

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. Метидології програмування»

«Дерева»

Варіант 23

Виконав студент ІП-13 Недельчев Євген Олександрович
(шифр, прізвище, ім'я, по батькові)

Перевірів Вєчерковська Анастасія Сергіївна
(прізвище, ім'я, по батькові)

Лабораторна робота 1

Варіант 23

23. Побудувати два бінарних дерева, елементами якого є цілі числа. Об'єднати їх, уникаючи дублювання елементів в сумарному дереві.

Код програми

C++

main.cpp

```
#include "tree.h"

int main() {
    setlocale(LC_ALL, "rus");
    srand(time(NULL));
    cout << "Введите количество n элементов дерева: ";
    int n; cin >> n;
    Branch* root1 = NULL;
    int m;
    cout << "Введите n элементов дерева: ";
    for (int i = 0; i < n; i++) {
        cin >> m;
        AddElement(root1, m);
    }
    print_tree(root1);
    Branch* root2 = NULL;
    cout << "Введите количество n элементов дерева: ";
    cin >> n;
    cout << "Введите n элементов дерева: ";
    for (int i = 0; i < n; i++) {
        cin >> m;
        AddElement(root2, m);
    }
    print_tree(root2);
    vector<int> base;
    pr_obh(root1, base);
    pr_obh(root2, base);
    Branch* root3 = NULL;
    for (const auto& i : base) {
        AddElement(root3, i);
    }
    cout << "Объединенное дерево без дублирования элементов: " << endl;
    print_tree(root3);
}
```

tree.h

```
#pragma once
#include <iostream>
#include <vector>
#include <algorithm>

using namespace std;
```

```

struct Branch {
    int number;
    Branch* left;
    Branch* right;
};

void AddElement(Branch*&, int);
void print_tree(Branch* root);
void pr_obh(Branch* branch, vector<int> &base);
bool is_element_in_vector(int elem, vector<int> base);

```

tree.cpp

```

#include "tree.h"

void AddElement(Branch*& branch, int number) {
    if (!branch)
    {
        branch = new Branch;
        branch->number = number;
        branch->left = 0;
        branch->right = 0;
        return;
    }
    else
    {
        if (number < branch->number) {
            AddElement(branch->left, number);
        }
        else if (number >= branch->number) {
            AddElement(branch->right, number);
        }
    }
}

int levels(Branch* root) {
    if (root == NULL) return 0;
    return max(levels(root->left), levels(root->right)) + 1;
}

void print_tree(Branch* root, int space) {
    if (root == NULL) return;
    print_tree(root->right, ++space);
    for (int i = 1; i < space; i++) {
        cout << '\t';
    }
    cout << root->number << endl;
    print_tree(root->left, space);
}

void print_tree(Branch* root) {
    cout << "#####\n";
    int l = levels(root);
    for (int i = 0; i < l; i++) cout << i << '\t';
    cout << endl;
    for (int i = 0; i < l; i++) cout << "|\t";
    cout << endl << endl;
    print_tree(root, 0);
    cout << "#####\n";
}

void pr_obh(Branch* branch, vector<int> &base)

```

```
{
    if (branch == NULL) return;
    if (!is_element_in_vector(branch->number, base)) {
        base.push_back(branch->number);
    }
    pr_obh(branch->left, base);
    pr_obh(branch->right, base);
}

bool is_element_in_vector(int elem, vector<int> base) {
    return (find(base.begin(), base.end(), elem) == base.end() ? false : true);
}
```

Тестування:

```
Консоль отладки Microsoft Visual Studio

Введите количество n элементов дерева: 6
Введите n элементов дерева: 2 8 6 4 0 1
#####
0      1      2      3
|      |      |      |
      8          6          4
2          1
      0
#####
Введите количество n элементов дерева: 5
Введите n элементов дерева: 1 7 4 9 3 5
#####
0      1      2      3
|      |      |      |
      9          4          3
7          1
      4          3
1
#####
Объединенное дерево без дублирования элементов:
#####
0      1      2      3      4
|      |      |      |      |
      9          7          3
8          6          4
2          1          3
      0
```

Висновки:

Я вивчив особливості організації та оброки дерев.