

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ТУЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Институт прикладной математики и компьютерных наук
Кафедра информационной безопасности

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
ПО ВЫПОЛНЕНИЮ КУРСОВОЙ РАБОТЫ

по дисциплине
ПРОГРАММИРОВАНИЕ

Уровень профессионального образования: *высшее образование – бакалавриат*
Направление подготовки: *090303 Прикладная информатика*
Профиль подготовки: *Прикладная информатика в экономике*
Квалификация выпускника: *бакалавр*

Форма обучения: *очная*

Тула 2015г.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	3
2. ОСНОВНЫЕ ТРЕБОВАНИЯ К КУРСОВОЙ РАБОТЕ.....	4
2.1. Тематика курсовой работы	4
2.2. Исходные данные к курсовой работе	4
2.3. Задание на курсовую работу.....	4
2.4. Объем курсовой работы, стадии и этапы разработки	4
2.5. Работа над курсовой работой	5
2.6. Защита курсовой работы.....	5
3. МЕТОДИЧЕСКИЕ УКАЗАНИЯ К КУРСОВОЙ РАБОТЕ.....	7
3.1. Содержание разделов пояснительной записки	7
3.2. Рекомендации по алгоритмизации, программированию и отладке	8
1. ЦЕЛЬ И ЗАДАЧИ КУРСОВОЙ РАБОТЫ	3
4. ЗАДАНИЯ НА КУРСОВУЮ РАБОТУ	12
ПРИЛОЖЕНИЕ 1	15
ПРИЛОЖЕНИЕ 2	16
ПРИЛОЖЕНИЕ 3	17

ВВЕДЕНИЕ

Характерной особенностью научно-технического прогресса в нашей стране является широкое внедрение вычислительной техники во все отрасли народного хозяйства. Предполагается значительно расширить работы по созданию программного обеспечения (ПО) ЭВМ.

Выполнение курсовой работы должно способствовать закреплению, углублению и обобщению знаний, полученных студентами за время обучения, а также позволит приобрести опыт проектирования и разработки программ, усвоить конкретные методы и технологию программирования, привить навыки работы с технической документацией, научной и справочной литературой.

1. ЦЕЛЬ И ЗАДАЧИ КУРСОВОЙ РАБОТЫ

Курсовая работа является завершающим этапом изучения дисциплин "ПРОГРАММИРОВАНИЕ" и "ИНФОРМАТИКА".

Основная цель работы заключается в закреплении навыков самостоятельного использования теоретического материала и практического опыта для решения задач проектирования и разработки ПО систем различного назначения. Курсовая работа предполагает выполнение всех основных стадий и этапов разработки программ и программной документации в соответствии с существующими государственными и отраслевыми стандартами.

2. ОСНОВНЫЕ ТРЕБОВАНИЯ К КУРСОВОЙ РАБОТЕ

Основными требованиями к курсовой работе являются обязательное выполнение в процессе проектирования требований ЕСПД [1-6], использование при разработке программ современных методов и приемов программирования, строгое планирование процесса проектирования, составления и отладки программ, безусловное выполнение графика курсовой работы.

2.1. Тематика курсовой работы

Курсовая работа посвящена разработке 5 отдельных программ.

2.2. Исходные данные к курсовой работе

Исходные данные определяются преподавателем при выдаче задания на курсовую работу. В процессе работы они могут уточняться или изменяться преподавателем.

2.3. Задание на курсовую работу

Задание на курсовую работу выдается индивидуально каждому студенту. Задание на курсовую работу оформляется студентом после постановки задачи на проектирование преподавателем и имеет единую форму, приведенную в прил.1. Задание подписывается студентом и руководителем и помещается за титульным листом в пояснительной записке. Задание является документом, на основании которого и выполняется курсовая работа.

2.4. Объем курсовой работы, стадии и этапы разработки

Курсовая работа предполагает выполнение всех основных этапов и стадий разработки программ, регламентируемых ГОСТ 19.102-77, к которым относятся:

- техническое задание ;
- эскизный проект ;
- технический проект ;
- рабочий проект ;
- внедрение.

На стадии технического задания производится постановка задачи, определяются общие требования к программе, выполняется технико-экономическое обоснование разработки программы, определяются входные и выходные данные, выбирается язык программирования, определяются требования к техническим средствам.

На стадии эскизного проекта уточняются методы решения задачи, определяется структура входных и выходных данных, разрабатывается общее

описание алгоритма решения задачи.

В рамках технического проекта производится детальная разработка алгоритма решения задачи, определяются формы представления входных и выходных данных, разрабатывается структура программы, окончательно определяется конфигурация технических средств.

На стадии рабочего проектирования производится программирование задачи, отладка программы, составление тестов для испытания программы, разрабатывается документация в соответствии с ЕСПД. Заключительным этапом рабочего проектирования является проведения испытания программы на контрольном примере, осуществление корректировки программы по результатам испытаний. На стадии внедрения осуществляется передача программы и программной документации в эксплуатацию, оформление акта о внедрении программы.

Хотя перечисленные стадии характерны для разработки больших программных систем и комплексов, они обязательно в той или иной мере должны быть отражены при выполнении курсовой работы.

2.5. Работа над курсовой работой

Курсовая работа рассчитана на 16 недель. Весь объем работы рекомендуется выполнять по следующему графику:

- 1 неделя - постановка задачи;
- 2 неделя - разработка технического задания;
- 3 неделя - составление эскизного проекта;
- 4 - 5 недели - техническое проектирование;
- 4 - алгоритмизация задачи;
- 5 - разработка структуры программы, входных и выходных данных;
- 6 - 13 недели - рабочее проектирование;
- 6 - 8 - программирование задачи;
- 9 - 12 - отладка программы;
- 13 - испытание программы;
- 14 неделя - разработка программной документации;
- 15 неделя - оформление пояснительной записки;
- 16 неделя - защита курсовой работы.

Ход выполнения курсовой работы каждым студентом фиксируется преподавателем в графике, который вывешивается на стенде курсового проектирования кафедры.

2.6. Защита курсовой работы

После выполнения всех этапов проектирования студент сдает руководителю на проверку пояснительную записку. Проверив представленные к

защите пояснительные записки, руководитель назначает срок защиты курсовой работы. Затем пояснительная записка оценивается рецензентом. Защита происходит перед комиссией в присутствии студентов данной группы. Студент готовит доклад на 5-10 минут, в котором отражает основные результаты работы, доводит его до сведения комиссии, используя при этом пояснительную записку и необходимый графический материал. После защиты члены комиссии оценивают выполнение курсовой работы и руководитель выставляет отметку.

3. МЕТОДИЧЕСКИЕ УКАЗАНИЯ К КУРСОВОЙ РАБОТЕ

3.1. Содержание разделов пояснительной записки

Перечень разделов пояснительной записки приведен в прил.2.

Во введении определяется цель курсовой работы, дается краткое содержание поставленной задачи.

Раздел "Техническое задание" составляется в соответствии с ГОСТ 19.201-78 [3] и включает подразделы:

- назначение разработки;
- требование к программе;
- стадии и этапы разработки.

В подразделе "Назначение разработки" приводятся исходные данные и тема курсовой работы, определяется функциональное назначение программы.

В подразделе "Требования к программе" указываются требуемые функциональные характеристики программы, обосновывается выбор языка программирования, определяется состав технических средств с указанием их основных технических характеристик, входные и выходные данные.

В подразделе "Стадии и этапы разработки" устанавливаются необходимые стадии разработки, этапы и содержание работ, а также сроки разработки.

Раздел "Описание программы" составляется в соответствии с ГОСТ 19.402-78 [4] и включает подразделы:

- функциональное назначение;
- описание логической структуры;
- входные данные;
- выходные данные.

В подразделе "Функциональное назначение" указываются наименования и назначение программы, язык программирования, специальное программное обеспечение, сведения о функциональных ограничениях на применение программы.

В подразделе "Описание логической структуры" указывается структура программы с описанием функций подпрограмм и связей между ними, приводится схема алгоритма программы. Описание логической структуры программы выполняется с учетом текста программы на исходном языке, который приводится в приложении.

В подразделе "Входные и выходные данные" описывается организация входных и выходных данных, приводятся форматы и способы кодирования данных.

Раздел "Инструкция по эксплуатации программы" составляется на основании ГОСТов [6-7] и включает подразделы:

- назначение программы;
- выполнение программы;

- описание контрольного примера.

В подразделе "Назначение программы" приводятся краткие сведения о назначении, функциях программы и условиях ее эксплуатации.

В подразделе "Выполнение программы" указывается последовательность действий, обеспечивающих запуск, выполнение и завершение программы; приводится описание режимов функционирования программы; указывается набор сообщений, выдаваемых в ходе выполнения программы.

В подразделе "Описание контрольного примера" приводятся сведения о функционировании программы на примере текстового набора данных; описываются входные данные, последовательность работы с программой, приводятся полученные результаты.

В разделе «Заключение» делается вывод по курсовой работе (по каждому этапу курсовой), приводятся полученные результаты.

3.2. Рекомендации по алгоритмизации, программированию и отладке

При разработке алгоритма следует пользоваться методом последовательной (поэтапной) детализации.

При этом на первом этапе составляется укрупненный алгоритм, затем выделяются его составные части, для выполнения которых используются стандартные процедуры. Алгоритм должен строиться по модульному принципу, который предполагает представление алгоритма в виде отдельных относительно независимых частей. При построении алгоритма следует пользоваться ограниченным числом типовых структур. Для представления схемы алгоритма рекомендуется применять графический способ описания в соответствии с ГОСТом 19.701-90 [1]. Полученный алгоритм должен обладать свойствами определенности, массовости, результативности, инвариантности.

При программировании задачи необходимо пользоваться принципами модульного программирования. Данная технология программирования заключается в разбиении задачи на определенное число программных модулей. При этом каждый модуль должен обладать свойствами функциональной законченности и минимальной связности. Такое представление программы позволяет вести параллельную разработку программы и легко устранять ошибки при отладке. При использовании неструктурированных языков программирования следует придерживаться методов структурного программирования, которые сводятся к использованию базовых типов структур (линейной, разветвляющейся, циклической). При этом рекомендуется применять как можно меньше операторов GOTO.

Разработанная программа должна быть универсальной, т.е. допускать многократное выполнение и обеспечивать независимость от конкретных данных. При реализации ввода-вывода необходимо выбирать такое представление данных, которое бы обеспечивало максимальное удобство при

подготовке данных, минимум возможных ошибок и удобочитаемость полученных результатов. При вводе необходимо всегда проверять достоверность данных и предусмотреть защиту от неправильных данных.

Текст программы должен легко восприниматься и быть удобочитаемым. Для этого необходимо использовать пробелы в операторах, пропуски и отступы строк, насыщать программу комментариями. Комментарии делятся на вводные и пояснительные. Вводные комментарии располагаются в начале программы и содержат краткие сведения о программе. Пояснительные комментарии сопровождают участки программы, которые трудно воспринимаются. При этом комментарии должны объяснять назначение группы операторов, а не описывать действия, выполняемые этими операторами. Средней нормой можно считать одну строку комментариев на 10 строк программы. Располагать комментарии в программе надо так, чтобы они были заметными. Для этого можно использовать пропуск строк, отделять комментарии заметными символами.

Отладка программы заключается в определении, локализации и устранении ошибок в программе. Отладку программы рекомендуется проводить с помощью специальных отладочных средств, которые входят в состав языка программирования. Допускается также использовать собственные средства отладки, которые вводятся в программу при ее написании. К таким отладочным средствам относятся: отображение значений переменных, отслеживание хода выполнения программы и обращение к подпрограммам. Процесс отладки производится в два этапа - сначала проводят автономную отладку подпрограммы, а затем и комплексную отладку всей программы.

После того, как программа становится работоспособной, производится ее тестирование, задачей которого является проверка правильности функционирования программы во всем диапазоне допустимых значений исходных данных. Тестирование производится с помощью контрольного примера. При составлении контрольного примера необходимо предусмотреть, чтобы программа была апробирована во всех режимах ее функционирования.

3.3. Оформление пояснительной записки

Материал в пояснительной записке размещается в последовательности, указанной в приложении 2. Образец титульного листа приведен в приложении 3, а бланк задания в приложении 1. Текстовая часть пояснительной записки, а также схемы алгоритмов и программа, которые являются основными результатами выполнения курсовой работы, должны выполняться в строгом соответствии с требованиями ЕСПД [1-8]. В приложении представляются листинг программы. Листинг программы должен быть разделен на страницы и сложен соответствующим образом. Пояснительная записка оформляется на сброшюрованных листах белой бумаги формата А4 в объеме 20-25 страниц. Поля страницы: слева - 30 мм, справа - 10 мм, сверху и снизу - 15 мм.

Основной текст рекомендуется выравнивать по ширине страницы, шрифт Times New Roman высотой 14пт с интервалом 21пт между строками. *РАЗДЕЛЫ* должны иметь порядковые номера в пределах всего документа, обозначенные арабскими цифрами без точки и записанные с абзацного отступа. *ПОДРАЗДЕЛЫ* должны иметь нумерацию в пределах каждого раздела. Номер подраздела состоит из номеров раздела и подраздела, разделенных точкой. В конце номера подраздела точка не ставится.

Формулы набираются с помощью редактора формул Microsoft Equation.

Схемы алгоритмов выполняются с помощью программ Visio, Dia.

Нумерация страниц сквозная и проставляется в правом нижнем углу страницы. Первой страницей является титульный лист, на котором номер страницы не ставится. Затем помещается бланк задания, который включается в нумерацию. Третьей страницей является содержание, которое нумеруется.

Рисунки нумеруются последовательно арабскими цифрами и сопровождаются содержательными подрисуночными подписями. Номер и название рисунка располагается внизу по центру относительно самого рисунка. Пример оформления рисунка приведен на рис.1.

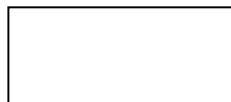


Рисунок 1- Схема программы

Рис.1. Пример оформления рисунка

На все рисунки должны быть ссылки в тексте, при этом указывается слово <рисунок> и указывается его номер. Рисунки помещаются после первого упоминания в тексте. Если рисунок размещается на нескольких листах, то на всех последующих листах пишется <Продолжение рисунка > и указывается его номер.

Цифровой материал, помещенный в записке, рекомендуется оформлять в виде таблицы. Таблицу помещают после первого упоминания в тексте. (Над правым верхним углом таблицы помещается надпись "Таблица" с указанием ее порядкового номера, затем, через тире, следует заголовок таблицы. При ссылке на таблицу в тексте пишется слово <таблица> полностью затем указывается ее номер. Пример оформления таблицы представлен на рис.2.

Таблица 1 – Структура исходных данных

Поле1	Поле2			ПолеN

Рис.2. Пример оформления таблицы

Формулы в записке должны быть пронумерованы. Номер формулы следует заключать в круглые скобки и помещать на правом поле страницы на уровне нижней строки формулы, к которой он относится. При ссылке на формулу необходимо указывать ее номер в скобках.

При оформлении текстовой части допускается сокращение слов, терминов, обозначений только общепринятых. Изложение должно быть кратким и четким.

ЗАПИСКА ПОДПИСЫВАЕТСЯ СТУДЕНТОМ И РУКОВОДИТЕЛЕМ.

4. ЗАДАНИЯ НА КУРСОВУЮ РАБОТУ

Задание №1. Тема «Рекурсивные алгоритмы»

Составить два варианта программы с использованием рекурсивного и итерационного алгоритмов

Вычислить функцию с заданной точностью:

$$\operatorname{arctg} x = \sum_{n=0}^{\infty} (-1)^n \cdot \frac{x^{2n+1}}{2n+1} \quad \text{Для } |x| < 1.$$

Точность вычисления задается студентом самостоятельно

Задание № 2 Тема «Текстовые файлы»

Создать 3 текстовых файла, используя простейший текстовый редактор. Разработать программу для работы с текстовыми файлами. Программа должна уметь делать следующее:

Объединять несколько текстовых файлов в один (дописывание содержимого указанных файлов к первому). Имена и последовательность объединяемых файлов задаются преподавателем при оценке работоспособности программы.

Задание № 3 Тема «Стеки»

Требуется разработать и отладить программу с применением стека, выполняющую указанные в индивидуальном задании действия.

Дано выражение $a*b+c*d$. Организовать вычисление этого выражения, используя алгоритм польской записи. При вычислениях применить программный стек.

Задание № 4 Тема «Односвязные списки»

Необходимо разработать и отладить программу решения задачи с использованием динамических объектов типа односвязный список.

1. Записать в список L N действительных чисел. Перенести 3-й элемент в начало списка. Исходный и модифицированный списки вывести на экран.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ГОСТ 19.701-90. Схемы алгоритмов, программ, данных и систем. Условные обозначения и правила выполнения. – Взамен ГОСТ 19.002-80, ГОСТ 19.003-80; введ. 01.01.1992. – М.: Межгосударственный совет по стандартизации, метрологии и сертификации, 1991. – 23 с. - (Единая система программной документации).
2. ГОСТ 19.102-77. Стадии разработки. – Введ. 01.01.1980. – М.: Межгосударственный совет по стандартизации, метрологии и сертификации, 1991. – 2 с. - (Единая система программной документации).
3. ГОСТ 19.201-78. Техническое задание. Требования к содержанию и оформлению. – Введ. 01.01.1980. – М.: Межгосударственный совет по стандартизации, метрологии и сертификации, 1991. – 2 с. - (Единая система программной документации).
4. ГОСТ 19.402-78. Описание программы. Введ. 01.01.1980. – М.: Межгосударственный совет по стандартизации, метрологии и сертификации, 1991. – 2 с. - (Единая система программной документации).
5. ГОСТ 19.404-79. Пояснительная записка. Требования к содержанию и оформлению. – Введ. 01.01.1981. – М.: Межгосударственный совет по стандартизации, метрологии и сертификации, 1991. – 2 с. - (Единая система программной документации).
6. ГОСТ 19.504-79. Руководство программиста. Требования к содержанию и оформлению. – Введ. 01.01.1980. – М.: Межгосударственный совет по стандартизации, метрологии и сертификации, 1991. – 1 с. - (Единая система программной документации).
7. ГОСТ 19.505-79. Руководство оператора. Требования к содержанию и оформлению. – Введ. 01.01.1980. – М.: Межгосударственный совет по стандартизации, метрологии и сертификации, 1991. – 2 с. - (Единая система программной документации).
8. ГОСТ 2.105-95. Общие требования к текстовым документам. - Взамен ГОСТ 2.105-79, ГОСТ 2.906-71; введ. 01.07.1996. – Минск: Межгосударственный совет по стандартизации, метрологии и сертификации, 1995. – 19 с. - (Межгосударственный стандарт. Единая система конструкторской документации).
9. Грызлов В.И. Турбо Паскаль 7.0 : самоучитель / В.И.Грызлов, Т.П.Грызлов. — 4-е изд., испр. — М. : ДМК Пресс, 2005 .— 416с. : ил. (10)
10. Попов В.Б. Паскаль и Дельфи : учебный курс / В.Б.Попов [и др.].— М.: Питер, 2005 .— 576с. : ил. (10)
11. Вирт Н. Алгоритмы и структуры данных с примерами на Паскале/ Н.Вирт; пер.с англ. 2-е изд.- СПб.: Невский Диалект, 2007. – 352 с. (7)
12. Фаронов В.В. Турбо Паскаль 7.0. Начальный курс : учеб.пособие для вузов / В.В.Фаронов .— М. : Кнорус, 2006 .— 576с. : ил. (7)

13. Секунов Н.Ю. Программирование на C++ в Linux / Н.Ю.Секунов .— СПб. : БХВ-Петербург, 2004 .— 368с. (4)
14. Фленов М.Е. Программирование на C++ глазами хакера / М.Е.Фленов .— СПб. : БХВ-Петербург, 2005 .— 336с (5)
15. Шмидский Я.К. Программирование на языке C+ : Самоучитель / Я.К.Шмидский .— М. и др. : Диалектика, 2004 .— 368с. (5)
16. Turbo Pascal 7.0 / Е.Р.Алексеев и [др.] – М.:ИТ Пресс, 2007. – 270 с.:ил.

УТВЕРЖДАЮ
ЗАВ.КАФЕДРОЙ

_____ г
« _____ » _____ 20 _____ г

З А Д А Н И Е

на курсовой проект (работу) по курсу _____

студенту гр. _____

Тема проекта: _____

Входные данные: _____

Задание выдал: _____

« _____ » _____ 20 _____ г.

Задание получил: _____

« _____ » _____ 20 _____ г.

График выполнения проекта: _____

Замечания консультанта:

К защите. Консультант проекта _____

« _____ » _____ 20 _____ г.

Графические материалы (схема, планировка, эскизы и др.) прикладываются к заданию.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ТУЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

Институт прикладной математики и компьютерных наук

Кафедра информационной безопасности

(тема курсовой работы)

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
к курсовой работе
по дисциплине

(полное наименование учебной дисциплины)

Студент гр.

_____ (индекс группы)

_____ (подпись и дата)

_____ (инициалы и фамилия)

Руководитель

_____ (должность и
ученая степень)

_____ (подпись и дата)

_____ (инициалы и фамилия)

ПЕРЕЧЕНЬ РАЗДЕЛОВ ПОЯСНИТЕЛЬНОЙ ЗАПИСКИ

Титульный лист

Бланк задания

Содержание

ВВЕДЕНИЕ

ЗАДАНИЕ 1

1 ТЕХНИЧЕСКОЕ ЗАДАНИЕ

- 1.1 Назначение разработки
- 1.2. Требования к программе
- 1.3 Стадии и этапы разработки

2 ОПИСАНИЕ ПРОГРАММЫ

- 2.1 Функциональное назначение
- 2.2 Описание логической структуры
- 2.3 Входные и выходные данные

3 ИНСТРУКЦИЯ ПО ЭКСПЛУАТАЦИИ ПРОГРАММЫ

- 3.1 Назначение программы
- 3.2 Выполнение программы
- 3.3 Описание контрольного примера

ЗАДАНИЕ 2

1 ТЕХНИЧЕСКОЕ ЗАДАНИЕ

- 1.1 Назначение разработки
- 1.2. Требования к программе
- 1.3 Стадии и этапы разработки

2 ОПИСАНИЕ ПРОГРАММЫ

- 2.1 Функциональное назначение
- 2.2 Описание логической структуры

2.3 Входные и выходные данные

3 ИНСТРУКЦИЯ ПО ЭКСПЛУАТАЦИИ ПРОГРАММЫ

3.1 Назначение программы

3.2 Выполнение программы

3.3 Описание контрольного примера

.....

ЗАДАНИЕ 5

1 ТЕХНИЧЕСКОЕ ЗАДАНИЕ

1.1 Назначение разработки

1.2. Требования к программе

1.3 Стадии и этапы разработки

2 ОПИСАНИЕ ПРОГРАММЫ

2.1 Функциональное назначение

2.2 Описание логической структуры

2.3 Входные и выходные данные

3 ИНСТРУКЦИЯ ПО ЭКСПЛУАТАЦИИ ПРОГРАММЫ

3.1 Назначение программы

3.2 Выполнение программы

3.3 Описание контрольного примера

ЗАКЛЮЧЕНИЕ

Библиографический список

Приложения