

C++ Programming

Multidimensional Arrays

Homework 2

Mostafa S. Ibrahim

Teaching, Training and Coaching since more than a decade!

Artificial Intelligence & Computer Vision Researcher

PhD from Simon Fraser University - Canada

Bachelor / Msc from Cairo University - Egypt

Ex-(Software Engineer / ICPC World Finalist)



Problem #1: How many primes

- Read 2 integers N and M, then Read **matrix** NxM.
- Then read integer Q, for Q queries.
- For each query read 4 integers: i j r c
 - Represents a grid (submatrix): **top left** (i, j) and (r, c) for (# rows, # cols)
- For each query, print how many prime numbers in the requested grid.
 - Each query should be answered using nested loops maximum not more
 - In future: with smart precomputation it can be done without loops
 - It is called Image integral in Computer Vision field.

Problem #1: How many primes

- Input \Rightarrow Output

- 3 4
- 8 2 9 5
- **3 2** 27 6
- **7 8** 29 22
- 2
- 1 0 2 2 \Rightarrow 3 (primes 3, 2, 7 in rectangle (0, 1) (2, 1))
- 0 1 2 3 \Rightarrow 3 (primes 2, 5, 2 in rectangle (0, 1) (1, 3))

Problem #2: Find mountains

- Read integers N, M, then Read **matrix** NxM. Print all positions that are mountain. Position is mountain if its value > 8 neighbours values
- *Code smartly*
- Input
 - 3 3
 - 8 6 1
 - 3 2 9
 - 1 6 4
- Output
 - 0 0 (8 > 6, 3, 2)
 - 1 2 (9 > 1, 2, 5, 4, 6)

Problem #3: Active Robot

- Read integers N, M represents a matrix. A robot start at cell (0, 0).
- Read integer K, then K commands. Each command is 2 values
 - Direction from 1 to 4: up, right, down, left
 - Steps: a number to number steps to take in the direction. Steps [1, 1000000000]
 - If the robot hits the wall during the move, it **circulates** in the matrix.
 - For every command, print where is the robot now
- Input
 - 3 4 4 2 1 3 2 4 2 1 3
 - 2 1 means to right 1 step - 3 2 means down 2 steps
- Output
 - (0, 1) (2,1) (2, 3) (2, 3)

Problem #3: Active Robot

X			

2 1 (right 1 step) \Rightarrow
New pos (0, 1)

	X		

3 2 (down 2 steps) \Rightarrow
New pos (2, 1)

	X		

4 2 (left 2 steps) \Rightarrow
New pos (2, 3)
Circulation

			X

1 3 (up 3 steps) \Rightarrow
New pos (2, 3)
Circulation

			X

Problem #4: Flatten 3D Array

- Read 3 numbers: DEPTH, ROWS, COLS the dimensions of 3D array
- Then read integer either 1 (convert 3D to 1D) or 2 (1D to 3D)
- If input was 1, then read 3 integers d, r, c then convert to position in 1D array
- If input was 2, then read 1 integer position, then convert to 3D array position
- Try to generalize if we have e.g. 6D array
- Input \Rightarrow Outputs
 - 3 4 5 1 1 0 0 \Rightarrow 20
 - 3 4 5 2 20 \Rightarrow 1 0 0
 - 3 4 5 1 1 1 1 \Rightarrow 26
 - 3 4 5 1 2 3 2 \Rightarrow 57
 - 3 4 5 1 2 0 0 \Rightarrow 40
 - 3 4 5 2 59 \Rightarrow 2 3 4

```
int idx = 0;
for (int dep = 0; dep < 3; ++dep)
    for (int row = 0; row < 4; ++row)
        for (int col = 0; col < 5; ++col)
            cout<<idx++ << " = "
                <<dep << " " << row << " " << col<< "\n";
```

“Acquire knowledge and impart it to the people.”

“Seek knowledge from the Cradle to the Grave.”