

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

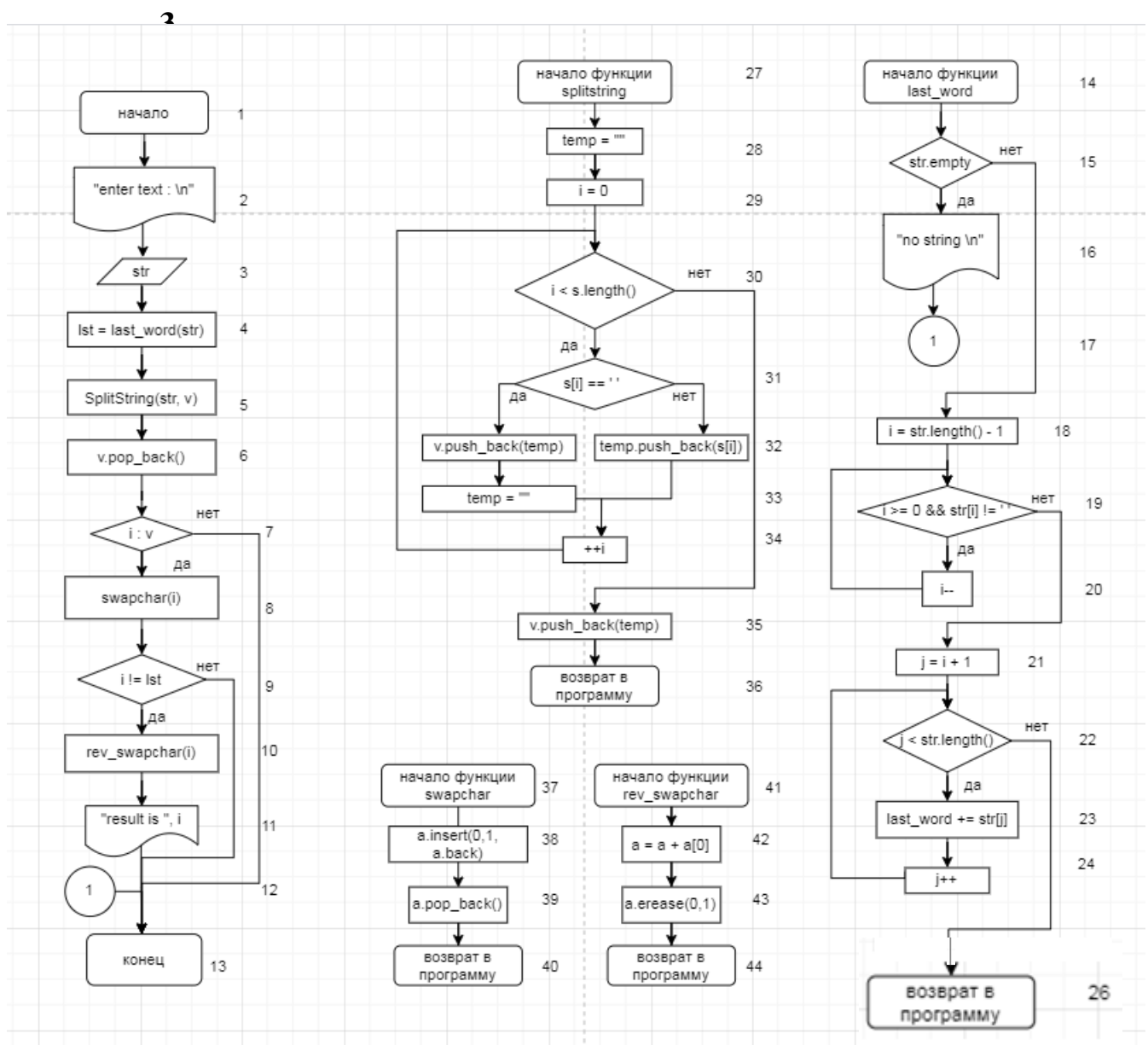
Задан текст. Напечатать все слова, отличные от последнего слова, предварительно преобразовав каждое из них по следующему правилу: перенести последнюю букву в начало слова. Использовать функции.

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

2. Блок-схема:

(внешние функции)



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)
12. Завершение внешней программы при условии..
13. Завершение программы

Внешние функции

14. Начало функции last_word
15. Проверка на Артема Радиковича Романенко
16. Вывод "string empty"
17. Завершение работы всей программы
18. int i = str.length() - 1 (14-1 = 13)
19. Проверка условия (i >= 0 && str[i] != ' ') (верно)
20. i--
.....
21. j = i + 1 (9 + 1)
22. Проверка условия (j < str.length()) (10 < 14 => верно)
23. Заполнение массива str[j] (b)
24. j++
....
25. (косяк в блок-схеме)
26. Возврат в основную программу

27. Начало внешней функции
28. Temp = ""
29. i = 0

- 30. Проверка условия (i<s.length()) (0 < 14 => верно)
- 31. Проверка условия (s[i] == ' ') (b != ' ' => неверно)
- 32. S[0] = ""
- 33. –
- 34. ++i
-
- 35. Возврат преобразованного вектора
- 36. Завершение функции splitstring, возврат к программе

- 37. Начало функции swarchar
- 38. Перенос последней буквы в начало
- 39. Удаление последнего символа
- 40. Завершение функции swarchar, возврат к программе

- 41. Начало функции rev_swarchar
- 42. Приписывание последней буквы ко всему слову
- 43. Удаление первой буквы из слова
- 44. Завершение функции rev_swarchar, возврат к программе

4. 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;
}

```

5. Результат выполнения работы программы:

Enter text : biba boba aboba abib result is : boba result is : aboba	Enter text : biba boba aboba biba result is : biba result is : boba result is : aboba	Enter text : biba boba result is : biba	Enter text : biba abib
---	---	---	---------------------------

6. 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.