**МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ**

**федеральное государственное автономное образовательное**

**учреждение высшего образования**

**«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»**

**(РУТ(МИИТ)**

Кафедра «Вычислительные системы, сети и информационная безопасность»

**Отчет По Дисциплине**

**«Алгоритмизация и программирование»**

**Лабораторная работа №3**

*Направление:* 10.03.01*Информационная безопасность*

*Профиль:**Безопасность компьютерных систем*

Выполнил:  
студент группы УИБ-115

Клепиков Степан Даниилович

Проверил:

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

(должность, ФИО)

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

(должность, ФИО)

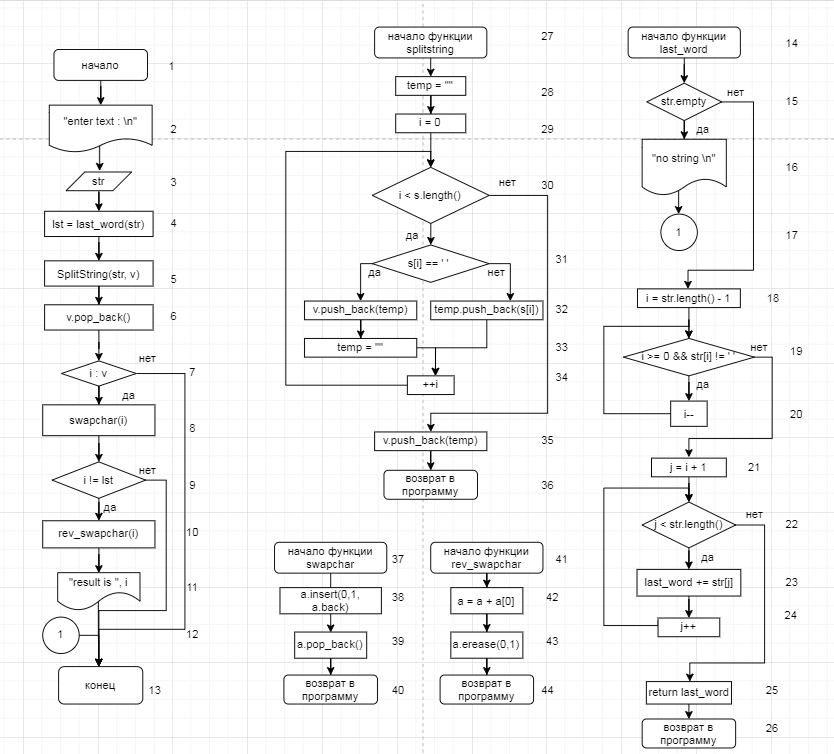
Москва 2022 г.

**Задание**

Задан текст. Напечатать все слова, отличные от последнего слова, предварительно преобразовав каждое из них по следующему правилу: перенести последнюю букву в начало слова. Использовать функции.

1. Таблица имён:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Рабочие переменные | | | | | |
| i | integer | | | Счетчик | |
| v | vector | | | Временная переменная ? | |
| s | string | | | Временная переменная | |
| last\_word | string | | | Переменная с последним словом | |
| temp | string | | | Временная переменная | |
| a | string | | | Временная переменная | |
| lst | integer | | | Переменная вызова функции | |
| Исходные данные | | | | | |
| str | | string | | | Текст |
| Результат | | | | | |
| i | | | auto | | Слова |

1. Блок-схема: ᅠᅠ ᅠᅠ ᅠᅠ ᅠᅠ ᅠ(внешние функции)
2. Отладочный пример:

Основная программа

1. Начало
2. Вывод ‘enter text’
3. Ввод текста (**biba boba biba**)
4. Вызов внешней функции last word (biba)
5. Вызов внешней функции splitstring
6. Удаление последнего элемента
7. Ranged for
8. Вызов внешней функции swapchar (abib abob biba)
9. Проверка условия (i != lst) (abib != biba)
10. Вызов внешней функции rev\_swapchar (abib => biba)
11. Вывод слова (biba)

11 Проверка условия (i != lst) (abob != biba)

12 Вызов внешней функции rev\_swapchar (abob => boba)

13 Вывод слова (boba)

1. Завершение внешней программы при условии..
2. Завершение программы

Внешние функции

1. Начало функции last\_word
2. Проверка на Артема Радиковича Романенко
3. Вывод “string empty”
4. Завершение работы всей программы
5. int i = str.length() – 1 (14-1 = 13)
6. Проверка условия (i >= 0 && str[i] != ' ') (верно)
7. i--

………

1. j = i + 1 (9 + 1)
2. Проверка условия (j < str.length()) (10 < 14 => верно)
3. Заполнение массива str[j] (b)
4. j++

….

1. (косяк в блок-схеме)
2. Возврат в основную программу
3. Начало внешней функции
4. Temp = “”
5. i = 0
6. Проверка условия (i<s.length()) (0 < 14 => верно)
7. Проверка условия (s[i] == ‘ ‘) (b != ‘ ‘ => неверно)
8. S[0] = “”
9. –
10. ++i

………

1. Возврат преобразованного вектора
2. Завершение функции splitstring, возврат к программе
3. Начало функции swapchar
4. Перенос последней буквы в начало
5. Удаление последнего символа
6. Завершение функции swapchar, возврат к программе
7. Начало функции rev\_swapchar
8. Приписывание последней буквы ко всему слову
9. Удаление первой буквы из слова
10. Завершение функции rev\_swapchar, возврат к программе
11. Код программы:

#include <iostream>//подключение библиотек

#include <algorithm>

#include <vector>

#include <string>

using namespace std;

string last\_word(const string& str) //функция поиска последнего слова

{

if (str.empty())//проверка пустую строчку

{

cout << "No string\n";//проверка на Артема Радиковича

}

int i = str.length() - 1;

while (i >= 0 && str[i] != ' ')

{

i--;

}

string last\_word;

for (int j = i + 1; j < str.length(); j++)

{

last\_word += str[j];

}

return last\_word; // возврат функцией последнего слова

}

void SplitString(string s, vector<string>& v) {//антипробел

string temp = "";

for (int i = 0; i < s.length(); ++i) {

if (s[i] == ' ') {

v.push\_back(temp);

temp = "";

}

else {

temp.push\_back(s[i]);

}

}

v.push\_back(temp);

}

void swapchar(string& a) {//смена букв местами

a.insert(0, 1, a.back());

a.pop\_back();

}

void rev\_swapchar(string& a) {//обратная смена для читабельности

a = a + a[0];

a.erase(0, 1);

}

int main() {//основная функция

cout << "Enter text : \n";

string str, lst; //объявления переменных

vector<string> v;

getline(cin, str); //ввод текста

lst = last\_word(str); //вызов функции last\_word

SplitString(str, v); //вызов антипробела

v.pop\_back();//удаление последнего элемента

for (auto i : v) {//ranged for

swapchar(i); //вызов функции смены

if (i != lst) {

rev\_swapchar(i); //вызов обратной функции

cout << "result is : " << i << "\n";//вывод результата

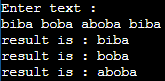
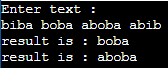
}

}

return 0;

}

1. Результат выполнения работы программы:



1. Вывод:

В ходе выполнения работы были изучены базовые алгоритмы и функции языка высокого уровня C++.

Был произведен анализ их работы, на основании которого была составлена блок-схема. На контрольных примерах было выяснено, что все написанные функции работают корректно и в коде отсутствуют ошибки.

Был оформлен комплект документации на программный код.