МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ОРЛОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ

И.С. ТУРГЕНЕВА»

#### Отчет по лабораторной работе №6

по дисциплине «Императивное программирование»

|  |
| --- |
| Работу выполнил(а) студент(ка)  **Тупикин М.Н.**  группа **91-ИТ**  **г. Орел** |
| Работу проверил(а)  **Амелина О.В.** |

Отметка о зачете дата «\_\_\_\_» \_\_\_\_\_\_\_\_\_20 г.

Подпись преподавателя \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**г. Орел, 20 г.**

Лабораторная работа №6

Тема: «Частотный словарь при помощи бинарного дерева поиска»

#include <iostream>

#include <clocale>

#include <stdio.h>

#include <string.h>

#include <string>

#include <sstream>

#include <vector>

using namespace std;

struct Word

{

char word[100];

int val;

Word \*left;

Word \*right;

};

Word\* Insert(Word\* node, char \* word)

{

if (node == nullptr)

{

node = new Word;

strcpy(node->word, word);

node->val = 1; //—лово еще не встретилось тогда значение val = 1

node->left = node->right = nullptr;

return node;

}

int res = strcmp(node->word, word);

//если всретилось слово , то val +1

if(res == 0) node->val++;

else

{

if (res < 0) node->left = Insert(node->left, word);

else node->right = Insert(node->right, word);

}

return node;

}

void Print(Word \*node)

{

if (node != nullptr)

{

Print(node->left);

cout <<"Слово - ["<< node->word << "] встретилось " << node->val << " раз(а).\n";

Print(node->right);

}

}

int main()

{

setlocale(0,"");

Word\* root = nullptr;

cout << "Введите слова на английском языке через пробел (enter - конец ввода): \n";

string s;

getline(cin, s);

cout << "Ваши слова: "<<s<<endl;

//cout << "Длина предложения = "<<s.length()<<endl;

char const \*text[] = {s.c\_str()};

char \*buff = new char[s.length()+1];

stringstream ss(\*(text));

// раскладываю текст на слова и заношу в дерево

while (ss >> buff)

root = Insert(root, buff);

// вывод слов и их количества

cout<<endl;

Print(root);

delete[] text;

delete[] buff;

}

