*Федеральное государственное бюджетное образовательное учреждение* *высшего образования*

|  |  |
| --- | --- |
|  | ***«Московский государственный технический университет  имени Н.Э. Баумана»***  ***(национальный исследовательский университет)*** |

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА Компьютерные Системы и сети (ИУ6)

**Отчет**

**по лабораторной работе № 4**

**Название лабораторной работы:**  Использование указателей при работе со строками С++

**Дисциплина**: Объектно-ориентированное программирование

Вариант №1 (Усложнённый)

Студент гр. ИУ6-22**\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** Н. И. Боярских

(Подпись, дата) (И.О. Фамилия)

Преподаватель  **\_\_\_\_\_\_\_\_**\_\_\_\_\_\_\_\_\_ Черноусова Татьяна Геннадьевна

(Подпись, дата) (И.О. Фамилия)

**Использование указателей при работе со строками С++**

**Задание**

Проверить правильность расположения в тексте скобок (количество открывающихся и закрывающихся скобок должно совпадать, порядок закрытия скобок должен соответствовать порядку открытия). Пользуясь ***указателями*** определить адрес символов, расположенных между 2й открывающейся и соответствующей закрывающейся скобками.

**Схема алгоритма программы ЛР№4**



*Рисунок 1 – Схема алгоритма проверки* *правильности расположения в тексте скобок, созданная в среде Microsoft Office Visio ©*

**Текст программы ЛР№4**

#include <iostream>

using namespace std;

int main(int argc, char \*\*argv) {

char str[255];

cout<<"Введите строку для проверки баланса скобок: "<<endl;

cin>>str;

// Начальная иницализация

bool find = false, done = false;

char \*open\_p = NULL, \*close\_p = NULL;

int checker = 0;

int counter = 0;

// Считываем строку либо до конца, либо пока не будет перевеса закрывающих скобок

for(int i=0; str[i] != '\0' && checker >= 0; i++) {

switch(str[i]) {

case '(':

counter++;

checker++;

// Нашли вторую открывающую скобку, сбрасываем счётчик

if(!find && counter == 2) {

open\_p = str + i;

find = true;

counter = 0;

}

break;

case ')':

// Нашли соответствующую 2-ой открывающей закрывающую скобку

if(!done && find && counter == 0) {

close\_p = str + i;

done = true;

}

if(find) counter--;

checker--;

}

}

// Проверяем сбалансированность

if(checker != 0)

cout<<"ОШИБКА: Скобки НЕ сбалансированы!"<<endl;

else {

cout<<"Скобки сбалансированы."<<endl;

// Выводим содержимое между второй парой скобок, если это возможно

if(open\_p && close\_p && open\_p < close\_p) {

cout<<"Между 2-ой парой скобок находится следующее: \"";

for(char \*i = open\_p + 1; i < close\_p; i++) cout<<\*i;

cout<<'\"'<<endl;

}

}

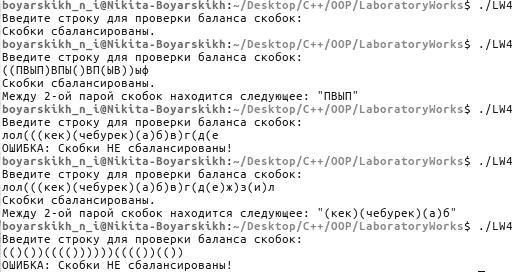
return 0;

}

**Таблица 1 – Тестирование программы ЛР№4**

|  |  |  |  |
| --- | --- | --- | --- |
| Входные данные | Выходные данные | Ожидание | Вердикт |
| ((ПВЫП)ВПЫ()ВП(ЫВ))ыф | Скобки сбалансированы  ПВЫП | Скобки сбалансированы  ПВЫП | ***Верно*** |
| лол(((кек)(чебурек)(а)б)в)г(д(е)ж)з(и)к | Скобки сбалансированы  (кек)(чебурек)(а)б | Скобки сбалансированы  (кек)(чебурек)(а)б | ***Верно*** |
|  | Скобки сбалансированы | Скобки сбалансированы | ***Верно*** |
| (()())(((())))))(((())(()) | Скобки не сбалансированы | Скобки не сбалансированы | ***Верно*** |

**Скриншоты, демонстрирующие работу программы ЛР№4**



*Рисунок 2 – скриншот, демонстрирующий работу программы ЛР№4*

**Вывод:** Разработан алгоритм проверки правильности расположения в тексте скобок, создана программа ЛР№4, реализующая данный алгоритм, успешно проведено тестирование программы.