обеспечение информационных технологий»

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
к курсовой работе по дисциплине «Программирование»
на тему «Тема курсовой работы»

КР ХХ.ХХ.ХХ.ХХХХХХХ.ХХ ПЗ

Исполнитель	_____	ФИО студента
	(подпись)	
Группа	_____	
Руководитель	_____	должность, ФИО рук.
	(подпись)	
Дата допуска к защите	_____	
Дата защиты	_____	
Оценка	_____	
Комиссия	_____	

Могилев 20XX

Приложение Б (рекомендуемое)

Пример описания предметной области

В работе требуется разработать программное средство для автоматизации учета получения и выдачи книг в библиотеке. Система должна предусматривать режимы ведения системного каталога, отражающего перечень областей знаний, по которым имеются книги в библиотеке. Внутри библиотеки области знаний в систематическом каталоге могут иметь уникальный внутренний номер и полное наименование. Каждая книга может содержать сведения из нескольких областей знаний. Каждая книга в библиотеке может присутствовать в нескольких экземплярах. Каждая книга, хранящаяся в библиотеке, характеризуется следующими параметрами:

- уникальный шифр;
- название;
- фамилии авторов (могут отсутствовать);
- место издания (город);
- издательство;
- год издания;
- количество страниц;
- стоимость книги;
- количество экземпляров книги в библиотеке.

Книги могут иметь одинаковые названия, но они различаются по своему уникальному шифру (ISBN).

В библиотеке ведется картотека читателей.

На каждого читателя в картотеку заносятся следующие сведения:

- фамилия, имя, отчество;
- домашний адрес;
- телефон (будем считать, что у нас два телефона — рабочий и домашний);
- дата рождения.

Каждому читателю присваивается уникальный номер читательского билета.

Каждый читатель может одновременно держать на руках не более 5 книг. Читатель не должен одновременно держать более одного экземпляра книги одного названия.

Каждая книга в библиотеке может присутствовать в нескольких экземплярах. Каждый экземпляр имеет следующие характеристики:

- уникальный инвентарный номер;
- шифр книги, который совпадает с уникальным шифром из описания книг;
- место размещения в библиотеке.

В случае выдачи экземпляра книги читателю в библиотеке хранится специальный вкладыш, в котором должны быть записаны следующие сведения:

- номер билета читателя, который взял книгу;
- дата выдачи книги;
- дата возврата.

Предусмотреть следующие ограничения на информацию в системе:

- книга может не иметь ни одного автора;

- в библиотеке должны быть записаны читатели не моложе 17 лет;
- в библиотеке присутствуют книги, изданные начиная с 1960 по текущий год.
- каждый читатель может держать на руках не более 5 книг;
- каждый читатель при регистрации в библиотеке должен дать телефон для связи: он может быть рабочим или домашним;
- каждая область знаний может содержать ссылки на множество книг, но каждая книга может относиться к различным областям знаний.

С данной информационной системой должны работать следующие группы пользователей:

- библиотекари;
- читатели;
- администрация библиотеки.

При работе с системой библиотекарь должен иметь возможность решать следующие задачи:

- принимать новые книги и регистрировать их в библиотеке;
- относить книги к одной или к нескольким областям знаний;
- проводить каталогизацию книг, то есть назначение новых инвентарных номеров вновь принятым книгам, и, помещая их на полки библиотеки, запоминать место размещения каждого экземпляра;

– проводить дополнительную каталогизацию, если поступило несколько экземпляров книги, которая уже есть в библиотеке, при этом информация о книге в предметный каталог не вносится, а каждому новому экземпляру присваивается новый инвентарный номер и для него определяется место на полке библиотеки;

– проводить списание старых и не пользующихся спросом книг. Списывать можно только книги, ни один экземпляр которых не находится у читателей. Списание проводится по специальному акту списания, который утверждается администрацией библиотеки;

– вести учет выданных книг читателям, при этом предполагается два режима работы: выдача книг читателю и прием от него возвращаемых им книг обратно в библиотеку. При выдаче книг фиксируется, когда и какой экземпляр книги был выдан данному читателю и к какому сроку читатель должен вернуть этот экземпляр книги. При выдаче книг наличие свободного экземпляра и его конкретный номер могут определяться по заданному уникальному шифру книги или инвентарный номер может быть известен заранее. Не требуется вести «историю» чтения книг, то есть требуется отражать только текущее состояние библиотеки. При приеме книги, возвращаемой читателем, проверяется соответствие возвращаемого инвентарного номера книги выданному инвентарному номеру, и она ставится на свое старое место на полку библиотеки;

– проводить списание утерянных читателем книг по специальному акту списания или замены, подписанному администрацией библиотеки;

– проводить закрытие абонемента читателя, т. е. уничтожение данных о нем, если читатель хочет выписаться из библиотеки и не является ее должником, то есть за ним не числится ни одной библиотечной книги;

Читатель должен иметь возможность решать следующие задачи:

- просматривать системный каталог, т. е. перечень всех областей знаний, книги по которым есть в библиотеке;

– по выбранной области знаний получить полный перечень книг, которые числятся в библиотеке;

– для выбранной книги получить инвентарный номер свободного экземпляра книги или сообщение о том, что свободных экземпляров книги нет. В случае отсутствия свободных экземпляров книги читатель должен иметь возможность узнать дату ближайшего предполагаемого возврата экземпляра данной книги. Читатель не может узнать данные о том, у кого в настоящий момент экземпляры данной книги находятся на руках (в целях обеспечения личной безопасности держателей требуемой книги).

– для выбранного автора получить список книг, которые числятся в библиотеке.

Администрация библиотеки должна иметь возможность получать сведения о должниках – читателях библиотеки, которые не вернули вовремя взятые книги; сведения о книгах, которые не являются популярными, т. е. ни один экземпляр которых не находится на руках у читателей; сведения о стоимости конкретной книги, для того чтобы установить возможность возмещения стоимости утерянной книги или возможность замены ее другой книгой; сведения о наиболее популярных книгах, т. е. таких, все экземпляры которых находятся на руках у читателей.

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 нецелесообразно включать в разделы руководства.

Приложение Д

Вариативная часть задания на курсовое проектирование

1 Разработка программы учета товаров на складе.

Программа предоставляет сведения о товарах, имеющих 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 x месяцев, стоимость которых превышает y рублей (x , y вводятся с клавиатуры).

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.

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 ч до заданного времени (интересующее время прибытия вводится с клавиатуры).

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).

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).

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 ч 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).

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.

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).

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.

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.

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).

12 Разработка программы ассортимента игрушек в магазине.

Сведения об ассортименте игрушек в магазине включают: название игрушки, цену, изготовителя, количество, минимальную рекомендуемую границу по возрасту.

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

13 Разработка программы учета ассортимента обуви в магазине.

Сведения об ассортименте обуви в магазине включают: артикул, наименование, цену, изготовителя, 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).

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).

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.

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.

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).

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).

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, невыбираемая – 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.

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 шесть лучших игроков (голы + передачи) с указанием их результативности.

21 Разработка программы продажи авиабилетов.

Авиакомпания осуществляет продажу билетов на самолеты с указанием: номера рейса; типа самолета; пункта назначения; даты вылета; времени вылета; времени 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»).

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.

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 четырем 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.

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 $1,5*x$.

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.

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.

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.

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.

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.

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).

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.

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 шесть лучших игроков (голы + передачи) с указанием их результативности. Вывести всех игроков, имеющих в активе красные карточки.

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.

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 ч до заданного времени (интересующее время прибытия вводится с клавиатуры).

35 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).

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).

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 ч 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).