

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»
ІНСТИТУТ КОМП'ЮТЕРНИХ НАУК ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ



ЗВІТ

Про виконання лабораторної роботи № 5

З дисципліни

«Дискретна математика»

Студентки групи ІТ-11

Проців Роксолани Василівни

Прийняв викладач

Юринець Р.В.

Дерева

Мета роботи: Вивчення основних означень та властивостей дерев, набуття практичних навичок розв'язування завдань, що базуються на деревах, та програмної реалізації алгоритмів обходу дерев, рекурсії, бектрекінгу та Краскала.

Хід роботи:

Варіант №20

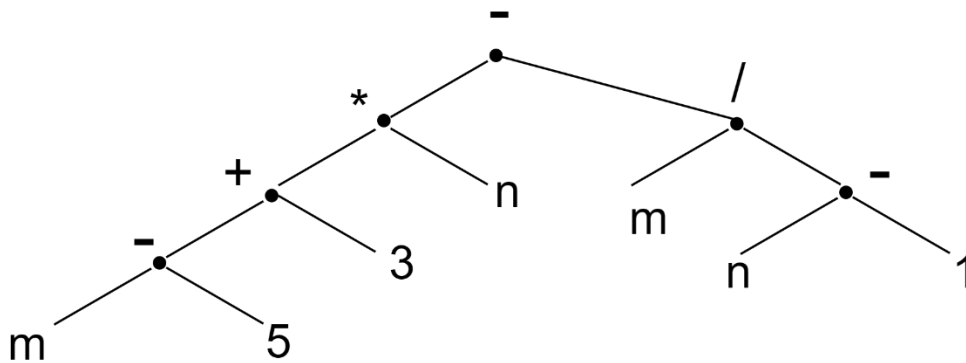
Завдання 1: Побудувати кореневе дерево, яке відповідає виразам у табл. 1. Записати ці вирази в інфіксній, префіксній та постфіксній формах.

20.	$(m-5)+3*n-m/(n-1)$
-----	---------------------

Інфіксна форма: $m-5+3*n-m/n-1$

Префіксна форма: $-+*-m53n/-n1m$

Постфіксна: $m5-3+n*n1-m/-$



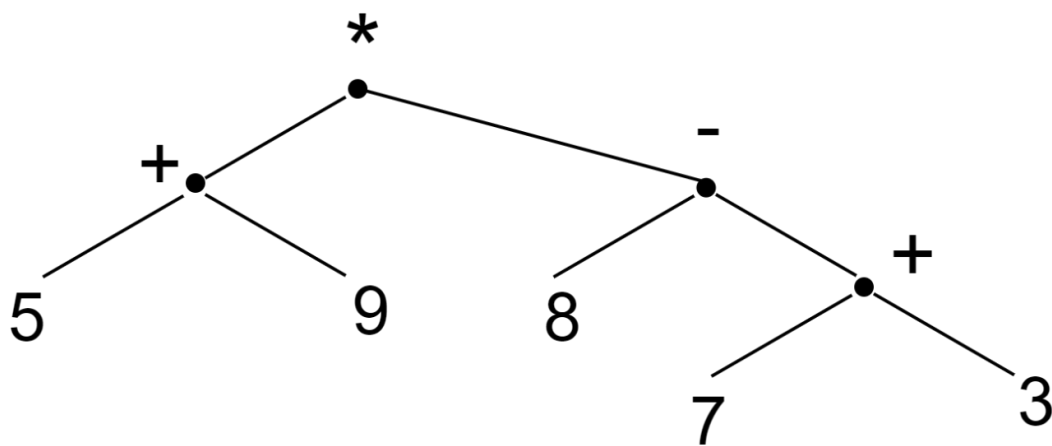
```
#include <iostream>
using namespace std;
int main()
{
    int m, n, S1, S2;
    cout << "m = "; cin >> m;
    cout << "n = "; cin >> n;
    S1 = m - 5 + 3 * n - m / n - 1;
    cout << "1. Infiksna forma = " << S1 << endl;
    return 0;
}
```

Завдання 2: Обчислити значення виразу у зворотній польській нотації і побудувати кореневі дерева

20. | 5 9 + 8 7 3 + - *

5 9+8 7 3+-*	5+9=14
14 8 7 3+-*	7+3=10
14 8 10-*	8-10= -2
14 -2 *	14*(-2) = - 28

Відповідь: -28



```
#include <iostream>
using namespace std;
int main()
{
    int m, n, S1, S2;
    S2 = (5 + 9) * (8 - (7 + 3));
    cout << "2. Result = " << S2 << endl;
    return 0;
}
```

Microsoft Visual Studio Debug Console

```
m = 8
n = 9
1. Infiksna forma = 29
2. Result = -28

C:\Users\User\source\repos\дискретна\Dyskretna5(2)\Debug\Dyskretna5(2).exe (
Press any key to close this window . . .
```

Завдання 3: Використовуючи бектрекінг, розфарбувати в n кольорів граfi із лабораторної роботи № 4.

Матриця суміжності:

	1	2	3	4	5	6
1	0	1	0	1	1	0
2	1	0	0	1	0	1
3	0	0	0	0	0	0
4	1	1	0	0	0	1
5	1	0	0	0	0	1
6	0	1	0	1	1	0

Вершини	1	2	3	4	5	6
Степені	3	3	0	3	2	3
Sort	1	2	4	6	5	3

```
#include <iostream>
using namespace std;
#define V 6
void printSolution(int color[])
{
    cout << "Solution Exists:"
         << " Following are the assigned colors"
         << "\n";
    for (int i = 0; i < V; i++)
        cout << " " << color[i] << " ";

    cout << "\n";
}
bool Safe(int v, bool graph[V][V],
         int color[], int c)
{
    for (int i = 0; i < V; i++)
        if (graph[v][i] && c == color[i])
            return false;

    return true;
}
bool ColoringUtil(bool graph[V][V], int m,
                 int color[], int v)
{
    if (v == V)
        return true;
    for (int c = 1; c <= m; c++)
```

```

    {
        if (Safe(v, graph, color, c))
        {
            color[v] = c;
            if (ColoringUtil(graph, m, color, v + 1) == true)
                return true;
            color[v] = 0;
        }
    }
    return false;
}
bool graphColoring(bool graph[V][V], int m)
{
    int color[V];
    for (int i = 0; i < V; i++)
        color[i] = 0;
    if (ColoringUtil(graph, m, color, 0) == false)
    {
        cout << "Solution does not exist";
        return false;
    }
    printSolution(color);
    return true;
}
int main()
{
    bool graph[V][V] = {
        { 0, 1, 0, 1, 1, 0 },
        { 1, 0, 1, 1, 0, 1 },
        { 1, 1, 0, 0, 0, 1 },
        { 0, 1, 0, 1, 1, 0 },
        { 1, 0, 0, 0, 0, 1 },
        { 0, 0, 0, 0, 0, 0 } };
    int m = 3;
    graphColoring(graph, m);
    return 0;
}

```

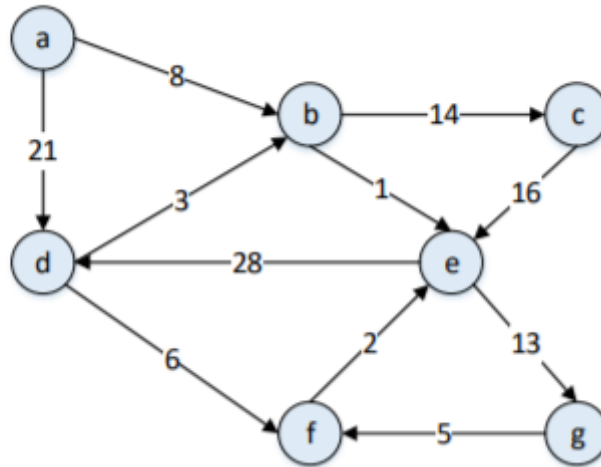
```

#include <iostream>
Microsoft Visual Studio Debug Console
Solution Exists:
1 2 3 1 2 1
C:\Users\User\source\repos\дискретна\Dysk5(3)\Debug\Dysk5(3).exe (process 4712) exited with code 0.
Press any key to close this window . . .

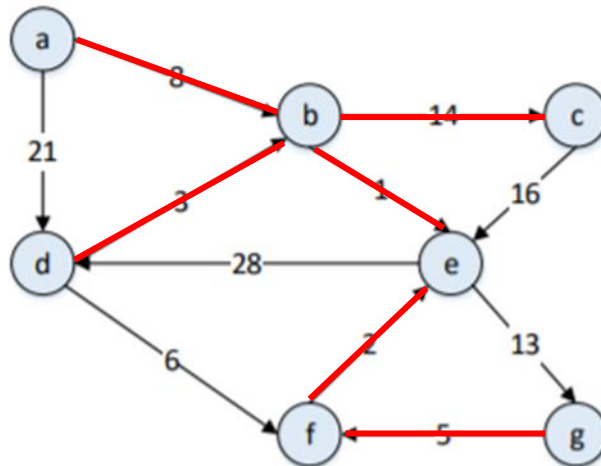
```

Завдання 4: За алгоритмом Крускала побудувати мінімальний каркас для наступних графів:

20.



20.



Висновок: я вивчила основні означень та властивостей дерев, набуття практичних навичок розв'язування завдань, що базуються на деревах, та програмної реалізації алгоритмів обходу дерев, рекурсії, бектрекінгу та Краскала.