

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ КУРСОВОГО ПРОЕКТА ПО ДИСЦИПЛИНЕ «ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ»

Курсовой проект предусмотрен программой изучения дисциплины «Основы алгоритмизации и программирования», является самостоятельной работой студента, позволяет оценить качество знаний и отражает приобретенные студентом практические навыки.

Тема и задание формулируется руководителем курсового проекта и утверждается заведующим кафедрой.

Перед студентом ставится задача разработать приложение в среде программирования Visual Studio с целью решения конкретной задачи (задач).

Результатом решения являются:

- 1)исполняемый файл программы и ее полный текст (листинг) на носителе;
- 2)пояснительная записка.

Проект должен иметь четкое и логическое построение. Он должен включать следующие структурные элементы (в порядке их представления в работе):

1. титульный лист пояснительной записки;
2. бланк задания к проекту (заполненный руководителем и подписанный обеими сторонам и – преподавателем и студентом, и утвержденный заведующим кафедрой);
3. содержание (оглавление);
4. введение;
5. основная часть (техническая);
6. заключение;
7. список использованных источников;
8. приложения.(Приложение А Блок –схемы алгоритмов программы, Приложение Б Текст программы)

Титульный лист является первым листом работы (нумерация на нем не проставляется). Титульный лист должен содержать все установленные реквизиты.

Бланк задания на тему курсового проекта, подписанный преподавателем, и студентом располагается за титульным листом. Задание не входит в нумерацию страниц работы.

Содержание проекта является третьей по порядку страницей курсового проекта, которое оформляется, используя возможности программы Microsoft Word.

Слово «СОДЕРЖАНИЕ» записывают в виде заголовка по центру текста. В содержание включаются все заголовки, имеющиеся в работе, в том числе список использованных источников и приложения. Название разделов и подразделов следует приводить в полном соответствии с их названиями, указанными в тексте проекта.

Содержание курсового проекта должно включать введение, три –четыре раздела основной части, заключение, список литературы и приложения.

В содержание включаются разделы и подразделы с указанием страниц.

При заполнении штампа тема проекта должна в точности соответствовать заданию на курсовой проект.

Шифр документа содержит буквы КП или КР, номер зачетной книжки, номер программы, разработанной студентом в ВУЗе (присваивается студентом самостоятельно, обычно нумеруются все курсовые работы и проекты), номер документа ПЗ –81 и версия документа ПЗ – 00.

Пример: КП.АС10.89977 – 11 81 00

При размещении содержания на нескольких листах штамп высотой 40 мм выполняется только на первом листе содержания.

Во **введении** кратко характеризуется проблема, решению которой посвящен курсовой проект, определяются цель и задачи, которые необходимо решить для раскрытия темы, описываются средства, посредством которых реализуется разрабатываемая программа, например, среда разработки Visual Studio.

Общий объем введения составляет 1–2 страницы. Введение, как и заключение, рекомендуется писать после полного завершения основной части.

В **основной (технической) части** дается краткая характеристика объекта исследования; анализируется организационное построение и его структура.

Целесообразно учитывать общую теоретическую направленность курсового проекта, то есть акцент следует сделать на те показатели, которые наиболее тесно связаны с темой проекта. Основная часть должна содержать конкретный практический материал исследуемого объекта.

Здесь необходимо отразить не только результаты самостоятельно проведенных обследований, но и состояние исследуемой задачи, предложить пути ее решения на рассматриваемом объекте.

Заключение содержит перечисление основных результатов, характеризующих полноту решения поставленных задач и подводящих итог содержания курсового проекта. Результаты следует излагать в форме констатации фактов с использованием слов «изучены», «сформулированы», «разработаны», «показаны», «предложены», «подготовлены» и т. п. Текст должен быть кратким и ясным.

В разделе «**Список использованных источников**» приводится список литературы, использованной в ходе выполнения курсового проекта. Данный раздел должен содержать перечень источников, цитируемых и используемых при написании курсового проекта, которые следует располагать по алфавиту авторов или заглавий (в случае четырех авторов и более). Сведения об источниках необходимо привести в соответствии с требованиями ГОСТ 7.1–2003.

Приложения. Приложения оформляются как продолжение работы на последующих ее страницах, располагая их в порядке появления ссылок на них в тексте. В приложения следует включать вспомогательный материал, необходимый для полноты восприятия: формы, структуры таблиц базы данных, таблицы вспомогательных данных, схемы алгоритмов и программ, разработанных в процессе выполнения проекта, иллюстрации вспомогательного характера, текст программы. Текст программы должен содержать достаточное количество комментариев (вводных и контекстных). Во вводном комментарии программного модуля должны содержаться: назначение модуля, сведения о разработчике, сведения об обрабатываемых в данном модуле данных и т.п. Каждая процедура и функция пользователя должны также содержать сведения о назначении данной подпрограммы, ее применении, описание формальных параметров.

Общий объем курсового проекта без учета приложений должен составлять 35 -40 страниц.

В общем случае техническая часть пояснительной записки курсового проекта должна содержать описание следующих этапов создания программного средства:

1. постановка задачи, анализ предметной области и формулировка требований к программе;
2. проектирование задачи;
3. реализацию программы (программирование);
4. руководство оператора (внедрение).

Первый раздел содержит анализ и постановку задачи: исследование предметной области, определение круга задач, который должна решать программа, выделение ключевых абстракций, обоснование необходимости компьютерной обработки информации или процессов, анализ существующих решений.

Дается точное описание исходных данных, условий задачи и целей ее решения.

Описываются ограничения, накладываемые на исходные данные, необходимая разрядность и точность представления исходных данных и результатов решения. Указываются возможные пределы изменения входных параметров задачи.

Второй раздел отражает этап проектирования задачи, который заключается в разработке моделей, позволяющих понять структуру будущей системы и процесс ее функционирования, сбалансировать требования и наметить схему ее применения, определить виды выходных данных, их представление и способы хранения, вычислительную систему.

В разделе дается обобщенное словесное описание алгоритма решения поставленной задачи, излагаются основные требования к алгоритму и пути их реализации.

Приводится схема алгоритма, состоящая из укрупненных модулей. Дается пояснение назначения и состава каждого модуля. Обобщенный алгоритм обычно использует обозначения и термины исходной задачи.

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

В разделе приводятся описания процедур.

Результатом должна стать детализированная модель системы, именно данная модель должна «служить» исходной информацией для написания программного кода.

Третий раздел (программирование) содержит описание программы. Здесь освещаются:

1. обоснование выбранных средств и инструментов разработки;
2. спецификация программы;
3. текст программы;
4. описание программы;
5. тестирование программы.

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

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

Не следует забывать о хорошем стиле программирования. После заголовка процедуры или функции записывается комментарий, содержащий поясняющий текст, а именно: назначение подпрограммы; перечень и назначение формальных параметров,

их тип. Комментариями должны быть снабжены и основные смысловые блоки программы или подпрограммы.

Результатом данного этапа является программное приложение, которое обладает требуемой функциональностью и способно решать нужные задачи в конкретной предметной области.

На этапе тестирования программы проводится проверка работоспособности программы на некоторой совокупности исходных данных или при некоторых специальных режимах эксплуатации. Результатом является повышение надежности программы, исключаящее возникновение критических ситуаций.

Четвертый раздел – руководство оператора (внедрение). В разделе описываются:

1. условия выполнения программы;
2. выполнение программы;
3. сообщения оператору.

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

В пункте «Выполнение программы» должна быть указана последовательность действий оператора, обеспечивающих загрузку, запуск, выполнение и завершение программы. В разделе приводятся сведения для проверки, обеспечения функционирования и настройки программы на условия конкретного применения. Перечисляются порядок и последовательность ввода исходных данных и получения результатов расчета.

В пункте «Сообщения оператору» должны быть приведены тексты сообщений, выдаваемых в ходе выполнения программы, описание их содержания и соответствующие действия оператора.

Содержание разделов допускается иллюстрировать поясняющим и примерами, таблицами, схемами.

Требования к оформлению курсового проекта

Курсовой проект оформляется на стандартных листах формата А4.

Сокращение слов, кроме общепринятых, не допускается. Текст излагается от третьего лица или в форме безличных предложений без использования личных местоимений.

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

К оформлению работы предъявляются следующие требования:

- 1) односторонняя печать;
- 2) текст размещается по ширине листа;
- 3) поля: слева 30 мм, справа 10, снизу и сверху – 20 мм;
- 4) Межстрочный интервал 1,2 (допускается использование междустрочного интервала 1,0 в таблицах и рисунках)
- 5) шрифт Times New Roman, кегль – 13 пт.
- 6) номер страницы проставляется внизу, справа, шрифт 10 пунктов;
- 7) абзацный отступ должен быть одинаковый – 15 мм;
- 8) каждая глава начинается с новой страницы;
- 9) все заголовки глав и параграфов должны быть выделены полужирным шрифтом и без точки;
- 10) расстояние между нумеруемым заголовком любого уровня и текстом – 3 межстрочных интервала (интервал после абзаца – 39 пт. для шрифта 13). Если между двумя заголовками текст отсутствует (например, между заголовками раздела и подраздела), то расстояние между ними – 2 межстрочных интервала;

11) сокращения слов в таблицах и рисунках не допускается (разрешается в таблицах и рисунках необходимые надписи делать более мелким шрифтом, чем в текстовой части);

12) ссылки на литературу приводятся в тексте в квадратных скобках (например [1], [4-8]).

13) Для оформления заголовков, списков, формул, подписей таблиц, рисунков и т.п. выбирается стиль на усмотрение студента, не противоречащий данному документу, который должен быть применен для всего документа, т.е. все единицы текста должны быть оформлены единообразно.

Порядок нумерации работы следующий. На первой странице располагается титульный лист (номер страницы не ставится). Затем постранично (последовательно) размещаются задание на разработку курсового проекта, содержание и т.д. Нумерация страниц работы начинается с титульного листа.

Все нумерации (страниц, глав, параграфов, рисунков и т.д.) выполняют только арабскими цифрами. При необходимости в оглавление и, соответственно, в основную часть вводят рубрикации типа 1.1 или 2, 2.2, 2.2.1 и т.п..

Каждая глава нумеруется и начинается с новой страницы.

Рисунки

Рисунки нумеруются в пределах раздела, содержат номер раздела, точку, номер рисунка в разделе. На каждый рисунок в тексте ПЗ должна быть ссылка.

Примеры:

Разработанная структурная схема приведена на рисунке 3.1.

Или ... (см. рисунок 3.1).

Рисунки могут располагаться в любом месте ПЗ, но обязательно после ссылки. Под рисунком помещается подпись по центру, которая содержит слово Рисунок, пробел, номер рисунка (номер раздела, точка, номер рисунка в разделе), тире, название рисунка (без точки).

Рисунки отделяются от текста одной пустой строкой сверху и снизу (после подписи). Допускается выносить рисунок на отдельный лист и поворачивать его по часовой стрелке (в альбомной ориентации), чтобы верх рисунка находится около переплета. В этом случае подпись рисунка выполняется также в альбомной ориентации, а расположение рамки и штампа не меняется.

Допускается располагать рисунок на нескольких страницах, при этом на первой странице делается обычная подпись рисунка, а на последующих подпись выглядит так:

Продолжение рисунка 3.1

и располагается по центру (без названия).

Таблицы

Таблицы нумеруются в пределах раздела, содержат номер раздела, точку, номер таблицы в разделе.

Таблица должна быть отцентрирована относительно текста. Ячейки могут объединяться. Допускается уменьшать шрифт в таблице. Минимальное расстояние между строками таблицы – 8 мм.

На каждую таблицу в тексте ПЗ должна быть ссылка.

Примеры:

Результаты экспериментов приведены в таблице 3.1.

Или ... (см. таблицу 3.1).

Таблицы могут располагаться в любом месте ПЗ, но обязательно после ссылки. Над таблицей помещается заголовок по центру, которая содержит слово Таблица, пробел, номер таблицы (номер раздела, точка, номер таблицы в разделе), тире, название таблицы (без точки).

Таблицы отделяются от текста одной пустой строкой сверху (до заголовка) и снизу .

Допускается выносить таблицу на отдельный лист и поворачивать его по часовой стрелке (в альбомной ориентации), чтобы верх таблицы находится около переплета. В этом случае заголовок таблицы выполняется также в альбомной ориентации, а расположение рамки и штампа не меняется.

Допускается располагать таблицу не нескольких страницах, при этом на первой странице делается обычный заголовок, а на последующих заголовок выглядит так:

Продолжение таблицы 3.1

и располагается по центру (без названия). В продолжении таблицы нужно повторить шапку (названия граф).

Не допускаются пустые графы в таблице. Если по смыслу не требуется указывать значение, то в этой графе нужно поставить прочерк.

Пример оформления таблицы приведен в приложении.

Формулы

Формула пишется на отдельной строке и отделяется от текста одной пустой строкой сверху и снизу, начинаются в строке с отступом, равным абзацу. Формулы нумеруются в пределах раздела (номер раздела, точка, номер формулы), номер помещается в последней строке формулы в круглых скобках с правой стороны на расстоянии абзацного отступа от правого края текста.

Ссылка на формулу может выглядеть так:

... по формуле (3.1).

Если нет необходимости использовать ссылку на формулу в тексте, то ее можно не нумеровать.

Каждый символ, входящий в формулу, должен быть пояснен после формулы. Первая строка пояснения начинается со слова «где», далее после следует символ, тире, его пояснение. Каждый символ начинается с новой строки под предыдущим символом.

Пояснение может отсутствовать только в случае, если в ранее приведенных формулах этот символ уже использовался и пояснение его было приведено.

Если необходимо привести численные вычисления, то сначала приводится формула с пояснением обозначений, затем в последующих строках приводится числовое выражение и результат. При этом единицы измерений результата пишутся в круглых скобках.

Списки

Допускается использовать нумерованные и маркированные списки и их комбинацию. В качестве маркеров можно использовать тире, точку, квадрат, в качестве номера пункта списка – арабские цифры, русские и латинские символы, в качестве разделителя – точку, тире, скобку.

Оформление вложенного списка должно отличаться от оформления списка верхнего уровня. Следует придерживаться единообразного оформления списков одного уровня во всей ПЗ. Например, маркированный списки изображать с квадратом, нумерованные списки первого уровня – с арабской цифрой и скобкой, вложенный нумерованный список – латинской буквой и скобкой.

Фрагменты программ

Можно включать в текст ПЗ фрагменты программ, файлов, консольные команды и т.д. Их рекомендуется выносить на отдельные строки, отделяя от текста одной пустой строкой сверху и снизу. Начинать на строке их следует с абзацного отступа. Рекомендуется использовать шрифт Courier New. Примеры:

```
lea dx, a
mov ax, dx
```

```
C:> echo "Good Morning"
```

Задание на курсовое проектирование

При выполнении курсового проекта студент выполняет все задания. При отсутствии хотя бы одного задания, курсовой проект не может быть зачтен.

Номер варианта индивидуального задания на курсовое проектирование определяется исходя из последних двух цифр номера зачетной книжки студента.

Ниже приведена таблица, в которой содержатся соответствия последних двух цифр номера зачетной книжки и номеров вариантов индивидуального задания на курсовое проектирование.

Не допускается выполнение студентом вариантов заданий, не соответствующих последним двум номерам его зачетной книжки.

Последние два номера зачетной книжки	Варианты заданий на курсовое проектирование
	Задание
01	01
02	02
03	03
04	04
05	05
06	06
07	07
08	08
09	09
10	10
11	11
12	12
13	13
14	14
15	15
16	1
17	2
18	3
19	4
20	5
21	6
22	7
23	8

24	9
25	10
26	11
27	12
28	13
29	14
30	15

Обобщенная формулировка задания. Выбрать предметную область (в соответствии с вариантом) для базы данных и предложить структуру для описания отдельных записей базы данных. Выбранная структура должна иметь не менее пяти полей (элементов) двух и более типов, включая пользовательский тип union и enum.

Задание 1 (общее для всех студентов).

Осуществлять запуск исполняемого файла с командной строки с одним параметром командной строки. Пример,

C:\имя_файла.exe Иванов

Параметр командной строки -это пароль, который позволит пользователю запустить программу на выполнение при совпадении значения параметра командной строки и параметра сохраненного в строковой переменной программы.

Пример реализации:

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char *argv[])
{
    if(argc!=2) {
        printf("Вы забыли ввести свое имя.\n");
        exit(1);
    }
    printf("Привет %s", argv[1]);
    return 0;
}
```

Если СИ-программа не использует передачу аргументов и значений параметров операционной системы, то целесообразно запретить использование библиотечных функций _setargv и _setenv поместив в СИ-программу перед функцией main функции с такими же именами, но не выполняющие никаких действий (заглушки). Начало программы в этом случае будет иметь вид:

```
_setargv()
{ return ; /* пустая функция */
}
_setenv()
{ return ; /* пустая функция */
}
int main()
{ /* главная функция без аргументов */
...
...
return (0);
}
```

Задание 2 (общее для всех).

Реализовать меню, разработанное с применением массива указателей на функции.

Пример,

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
void f1(float, float);
```



```

void f2(float, float);
void f3(float, float);
int menu();
void(*masf[])(float, float)={f1, f2, f3};
int main(int argc, char *argv[]) {
    float a=0, e=0.00001;
    while(1)
        (*masf[menu()]) (a, e);
    return 0;
}
int menu()
{
    char n;
    do
    {
        printf ("1. Function f1\n ");
        printf ("2. Function f2\n ");
        printf ("3. Function f3\n ");
        printf ("4. Exit\n");
        printf ("Сделайте выбор\n");
        n=getche();
        printf("\n");
        system("cls");
    }
    while(strchr("1234", n)==NULL);
    if(n=='4') exit(0);
    return n-49;
}
void f1(float i, float a)
{
    printf ("Function f1\n ");
}
void f2(float a ,float i)
{
    printf ("Function f2\n ");
}
void f3(float a, float i)
{
    printf ("Function f3\n ");
}

```

Задание 3 (общее для всех)

Для заданной базы данных написать следующие функции:

Задание 3.1. Функцию формирования динамического одномерного массива структур, значения которых вводятся с клавиатуры. Предусмотреть возможность заполнения одного поля структуры, используя известные значения других полей структуры. При вводе структур можно реализовать один из следующих механизмов:

- ввод заранее заданного количества структур;
- ввод до появления структуры с заданным признаком;
- диалог с пользователем о необходимости продолжать ввод.

Задание 3.2. Функцию просмотра содержимого динамического массива структур.

Задание 3.3. Функцию дополнения уже существующего массива структур новыми структурами. Данная функция должна содержать параметр со значение по умолчанию (количество новых структур).

Пример,

```
void func (int a = 0, int b = 10){}
//вызов
func();
func(-1);
func(-1, 99);
```

Задание 3.4. Функцию поиска и вывода на экран структуры (структур) с заданным значением элемента.

Задание 3.5. Функцию упорядочения массива структур по заданному полю (элементу).

Задание 3.6. Организовать функцию с переменным числом параметров

Индивидуальные варианты :

Вариант 1. Структура «Автосервис»: регистрационный номер автомобиля, марка, пробег, мастер, выполнивший ремонт, сумма ремонта.

Вариант 2. Структура «Сотрудник»: фамилия, имя, отчество; должность; год рождения; заработная плата.

Вариант 3. Структура «Государство»: название; столица; численность населения; занимаемая площадь.

Вариант 4. Структура «Гражданин»: фамилия, имя, отчество; домашний адрес; номер телефона; возраст.

Вариант 5. Структура «Читатель»: Фамилия И.О., номер читательского билета, название книги, срок возврата.

Вариант 6. Структура «Школьник»: фамилия, имя, отчество; пол; класс; номер телефона; оценки по предметам (математика, физика, русский язык, литература).

Вариант 7. Структура «Студент»: фамилия, имя, отчество; домашний адрес; группа; рейтинг.

Вариант 8. Структура «Покупатель»: фамилия, имя, отчество; домашний адрес; номер телефона; номер кредитной карточки.

Вариант 9. Структура «Пациент»: фамилия, имя, отчество; домашний адрес; номер медицинской карты; номер страхового полиса.

Вариант 10. Структура «Информация»: носитель; объем; название; автор.

Вариант 11. Структура «Клиент банка»: Фамилия И.О., номер счета, сумма на счете, дата последнего изменения.

Вариант 12. Структура «Склад»: наименование товара, цена, количество, процент торговой надбавки.

Вариант 13. Структура «Авиарейсы»: номер рейса, пункт назначения, время вылета, дата вылета, стоимость билета.

Вариант 14. Структура «Вокзал»: номер поезда, пункт назначения, дни следования, время прибытия, время стоянки.

Вариант 15. Структура «Кинотеатр»: название кинофильма, сеанс, стоимость билета, количество зрителей.