# Министерство образования и науки Российской федерации

# Федеральное государственное автономное образовательное учреждение высшего профессионального образования «Санкт-Петербургский Государственный Электротехнический Университет «ЛЭТИ» им. В. И. Ульянова (Ленина)» (СПбГЭТУ «ЛЭТИ»)

Факультет компьютерных технологий и информатики

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Кафедра вычислительной техники**

**Отчёт по лабораторной работе №6**

по дисциплине «Программирование»

на тему

«Строки в языке С/C++»

Выполнил студент гр.5307 Самоуков Н.В.

Проверил: к.т.н., доцент кафедры ВТ Сискович Т.И.

"Выполнено" "\_\_\_\_"\_\_\_\_\_\_\_\_\_\_

Подпись преподавателя \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Санкт-Петербург**

**2015 г.**

Оглавление

[Цель работы 3](#_Toc438172077)

[1.Задание 3](#_Toc438172078)

[2.Уточнение задания 3](#_Toc438172079)

[3.Контрольные примеры 3](#_Toc438172080)

[4.Описание переменных главной функции 4](#_Toc438172081)

[5.Краткое описание алгоритма 4](#_Toc438172082)

[6.Иерархическая структура программы 5](#_Toc438172083)

[7.Схема алгоритма главной функции 5](#_Toc438172084)

[8.Описание функций 10](#_Toc438172085)

[8.1.Функция str\_emepty 10](#_Toc438172086)

[8.2 Функция razdel 13](#_Toc438172087)

[8.3 Функция copy\_sl 15](#_Toc438172088)

[8.4 Функция comp\_str 17](#_Toc438172089)

[8.5 Функция my\_getch 19](#_Toc438172090)

[8.6 Функция input\_str 21](#_Toc438172091)

[9.Текст программы 23](#_Toc438172092)

[10.Результат 27](#_Toc438172093)

[Вывод 27](#_Toc438172094)

# Цель работы

Получить практические навыки использования строк в языке С/С++.

# 1.Задание

Разработать алгоритм и написать программу, выполняющую многократно по желанию пользователя ввод строки, контрольный вывод, обработку исходной строки и вывод результата. Обработанная строка формируется из слов исходной строки в которых чётное число букв.

# 2.Уточнение задания

Обработанная строка содержит максимум 80 символов. Слова разделяются разделителями. Разделители задаются в программе.

# 3.Контрольные примеры

Контрольные примеры представлены в таблице 1. Контрольные примеры.

Таблица 1. Контрольные примеры

|  |  |  |
| --- | --- | --- |
| **№ примера** | **Исходные данные** | **Результат** |
| **Строка s\_in** | **Строка s\_out** |
| **1** | “The test message.” | “test” |
| **2** | “MMMMMh&&hhhhhh&TRTTT,,,8777” | “MMMMMh hhhhhh 8777” |
| **3** | “hhh hhhhg&&& hhhhhh0” | “” |
| **4** | “” | “” |
| **5** | “ ” | “” |
| **6** | “ ff&&& ” | “ff” |

# 4.Описание переменных главной функции

Переменные, используемые в главной функции, приведены в таблице 2 Переменные, используемые в главной функции.

Таблица 2. Переменные, используемые в главной функции

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Назначение |
| s\_in | char\* | Указатель на первый символ исходной строки |
| s\_out | char\* | Указатель на первый символ обработанной строки |
| i, r | int | Вспомогательные переменные |
| menu | int | Оператор управления меню (1-ввести исходной строки, 2-вывести исходную строку, 3-обработать исходную строку, 4-вывести обработанную строку, 5-справка, 6-выход) |
| ok\_in | int | Флажок наличия исходной строки (1-исходная строка введена, 0-исходная строка не введена) |
| ok\_out | int | Флажок наличия обработанной строки (1-обработанная строка есть, 0-нет обработанной строки) |
| exit | int | Переменная выхода из программы (0-не выходим, 1-выходим) |

# 5.Краткое описание алгоритма

Шаг 1. exit = 0, ok\_in = 0, ok\_out = 0.

Шаг 2. Выводим меню. Вводим menu.

Шаг 3. Если menu==1, то вводим строку с помощью функции input\_str, out\_ok=0,ok\_in=1 и

Шаг 9 Иначе Шаг 4.

Шаг 4. Если menu==2, то если ok\_in==1, то выводим исходную строку и ok\_in = 1, иначе

выводим сообщение, что нет исходной строки. Далее Шаг 9. Иначе Шаг 5.

Шаг 5. Если menu==3, то если ok\_in==1, то если ok\_out==0, то обрабатываем строку с

помощью функции comp\_str и

out\_ok=1, иначе выводим сообщение, что обработка уже проводилась, иначе выводим

сообщение, что нет исходной строки. Далее Шаг 9. Иначе Шаг 6.

Шаг 6. Если menu==4, то если ok\_out==0, то если ok\_in==0, то выводим сообщение, что нет

исходной строки, иначе выводим сообщение, что обработка не проводилась, иначе

выводим обработанную строку. Далее Шаг 9. Иначе Шаг 7.

Шаг 7. Если menu==5, то выводим справку. Далее Шаг 9. Иначе Шаг 8.

Шаг 8. Если menu==6, то exit=1

Шаг 9. Если exit==0, то Шаг 2, иначе конец программы.

# 6.Иерархическая структура программы

Иерархическая структура программы представлена на рисунке 1.

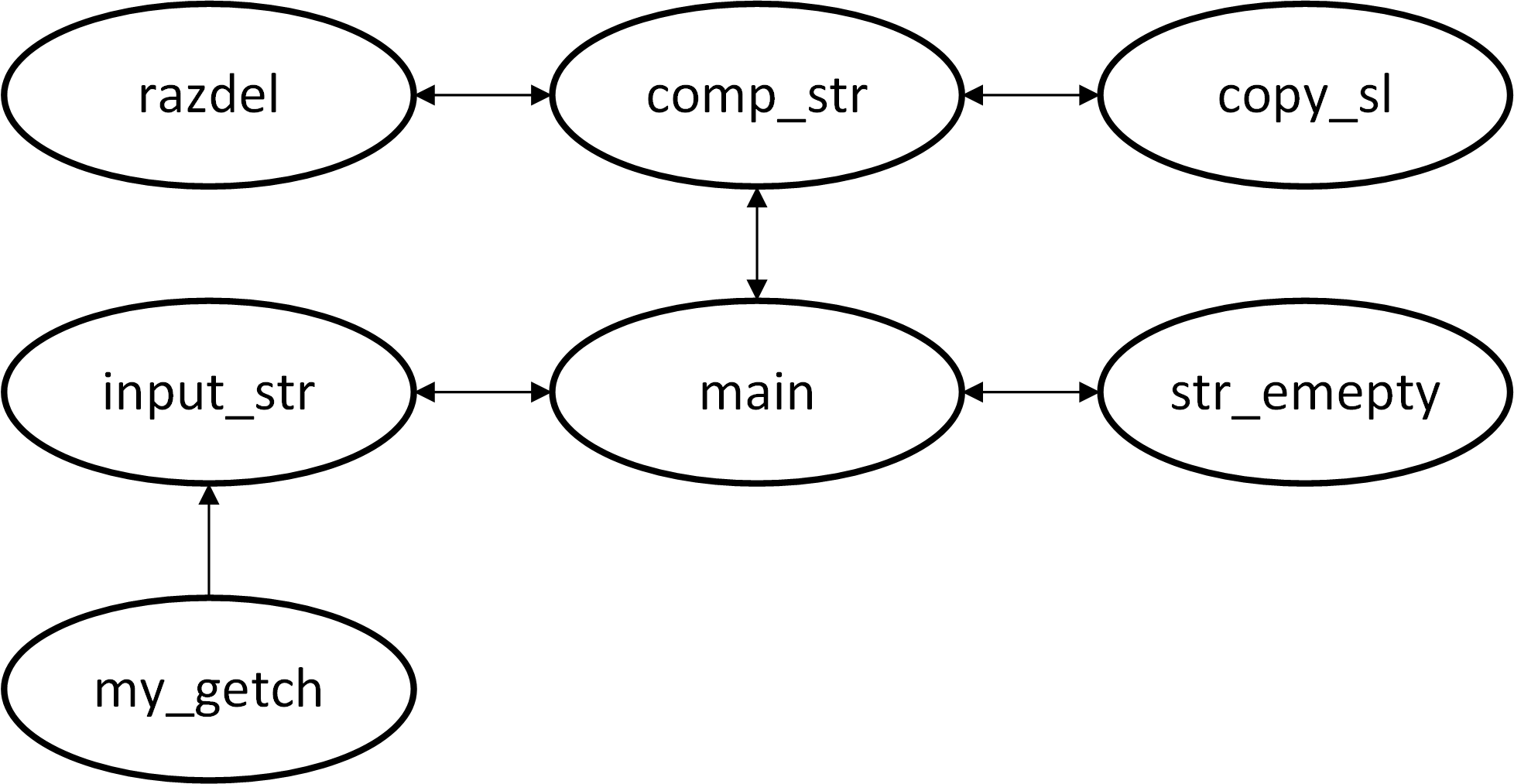


Рисунок 1. Иерархическая структура программы

# 7.Схема алгоритма главной функции

Схема алгоритма представлена на рисунках 2.1-2.5.

Рисунок 2.1. Схема алгоритма главной функции

Рисунок 2.2. Схема алгоритма главной функции

Рисунок 2.3. Схема алгоритма главной функции

Рисунок 2.4. Схема алгоритма главной функции

Рисунок 2.5. Схема алгоритма главной функции

# 8.Описание функций

# 8.1.Функция str\_emepty

**Назначение функции:**

Проверка пустоты строки.

**Прототип:**

bool str\_emepty(char \*);

**Возвращаемое значение:**

Эта функция возвращает 1-если строка пустая, 0-если строка не пустая.

**Пример вызова:**

str\_emepty(s\_in) ? "Внимание пустая строка.\n" : "Сторка не пустая"

**Описание переменных:**

Описание переменных в таблице 3. Переменные функции str\_emepty

Таблица 3. Переменные функции str\_emepty

|  |  |  |  |
| --- | --- | --- | --- |
| **Вид параметра** | **Имя**  **переменной** | **Тип переменной** | **Назначение переменной** |
| **Фактические параметры** | s\_in | char \* | Указатель на первый символ строки. |
| **Формальные параметры** | s\_in | char \* | Указатель на проверяемый элемент строки. |
| **Локальные переменные** | Локальные переменные отсутствуют. | | |

# 9.Текст программы

# 10.Результат

При выполнении программы получены результаты, совпадающие с контрольными примерами, представленными в таблице 1. Контрольные примеры. Ошибки не обнаружены.

# Вывод

В результате выполнения лабораторной работы получены практические навыки работы с строками в языке C/C++.