

Разработка консольных приложений на языке Си (Часть 1). Лабораторный практикум по дисциплине "Алгоритмические языки программирования" для студентов специальности 113 Прикладная математика согласно НБ-14-113/16 (Лекции 34 часов, практические занятия 68 часов, самостоятельная работа 108 часов)

А.Г. Пискунов

2 мая 2019 г.

1 ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Целью преподавания дисциплины "Алгоритмические языки программирования" является ознакомление студентов с начальными концепциями, понятиями, методами и технологиям программирования; выработкой у них практических навыков, достаточных для самостоятельной разработки программного обеспечения;

Данная дисциплина состоит из одного тематического модуля, который завершается сдачей модульной работы и домашнего задания.

Сдача каждой работы (все равно лабораторной, модульной или домашнего задания) предполагает передачу файлов исходного кода приложения, файлов с тестами, файлов командного интерпретатора для выполнения тестов и файла с отчетом о работе. Кроме того, отчеты по модульной работе сдаются в напечатанном виде.

2 МОДУЛЬ: ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ СИ

2.1 ЛАБОРАТОРНАЯ РАБОТА: ФАКТОРИАЛ

Тема:

Создание консольного приложения.

Цель:

Любым редактором набрать программу для вычисления факториала и построить приложение.

Задание:

- Создать папку для приложения.
- Набрать программу.
- Построить приложение.

2.2 ЛАБОРАТОРНАЯ РАБОТА: ВЫБОР ЧЕТВЕРТИ НА КООРДИНАТНОЙ ПЛОСКОСТИ

Тема:

Основные элементы языка Си++.

Цель:

Создать приложение для определения четверти, в которую попадает точка на плоскости.

Задание:

- Проанализировать и формализовать поставленную задачу.
- Научиться вводить целое число.
- Научиться использовать оператор ветвления для сложного выбора.

2.3 ЛАБОРАТОРНАЯ РАБОТА: ПОДСЧЕТ ПРОБЕЛОВ В ТЕКСТОВОМ ФАЙЛЕ

Тема:

Создание консольного приложения.

Цель:

Создать приложение для подсчета количество пробелов в текстовом файле.

Задание:

- Использовать оператор цикла `while`.
- Научиться связывать текстовые файлы со стандартным вводом приложения.
- Научиться просматривать все символы файла.
- Познакомиться с правилами преобразования длинных целых, целых, коротких целых и символов.

2.4 ЛАБОРАТОРНАЯ РАБОТА: ТАБЛИЦА ASCII

Тема:

Операторы, операции и выражения.

Цель:

Создать приложение для вывода расширенной таблицы ASCII.

Задание:

- Использовать оператор цикла `for`.
- Выучить правила преобразования символов в целое.
- Познакомиться с знаковыми и беззнаковыми типы.
- Выучить спецификаторы места вывода для функции `printf`.

2.5 ЛАБОРАТОРНАЯ РАБОТА: ТРАНСЛИТЕРАЦИЯ

Тема:

Создание консольного приложения.

Цель:

Создать приложение для замены символы кириллицы текстового файла на символы латиницы.

Задание:

- Научиться редактировать текстовые файлы.

2.6 ЛАБОРАТОРНАЯ РАБОТА: ВЫВОД МАССИВОВ

Тема:

Одномерные массивы и препроцессор.

Цель:

Создать приложение для вывода массивов.

Задание:

- Научиться описывать массивы разными способами.
- Научиться использовать директиву препроцессора `#define` и константы языка Си++.
- Научиться определять длину массива при помощи операции `sizeof`.

2.7 ЛАБОРАТОРНАЯ РАБОТА: ПОИСК МАКСИМУМА И МИНИМУМА

Тема:

Одномерные массивы и препроцессор.

Цель:

Создать приложение для нахождения минимума или максимума в массиве.

Задание:

- Разработать алгоритм нахождения минимума-максимума в целом массиве.
- Повторить знаковые и беззнаковые типы.
- Найти в документации по компилятору стандартные макросы для максимального целого и минимального целого (`INT_MIN`, `INT_MAX`, `UINT_MAX`).

2.8 ЛАБОРАТОРНАЯ РАБОТА: ФУНКЦИЯ ИЗМЕНЕНИЯ ЗНАЧЕНИЙ ДВУХ ПЕРЕМЕННЫХ (swap)

Тема:

Функции в языке Си.

Цель:

Написать функцию для обмена значений двух переменных. Создать приложение для её тестирования. Прототип функции:

```
void swap1 (int left, int right); // функция для обмена значений
//void swap1 (int *left, int *right); // функция для обмена значений
```

Задание:

- Научиться разрабатывать и использовать функции.
- Научиться описывать внешние объекты.
- Научиться разрабатывать приложения, состоящие из более чем одного файла.
- Выучить правила передачи фактических параметров в функцию.

2.9 ЛАБОРАТОРНАЯ РАБОТА: СОРТИРОВКА МАССИВА

Тема:

Функции и массивы в языке Си.

Цель:

Разработать функции для вывода целочисленного массива, выбора индекса максимума (и минимума) из массива, сортировки массива выбором. Прототипы функций:

```
void print (const int arr[], int arrSz, int elemPerLine=5); // вывод массива
int maxIndex (const int arr[], int arrSz); // поиск индекса максимума
void sort (int arr[], int arrSz); // сортировка
```

Создать приложение для тестирования этих функций.

Задание:

- Создать для разрабатываемых функций создать заголовочный файл и применить директиву препроцессора `#include`.
- Научиться программировать сортировку выбором максимума (минимума).
- Познакомиться с квалификатором `const`.
- Научиться передавать одномерные массивы в функции.

2.10 ЛАБОРАТОРНАЯ РАБОТА: СТРОКИ

Тема:

Указатели на символы.

Цель:

Разработать функции копирования и определения длины строки. Прототипы функций:

```
#define UINT unsigned int
UINT strlen1 (const char str[]); // длина строки
void strcpy1 ( char dest[], const char src[]); // копирование строки из src в dest
```

Создать приложение для их тестирования. Строки для тестирования функций передавать в приложение через аргументы командной строки.

Задание:

- Познакомьтесь с передачей аргументов командной строки в приложение.
- Научитесь использовать символьные одномерные массивы для хранения строк.
- Научитесь определять длину строки.
- Научитесь копировать строку в другое место оперативной памяти.
- Познакомьтесь с переопределением функций в языке C++. Добавьте в предыдущую лабораторную работу (см. 2.8) функцию обмена со следующим прототипом:

```
void swap1 (int *left, int *right); // функция для обмена значений
```

2.11 ЛАБОРАТОРНАЯ РАБОТА: РЕКУРСИВНЫЕ ФУНКЦИИ

Тема:

Рекурсия.

Цель:

Создать приложение для простых вычислений при помощи рекурсивной функции. Использовать одно из следующих заданий: вычислить факториал, длину строки или сумму элементов массива. Число для рекурсии получить из аргументов командной строки.

Задание:

- Познакомиться со стандартными функциями для преобразования строчек в числа (`atoi`, `atof`).

```
int atoi (const char * str);
```

- Познакомиться с рекурсией, как методом программирования.
- Научиться разрабатывать рекурсивные функции.
- Научиться передавать численные аргументы командной строки в приложение.

2.12 ЛАБОРАТОРНАЯ РАБОТА: ТЕКСТОВОЕ ПРЕДСТАВЛЕНИЕ ЦЕЛЫХ ЧИСЕЛ

Тема:

Работа с символами в языке Си.

Цель:

Разработать функцию для решения вопроса: представляет ли текст целую константу (не пустая последовательность цифр, которая возможно начинается с символа '-' или '+'). Прототип функции:

```
int isInt (const char *str); // текстовое представление целого числа.
```

Задание:

- Познакомиться с библиотекой обработки символов `ctype.h`.
- Познакомится со способами обработки текстов.
- Научиться записывать формальные требования к лексемам в виде функции.

2.13 ЛАБОРАТОРНАЯ РАБОТА: СРАВНЕНИЕ СТРОК

Тема:

Работа с символами.

Цель:

Разработать функцию для сравнения двух строк. Прототип функции:

```
int strcmp1 (const char *left, const char *right); // сравнение двух строчек.
```

В качестве дополнительной информации выдать длины сравниваемых строк при помощи стандартной функции `strlen`.

Задание:

- Познакомиться с библиотекой обработки строчек `string.h`.
- Познакомиться с директивой препроцессора `#pragma`.
- Для сравнения строчек использовать цикл с постусловием `do - while`.
- Познакомиться с правилами создания консольного приложения и начать обращаться к ключи приложения, в частности, `'-?'` - для вывода справки по использованию приложения и `'-v'` - для вывода дополнительной информации о работе приложения.

2.14 ЛАБОРАТОРНАЯ РАБОТА: ДИНАМИЧЕСКАЯ ПАМЯТЬ

Тема:

Захват и освобождение памяти.

Цель:

Разработать функции для построчного считывания целого файла в оперативную память и освобождения памяти. Прототипы требуемых функций:

```
void addStr ( const char *str); // добавления строчки к памяти
void freeMem ( );           // освобождение памяти
void priMem ( );            // вывод содержимого введенного файла
```

Задание:

- Выучить виды памяти в Си.
- Познакомиться с библиотекой управления памятью `malloc.h` (функции `malloc`, `free`).
- Научиться использовать ключевое слово `static` для инкапсуляции переменной, общей для всех трех перечисленных функций.
- Научиться использовать построчное чтение текстового файла при помощи следующей функции ввода-вывода из `stdio.h`:

```
char *gets(char *str);
```

3 ТЕМАТИКА МОДУЛЬНОЙ РАБОТЫ

3.1 ЗАДАНИЯ ДЛЯ МОДУЛЬНОЙ РАБОТЫ

Разработать консольное приложение для:

1. сложения, отнимания, умножения и деления двух целых чисел.
2. преобразования десятичного целого в двоичную систему исчисления.
3. преобразования десятичного целого в восьмеричную систему исчисления.

- преобразования десятичного целого в шестнадцатиричную систему исчисления.
- изменения порядка элементов массива на обратный.
- отбрасывания лидирующих и завершающих пробельных символов строки. Прототип функции:

```
char * trim (char * str);
```

- выбора значения параметра из строки вида

параметр=значение

Прототип функции:

```
char * getValue (const char * param, const char * str);
```

Если строка param входит в строку str до символа '=', то вернуть указатель на символ следующий за '=', иначе вернуть 0.

Приложение должно:

- данные брать из аргументов командной строки.
- проверять корректность введенной информации при помощи функции isInt.
- по ключам '-?', '/?', '-h', '/h', '-help', '/help' выдавать подсказку для использования.
- результаты вычислений выводить в стандартный вывод.
- состоять более чем из одного файла и иметь глобальную переменную.

3.2 ВОПРОСЫ ДЛЯ МОДУЛЬНОЙ РАБОТЫ

- Что такое интерпретатор? Пример интерпретатора.
- Что такое компилятор?
- Шаги построения консольного приложения.
- Какие есть директивы препроцессора?
- Что такое тип?
- Встроенные типы языка Си?
- Примеры констант основных типов в языке Си.
- Что такое имя?
- Что такое операция? Операции в языке Си.
- Что такое переменная?

11. Что такое выражение?
12. Что такое оператор? Операторы языка Си.
13. Что такое макрос? Указать несколько макросов из стандартных заголовочных файлов.
14. Какое средство языка C++ предназначено для замены макросов с параметрами? Пример.
15. Что такое функция?
16. Что такое область видимости? Области видимости.
17. Какие есть возможности описания составных типов в Си?
18. Операторы ветвления.
19. Операторы цикла.
20. Операторы безусловного перехода.
21. Составной оператор.
22. Оператор выражение.
23. Стандартные файлы ввода-вывода консольного приложения. Способы их перенаправления. Переменные языков C и C++, которые соответствуют перечисленным файлам.
24. Функции ввода-вывода Си.
25. Что такое перегрузка функции?
26. Как передаются фактические параметры в функцию?
27. Стандартное преобразования типов в Си.
28. Каким образом параметры командной строки передаются в приложение?
29. Функции работы со строками в Си.
30. Функции работы с символами в Си.
31. Функции работы с памятью в Си.
32. Что такое рекурсия?
33. Что такое итерация?
34. Как описываются внешние объекты?

4 ТРЕБОВАНИЯ К ОФОРМЛЕНИЮ ОТЧЕТОВ СДАВАЕМЫХ РАБОТ

Титульная страница должна содержать название министерства, университета, института, кафедры, темы модульной работы, исполнителя, преподавателя, город и год.

Объем работы не менее 10 листов. Текст надо набирать шрифтом Times New Roman, 12 pt, интервал между строчками - 1, форматирование по ширине, отступы: 2 см слева, 1.5 - справа, сверху и снизу страницы, абзац - 1 см. Нумерация страниц - внизу по центру.

Список литературы оформлять согласно требований ДСТУ ГОСТ 7.1:2006 "Библиографическая запись. Библиографическое описание. Общие требования и правила составления (ГОСТ 7.1-2003, ИДТ)". - К.: Гостстандарт Украины, 2007.

Отчеты к модульным работам должны содержать следующие части:

- Титульный лист;
- Содержание;
- Постановка задачи;
- Теоретическая часть (описание предметной области, краткие теоретические сведения, необходимые для выполнения задачи, описание инструментов программирования и т.д.);
- Описание алгоритма программы;
- Описание использования приложения;
- Тесты для проверки работоспособности;
- Выводы;
- Список используемой литературы.

В отчетах к лабораторным некоторым частям разрешается опускаться.

5 СПИСОК ЛИТЕРАТУРЫ ДЛЯ КУРСА

- Керниган Б., Ритчи Д. Язык программирования Си, Издательство Невский Диалект, 3-е издание,-2001.- с.290.
- Болски М.И. Язык программирования Си. Справочник: Пер. с англ. - М.: Радио и связь, 1988, - 96 с.
- Страуструп Б. Программирование: принципы и практика использования C++, исправленное издание = Programming: Principles and Practice Using C++ — М.: «Вильямс», 2011. — С. 1248.

- Айвор Хортон Visual C++ 2010: полный курс = Ivor Horton's Beginning Visual C++ 2010 — М.: «Диалектика», 2010. — С. 1216.
- Б. Страуструп. Язык программирования C++ = The C++ Programming Language / Пер. с англ — 3-е изд. — СПб.; М.: Невский диалект — Бином, 1999. — 991 с. — 3000 экз. — ISBN 5-7940-0031-7 (Невский диалект), ISBN 5-7989-0127-0 (Бином), ISBN 0-201-88954-4 (англ.).
- Прата, Стивен. Язык программирования C++. Лекции и упражнения , 5-е изд. Пер. с англ. М. «Вильямс», 2007. — С. 1184с.
- Страуструп Б. Язык программирования C++. Специальное издание = The C++ programming language. Special edition — М.: Бином-Пресс, 2007. — 1104 с.
- Герберт Шилдт Полный справочник по C++, 4-е издание = C++: The Complete Reference, 4th Edition — М.: «Вильямс», 2011. — 800 с.
- Дэвид Р. Мюссер, Жилмер Дж. Дердж, Атул Сейни C++ и STL: справочное руководство, 2-е издание (серия C++ in Depth) = STL Tutorial and Reference Guide: C++ Programming with the Standard Template Library, 2nd edition, (C++ in Depth Series) — М.: «Вильямс», 2010. — 432 с.
- Джесс Либерти, Дэвид Хорват. Освой самостоятельно C++ за 24 часа = Sams Teach Yourself C++ in 24 Hours, Complete Starter Kit — 4-е изд. — М.: Вильямс, 2007. — 448 с.

Содержание

1	ПОЯСНИТЕЛЬНАЯ ЗАПИСКА	2
2	МОДУЛЬ: ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ СИ	2
2.1	ЛАБОРАТОРНАЯ РАБОТА: ФАКТОРИАЛ	2
2.2	ЛАБОРАТОРНАЯ РАБОТА: ВЫБОР ЧЕТВЕРТИ НА КООРДИНАТНОЙ ПЛОСКОСТИ	2
2.3	ЛАБОРАТОРНАЯ РАБОТА: ПОДСЧЕТ ПРОБЕЛОВ В ТЕКСТОВОМ ФАЙЛЕ	3
2.4	ЛАБОРАТОРНАЯ РАБОТА: ТАБЛИЦА ASCII	3
2.5	ЛАБОРАТОРНАЯ РАБОТА: ТРАНСЛИТЕРАЦИЯ	4
2.6	ЛАБОРАТОРНАЯ РАБОТА: ВЫВОД МАССИВОВ	4
2.7	ЛАБОРАТОРНАЯ РАБОТА: ПОИСК МАКСИМУМА И МИНИМУМА	4
2.8	ЛАБОРАТОРНАЯ РАБОТА: ФУНКЦИЯ ИЗМЕНЕНИЯ ЗНАЧЕНИЙ ДВУХ ПЕРЕМЕННЫХ (<i>swap</i>)	5
2.9	ЛАБОРАТОРНАЯ РАБОТА: СОРТИРОВКА МАССИВА	5
2.10	ЛАБОРАТОРНАЯ РАБОТА: СТРОКИ	6
2.11	ЛАБОРАТОРНАЯ РАБОТА: РЕКУРСИВНЫЕ ФУНКЦИИ	6
2.12	ЛАБОРАТОРНАЯ РАБОТА: ТЕКСТОВОЕ ПРЕДСТАВЛЕНИЕ ЦЕЛЫХ ЧИСЕЛ	7
2.13	ЛАБОРАТОРНАЯ РАБОТА: СРАВНЕНИЕ СТРОК	7
2.14	ЛАБОРАТОРНАЯ РАБОТА: ДИНАМИЧЕСКАЯ ПАМЯТЬ	8
3	ТЕМАТИКА МОДУЛЬНОЙ РАБОТЫ	8
3.1	ЗАДАНИЯ ДЛЯ МОДУЛЬНОЙ РАБОТЫ	8
3.2	ВОПРОСЫ ДЛЯ МОДУЛЬНОЙ РАБОТЫ	9
4	ТРЕБОВАНИЯ К ОФОРМЛЕНИЮ ОТЧЕТОВ СДАВАЕМЫХ РАБОТ	11
5	СПИСОК ЛИТЕРАТУРЫ ДЛЯ КУРСА	11