

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 №1**

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.

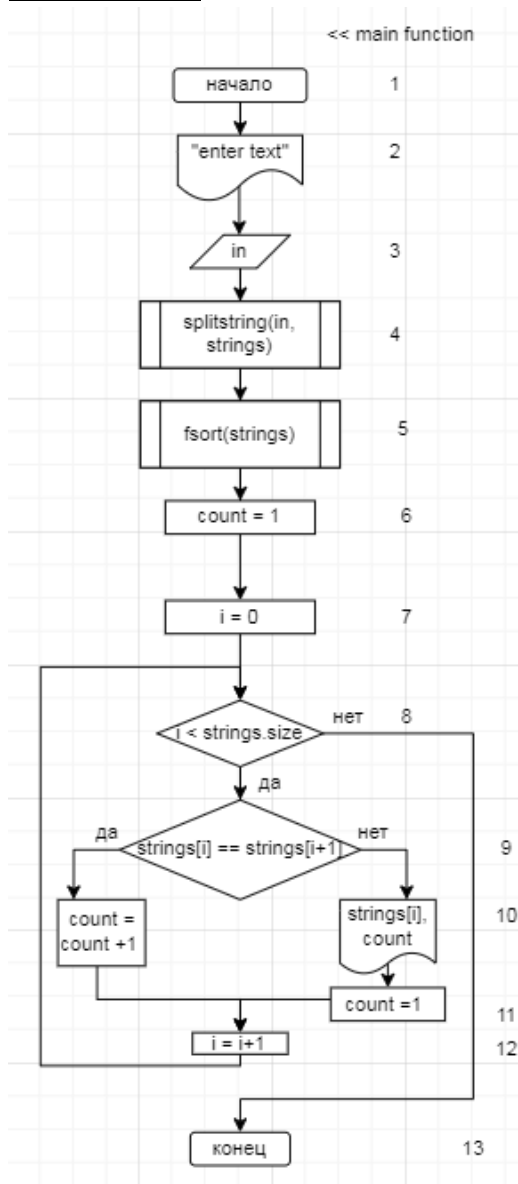
Задание

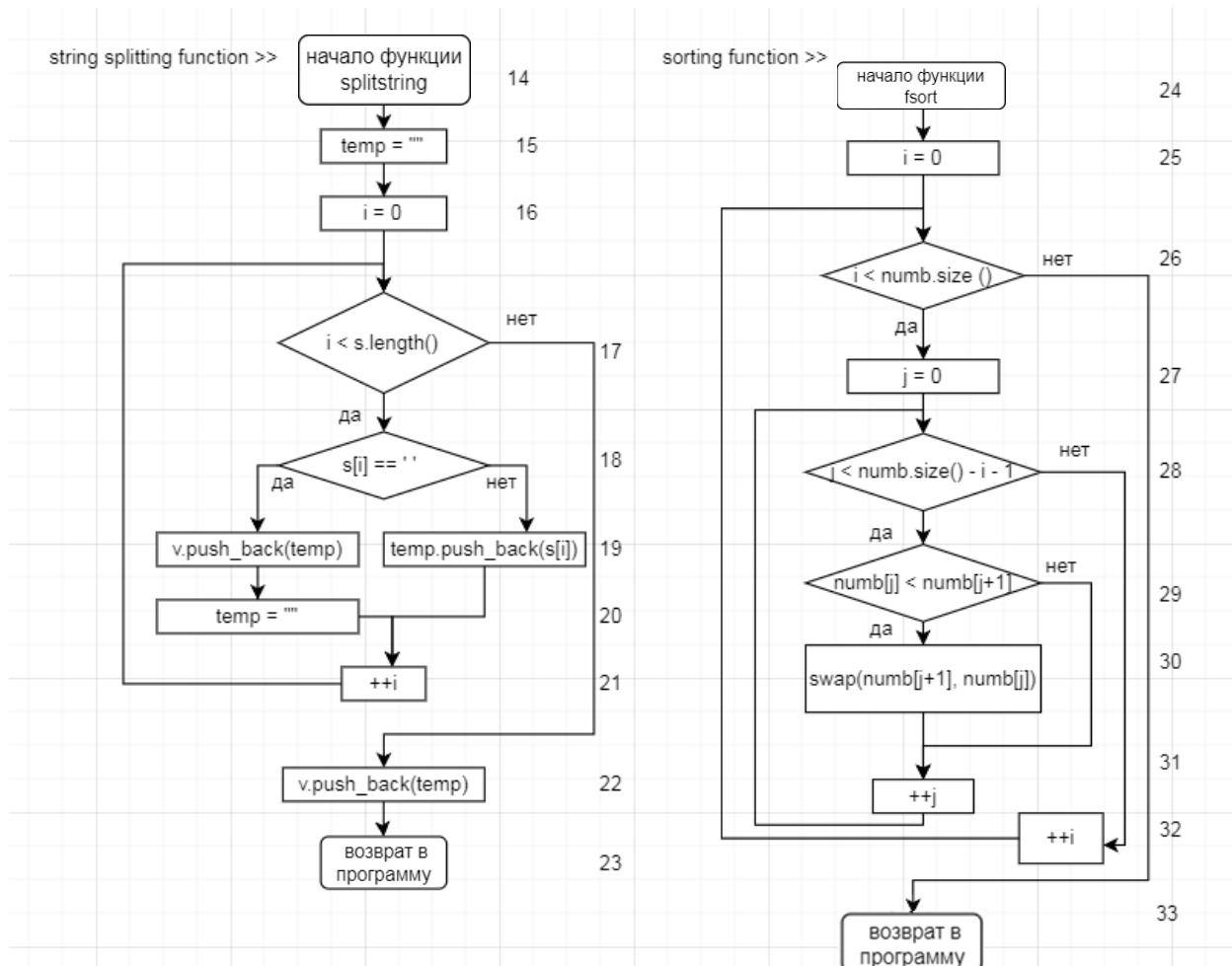
Задан текст. Определить сколько раз встречается в тексте каждое слово.

1. Таблица имён:

Рабочие переменные		
i	integer	Счетчик
j	integer	Счетчик
temp	string	Временная переменная
s	string	Временная переменная
numb	vector	Временная переменная
Исходные данные		
in	string	Текст
Результат		
strings	vector	Слово
count	integer	Счетчик

2. Блок-схема:





3. Отладочный пример:

1. Начало
2. Вывод 'enter text'
3. Ввод текста (**biba boba biba**)
4. Вызов внешней функции splitstring
5. Вызов внешней функции fsort
6. Count = 1
7. i = 0
8. Проверка условия (i < strings.size) (0 < 3 => верно)
9. Проверка условия (strings[0] = strings[1]) (неверно)
10. Вывод strings[1], count
11. count = 1
12. i++
8. Проверка условия (i < strings.size) (1 < 3 => верно)
9. Проверка условия (strings[1] = strings[2]) (верно)
10. Count ++ (**2**)
11. —
12. i++
-

13.Конец программы

Внешние функции

splitstring

14.Начало внешней функции

15.Temp = ""

16. i = 0

17. Проверка условия (i<s.length()) (0 < 14 => верно)

18. Проверка условия (s[i] == ' ') (b != ' ' => неверно)

19. S[0] = ""

20. –

21. ++i

.....

22.Возврат преобразованного вектора

23.Завершение функции splitstring, возврат к программе

fsort

24. Начало функции fsort

25. i = 0

26.Проверка условия (i<numb.size()) (0<3 => верно)

27. j = 0

28. Проверка условия (j<numb.size() – i - 1) (0 < 2 => верно)

29. Проверка условия (numb[j] < numb[j+1]) (boba < biba => неверно)

30.-

31.++j

28. проверка условия

.....

32. ++i

26. Проверка условия

.....

33.Завершение функции splitstring, возврат к программе

4. Код программы:

```
#include <iostream> //подключение библиотек
#include <string>
#include <vector>
using namespace std;

void fsort(vector <string>& numb) { //сортировка
    for (int i = 0; i < numb.size(); ++i) {
        for (int j = 0; j < numb.size() - i - 1; ++j) {
            if (numb[j] < numb[j + 1])
                swap(numb[j + 1], numb[j]);
        }
    }
}

void splitString(string s, vector<string>& v) { //преобразование текста

    string temp = ""; //временная переменная
    for (int i = 0; i < s.length(); ++i) {

        if (s[i] == ' ') { //антипробел
            v.push_back(temp);
            temp = "";
        }
        else {
            temp.push_back(s[i]);
        }
    }
    v.push_back(temp);
}

int main() //основная функция
{
    vector<string> strings; //объявление переменных
    string in;
    cout << "enter text : \n"; //ВЫВОД
    getline(cin, in); //ВВОД
    SplitString(in, strings); //вызов функций
    fsort(strings);

    int count = 1;
```

```

for (int i = 0; i < strings.size(); i++)
{
    if (strings[i] == strings[i + 1]) //проверка условия
    {
        count++;
    }
    else
    {
        cout << strings[i] << ":" << count << " "; //ВЫВОД
        count = 1;
    }
}
return 0;
}

```

5. Результат выполнения работы программы:

enter text : biba boba aboba biba boba:1 biba:2 aboba:1	enter text : biba biba biba biba biba:4	enter text : boba boba boba:2
---	---	-------------------------------------

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.