**Algorithmic Problem Solving 2021**

**17ECSE309**

**Q-Box Assignment Set**

Student Name: Aditya Mishra

SRN: 01FE18BCS018

Branch: CSE

**Question 01**

Title: Shortest Route

Level: Easy

Concepts Tested: Implementation

**Problem Statement:**

There are n buildings in a line each having h floors, there is a bridge between any two adjacent building on every floor x, where a ≤ x ≤ b. It takes exactly 1 min to get between any two adjacent floors in a building or to get to any adjacent building from one building using the bridge. Your job is to find the minimum time it would take to get from one location (certain floor in a certain building) to another. From (b1f1) to (b2f2).

**Input Format:**

First line contains 5 integers n, h, a, b & q, q denotes the total number of queries.

Each of the next q lines contains 4 integers b1, f1, b2, f2 where (1 ≤ b1, b2 ≤ n, 1 ≤ f1, f2 ≤ h).

**Constraints:**

1 ≤ n, h ≤ 108

1 ≤ k ≤ 104

**Output Format:**

For each query print a single integer: the minimum time to get from (b1f1) to (b2f2).

**Solution:**

C++ solution: [answer](https://github.com/Adityamishra018/CPP/blob/main/Q-box/easy.cpp)

**Sample Test Cases:**

3 6 2 3 3  
1 2 1 3

1 2 2 3

//output: 1 & 2

**Test Cases:**Data can be found [here](https://github.com/Adityamishra018/CPP/tree/main/Q-box)

**Question 02**

Title: Maximum Chopping

Level: Medium

Concepts Tested: DFS

**Problem Statement:**

You are given a cycle free, connected graph containing n nodes and m edges. Aditya is interested in finding how much can you chop this graph down (by removing edges) such that all the remaining connected components satisfy N(component) % 2 == 0.

Where N(x) denotes number of nodes in x, since Aditya is busy solving other aps assignments, help him out.

**Input Format:**

The first line provides two integers n & m, n denoting the total number of nodes and m denoting the total no of edges.

The next m lines provide two integers each, a & b indicating there is an edge from a to b and b to a.

**Constraints:**

2 <= N <= 100

**Output Format:**

A single integer value denoting the maximum number of edges that can be removed.

**Solution:**

C++ solution: [answer](https://github.com/Adityamishra018/CPP/blob/main/Q-box/medium.cpp)

**Sample Test Cases:**

4 3

1 2

1 3

3 4

//output: 1, at max the edge 1-3 can be removed.

**Test Cases:**

Data can be found [here](https://github.com/Adityamishra018/CPP/tree/main/Q-box)