

Міністерство освіти і науки України
Національний технічний університет України «Київський політехнічний інститут
імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра інформатики та програмної інженерії

Звіт
з лабораторної роботи № 6 з
дисципліни «Основи програмування 2.
Модульне програмування»

«Дерева»

Варіант 14

Виконав студент ІП-13, Замковий Дмитро Володимирович

(шифр, прізвище, ім'я, по батькові)

Перевірів Вечерковська Анастасія Сергіївна

(прізвище, ім'я, по батькові)

Лабораторна робота 6

Дерева

Мета заняття: вивчити особливості організації і обробки дерев

Завдання: Побудувати дерево, елементами якого є символи. Визначити максимальну глибину дерева (число гілок на найбільшому з маршрутів від кореня дерева до листків).

Хід роботи:

Програма виконана на мові програмування C++

Файл main.cpp:

```
#include <string>
#include <iostream>
#include "help_modul_6.h"

using namespace std;

/*
 * Побудувати дерево, елементами якого є символи. Визначити максимальну глибину дерева (число
 * гілок на найбільшому з маршрутів від кореня дерева до листків).
 */

int main()
{
    string str;
    cout << "Enter string to generate tree: ";
    cin >> str;

    TNode* curr = 0;

    for (int i = 0; str[i]; i++)
    {
        add(str[i], curr);
    }

    print(curr);

    char* list = 0;
    int* num = 0;

    cout << endl << endl << "Number of branches on largest of routes from root of the
tree to the leaves: " << max_len(curr) - 1 << endl;

    del_tree(curr);
}
```

Файл help_modul_6.h:

```
#pragma once

struct TNode
{
    char inf;
    TNode* left;
    TNode* right;
};

void add(char a, TNode*& curr);
void print(TNode* curr);
void del_tree(TNode* curr);
void print_rec(TNode* curr, int tab, int max);
int max_len(TNode* curr);
```

Файл help_modul_6.cpp:

```
#include <iostream>
#include "help_modul_6.h"
using namespace std;

void add(char a, TNode*& curr)
{
    if (!curr)
    {
        curr = new TNode;
        curr->inf = a;
        curr->left = 0;
        curr->right = 0;
        return;
    }
    if (curr->inf > a)
    {
        add(a, curr->left);
    }
    else if (curr->inf < a)
    {
        add(a, curr->right);
    }
}

void print(TNode* curr)
{
    cout << "Tree:";
    cout << endl << "===== " <<
endl;
    print_rec(curr, 0, 0);
    cout << "===== ";
}

void del_tree(TNode* curr)
{
    if (!curr)
    {
        return;
    }
    del_tree(curr->left);
    del_tree(curr->right);
    delete curr;
}
```

```

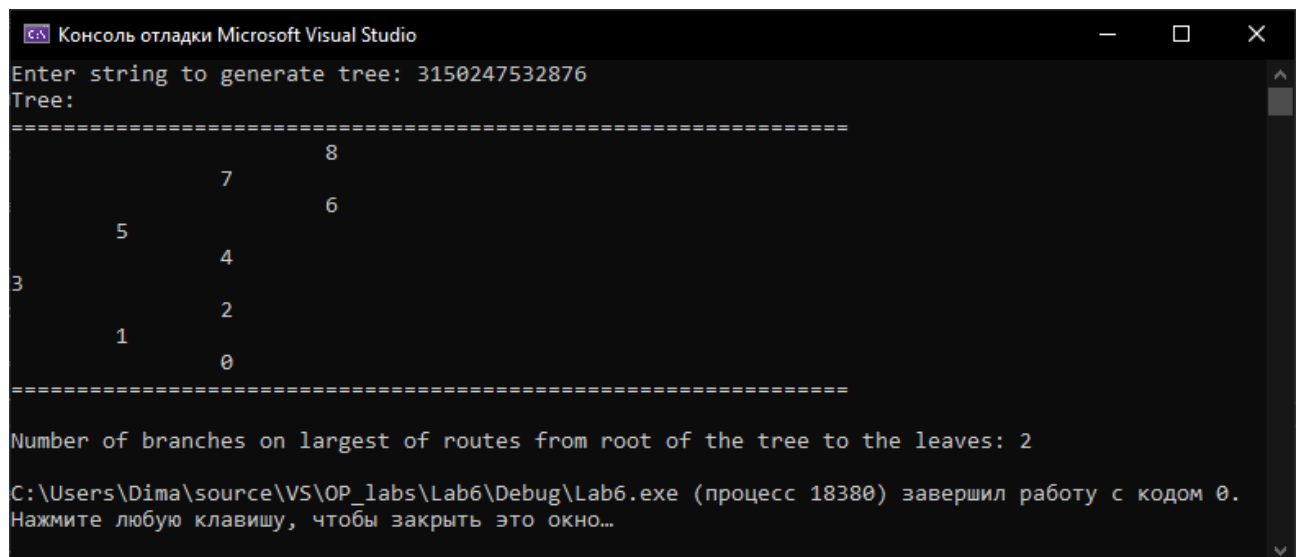
}

void print_rec(TNode* curr, int tab, int max)
{
    if (!curr)
    {
        return;
    }
    print_rec(curr->right, ++tab, ++max);
    for (int i = 1; i < tab; i++) {
        cout << '\t';
    }
    cout << curr->inf << endl;
    print_rec(curr->left, tab, ++max);
}

int max_len(TNode* curr)
{
    if (curr == 0)
        return 0;
    int left, right;
    if (curr->left != NULL) {
        left = max_len(curr->left);
    }
    else
        left = -1;
    if (curr->right != NULL) {
        right = max_len(curr->right);
    }
    else
        right = -1;
    int max = left > right ? left : right;
    return max + 1;
}

```

Результат:



```

Консоль отладки Microsoft Visual Studio
Enter string to generate tree: 3150247532876
Tree:
=====
                8
            7
        5
    3
1
    2
        4
            6
                0
=====

Number of branches on largest of routes from root of the tree to the leaves: 2

C:\Users\Dima\source\VS\OP_labs\Lab6\Debug\Lab6.exe (процесс 18380) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно...

```