

Міністерство освіти і науки України  
Національний університет «Львівська політехніка»  
Кафедра систем штучного інтелекту



## **Звіт**

про виконання

### **Лабораторних та практичних робіт № 6**

**з дисципліни:** «Мови та парадигми програмування»

**з розділу:** «Рік 6. Динамічні структури (Черга, Стек, Списки, Дерево).  
Алгоритми обробки дерев»

***Виконала:***

студентка групи ШІ-14 - Гуменюк Тетяна В'ячеславівна

Львів 2023

## Тема роботи:

Вивчення динамічних структур(Черга, Стек, Списки, Дерево) та алгоритму обробки дерев.

## Мета роботи:

Застосування знань про динамічні структури, обробка бінарних дерев.

## Теоретичні відомості:

1) Теоретичні відомості з переліком важливих тем:

- Тема №1: Черга
- Тема №2: Стек
- Тема №3: Списки
- Тема №4: Дерево

2) Індивідуальний план опрацювання теорії:

- Тема №1: Черга
  - Джерела Інформації:
    - <https://youtu.be/bM4iWKcaJec?si=KaGLfCrmV67DKf0W>
  - Що опрацьовано:
    - Реалізація черги
  - Статус: Ознайомлена
  - Початок опрацювання теми: 1 січня
  - Звершення опрацювання теми: 2 січня
- Тема №2: Стек
  - Джерела Інформації:
    - <https://www.youtube.com/watch?v=NyOjKd5Qruk>  
<https://www.bestprog.net/uk/2019/09/18/c-the-concept-of-stack-operations-on-the-stack-an-example-implementation-of-the-stack-as-a-dynamic-array-ua/>
  - Що опрацьовано:
    - Реалізація стеку
  - Статус: Ознайомлена
  - Початок опрацювання теми: 1 січня
  - Звершення опрацювання теми: 2 січня
- Тема №3: Списки
  - Джерела Інформації:
    - [https://www.youtube.com/watch?v=-25REjF\\_atI](https://www.youtube.com/watch?v=-25REjF_atI)  
▪ [https://www.youtube.com/watch?v=QLzu2-\\_QFoE](https://www.youtube.com/watch?v=QLzu2-_QFoE)
  - Що опрацьовано:
    - Однозв'язні та двохзв'язні списки, методи їх обробки
  - Статус: Ознайомлена
  - Початок опрацювання теми: 1 січня
  - Звершення опрацювання теми: 4 січня
- Тема №4: Дерево
  - Джерела Інформації:
    - <https://www.youtube.com/watch?v=qBFzNW0ALxQ>
  - Що опрацьовано:
    - Створення бінарного дерева, вивід посорттованих елементів
  - Статус: Ознайомлена

- Початок опрацювання теми: 1 січня
- Звершення опрацювання теми: 4 січня

## Виконання роботи:

### 1. Опрацювання завдання та вимог до програм та середовища:

- Завдання №1: Algotester lab 5 task 3
- Деталі завдання:

У вас є карта гори розміром  $N \times M$ .

Також ви знаєте координати  $\{x, y\}$ , у яких знаходиться вершина гори.

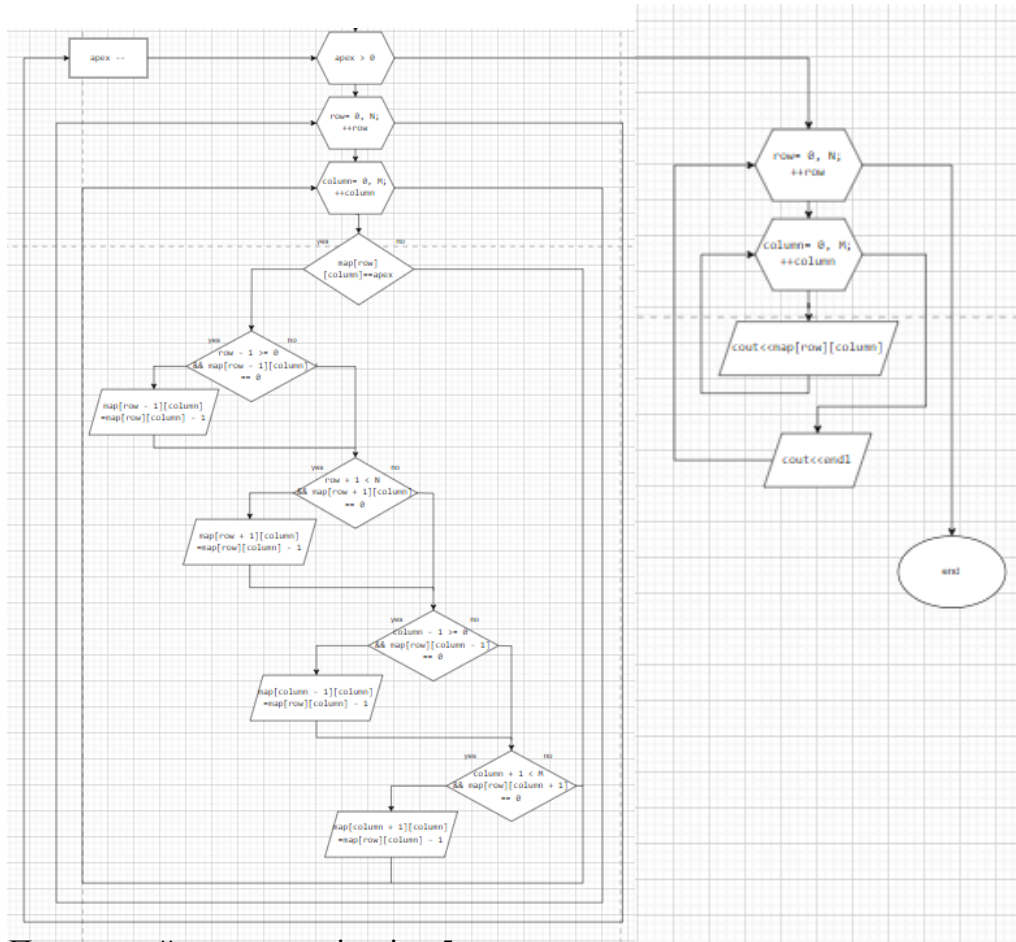
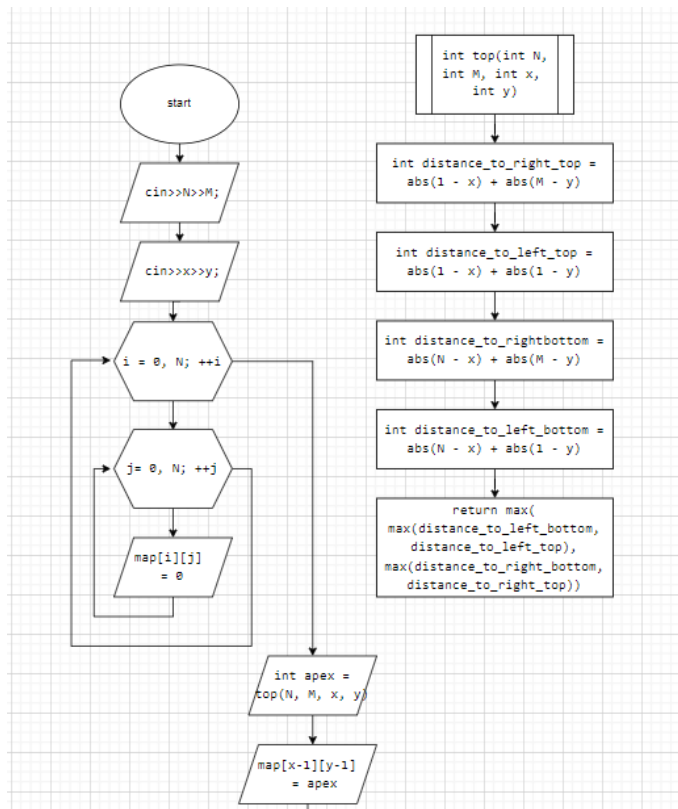
Ваше завдання - розмалювати карту таким чином, щоб найнижча точка мала число 0, а пік гори мав найбільше число.

Клітинки які мають суміжну сторону з вершиною мають висоту на один меншу, суміжні з ними і не розфарбовані мають ще на 1 меншу висоту і так далі.

### 2. Дизайн та планована оцінка часу виконання завдань:

Програма №1: Algotester lab 5 task 3

- Блок-схема



- Планований час на реалізацію: 5 годин

### 3. Код програм:

Завдання №1 Algotester lab 5 task 3

```

#include <iostream>
#include <algorithm>
#include <math.h>
using namespace std;

int top(int N, int M, int x, int y)
{
    int distance_to_right_top = abs(1 - x) + abs(M - y);
    int distance_to_left_top = abs(1 - x) + abs(1 - y);
    int distance_to_right_bottom = abs(N - x) + abs(M - y);
    int distance_to_left_bottom = abs(N - x) + abs(1 - y);
    return max( max(distance_to_left_bottom, distance_to_left_top), max(distance_to_right_bottom,
distance_to_right_top));
}

int main()
{
    int N, M, x, y;
    cin>>N>>M;
    cin>>x>>y;
    int map[N][M];
    for (int i = 0; i < N; ++i) {
        for (int j = 0; j < M; ++j)
        {
            map[i][j] = 0;
        }
    }
    int apex = top(N, M, x, y);
    map[x-1][y-1] = apex;
    while(apex > 0)
    {
        for(int row = 0; row<N; row++)
        {
            for(int column = 0; column<M; column++)
            {
                if(map[row][column]==apex)
                {
                    if (row - 1 >= 0 && map[row - 1][column] == 0) map[row - 1][column] = map[row][column] - 1;
                    if (row + 1 < N && map[row + 1][column] == 0) map[row+ 1][column] = map[row][column] - 1;
                    if (column - 1 >= 0 && map[row][column - 1] == 0) map[row][column - 1] = map[row][column] - 1;
                    if (column + 1 < M && map[row][column + 1] == 0) map[row][column + 1] = map[row][column] - 1;
                }
            }
        }
        apex --;
    }
    for(int row = 0; row<N; row++)
    {
        for(int column = 0; column<M; column++)
        {
            cout<<map[row][column]<<" ";
        }
        cout<<endl;
    }
}

```

#### 4. Результати виконання завдань, тестування та фактично затрачених час:

Завдання №1 Algotester lab 5 task 3

```
6 7
3 4
1 2 3 4 3 2 1
2 3 4 5 4 3 2
3 4 5 6 5 4 3
2 3 4 5 4 3 2
1 2 3 4 3 2 1
0 1 2 3 2 1 0
```

Час затрачених на виконання завдання: 4 години