**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 №9**

*Направление:* 10.03.01*Информационная безопасность*

*Профиль:**Безопасность компьютерных систем*

Выполнил:  
студент группы УИБ-115

Клепиков Степан Даниилович

Проверил:

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

(должность, ФИО)

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

(должность, ФИО)

Москва 2021 г.

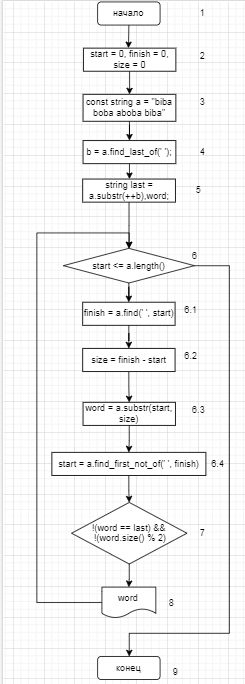
**Задание**

Задан текст. Напечатать все слова отличные от последнего слова текста, имеющие четное число букв.

1. Таблица имён:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Исходные данные | | | | | |
| start | Целочисленное | | | Значение start | |
| finish | Целочисленное | | | Значение finish | |
| a | Строка | | | Константа a | |
| b | Целочисленное | | | - | |
| Рабочие переменные | | | | | |
| size | | Целочисленное | | | Значение size |
| Результат | | | | | |
| word | | | Строка | | Выводимое слово |

1. Блок-схема:



1. Отладочный пример:
   * + 1. Начало
       2. Присвоение переменным start, finish, size значения 0
       3. Присвоение константной переменной a значения (“biba boba aboba biba”)
       4. Поиск последнего пробела (biba boba aboba\_\_biba”)
       5. Присвоение переменной last значения, являющегося предыдущим для переменной b
       6. Проверка условия (цикл while) start
2. Код программы:

#include <iostream>//подключение библиотеки ввода-вывода

#include <string> //подключение библиотеки строк

using namespace std; //подключение пространства имён std

int main()//точка входа в программу

{

int start = 0, finish= 0, size = 0; // объявление переменных типа integer

const string a = "biba boba aboba biba";// объявление константы типа integer

int b = a.find\_last\_of(' '); // находим позицию последнего пробела

string last = a.substr(++b),word; //берем последнее слово , начиная от следующей позиции, и выполняем объявления

while (start <= a.length()){

finish= a.find(' ', start); //находим первый пробел в строке, начиная от начала слова

size = finish - start; // размер

word = a.substr(start, size); // подстрока от начала слова, с данным размером

//начало следующего слова, это позиция первого символа\_непробела,

// начиная от конца последнего слова

start= a.find\_first\_not\_of(' ', finish);

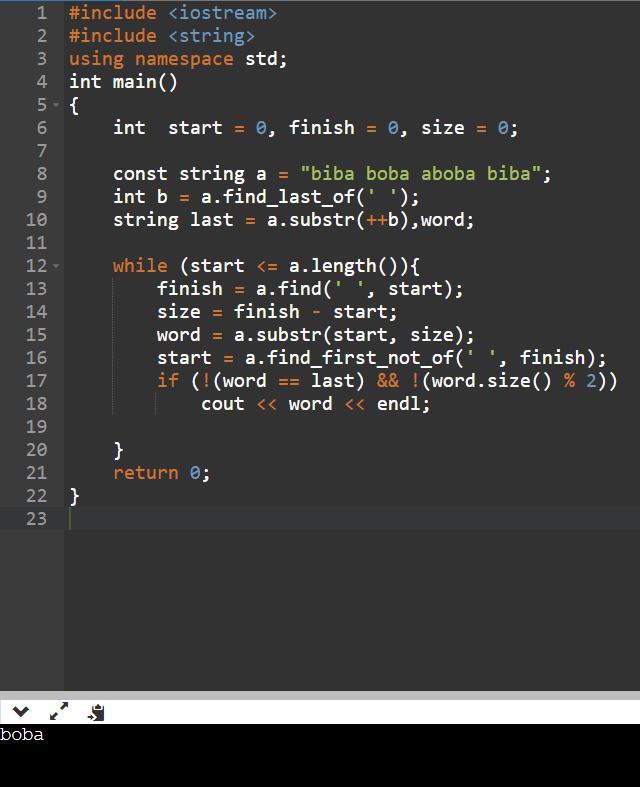
if (!(word == last) && !(word.size() % 2)) //выполняем проверку

cout << word << endl;

}

return 0; //успешное завершение программы

}

1. Результат выполнения работы программы:
2. Вывод:

В ходе выполнения работы были изучены базовые алгоритмы языка C++, библиотека <algorithm>.  
Был проделан анализ работы, на основании которого была составлена  
блок-схема.  
На контрольных примерах мы убедились, что программа работает корректно.  
Был оформлен комплект документации на программный код.