Name: Roll Number:

ESO207: Data Structures and Algorithms

Programming Assignment 3

Due Date: 27th October 11:59pm, 2025

Total Number of Pages: 5 Total Points 500

Note :

- The questions have to be answered through a contest in Hackerrank. The contest has 3 challenges, each corresponding to a question. You have to submit your code through the contest.
- Contest Link: `http://www.hackerrank.com/programming-assignment-3-eso-207-2025-26-i`
- Additionally you must upload your solutions on hello.iitk.ac.in as well. You need to upload 3 files corresponding to the 3 programs.
- Your codes will be checked for possible plagiarism of any sorts. If we find such cases, then we will possibly award an F grade.
- Allowed Languages for challenge code submission : C, C++
- Allowed libraries : stdio.h for C and iostream for C++

Question 1. (150 points) **The Clockwork Castle**

King Horace has a peculiar castle with N rooms, connected by $N-1$ corridors in such a way that it is possible to travel between any two rooms. Each room contains a magical clock with numbers 1 through 12, but the clocks are faulty: the hand only points to an integer and never moves on its own.

A knight is tasked with synchronizing all the clocks to point at 12. The knight can start in any room. Whenever he **enters a room**, he advances its clock by 1 (from 12 it wraps around to 1). The knight can move freely along corridors, but once he enters a corridor, he must traverse it to the room at the other end—turning back midway is not allowed. **The starting room is not advanced initially**, but subsequent visits will advance its clock as usual.

Determine how many rooms the knight can start in so that it is **possible** to set all clocks to 12 after some sequence of moves.

- **Input**
  - The first line contains an integer N — the number of rooms.
  - The second line contains N integers $c_1, c_2, \ldots, c_N$ ($1 \le c_i \le 12$) — the initial clock values for each room.
  - Each of the next $N-1$ lines contains two integers a and b ($1 \le a, b \le N$) — representing a corridor connecting rooms a and b.
- **Output**

  Print a single integer — the number of rooms from which the knight can start and successfully synchronize all clocks to 12.
- **Constraints**
  - $2 \le N \le 10^5$
  - $1 \le c_i \le 12$
  - The corridors form a **tree** (connected graph with $N-1$ edges)
- **Sample Input**

```
4
11 10 11 11
1 2
2 3
2 4
```

- **Sample Output**

```
1
```

- **Explanation**

  If the knight starts in **room 2**, he can traverse the rooms in a sequence like $2 \to 1 \to 2 \to 3 \to 2 \to 4$. Each time he enters a room, the clock advances by 1. After carefully planning his path, all clocks point to 12. No other starting room allows synchronization.

Question 2. (250 points) **Problem 2 - The Whispering Paths**

After synchronizing all the clocks, King Horace discovers a new mystery within the castle. Hidden within the corridors lies an ancient mechanism that hums softly — the "Whispering Paths". These paths connect the rooms in a peculiar way, forming a network of N rooms, linked by $N - 1$ corridors. Room 1 is the entrance hall — the root of the castle.

The court scholars present M magical patterns, each a collection of certain rooms. For each pattern, they ask whether there exists a path that begins at the entrance hall (room 1) and leads to some room u, such that:

Every room in the given set either lies on this path, or is directly adjacent to a room on this path.

In other words, for every magical pattern, you have to answer the following question – Is there any room u, such that every room in the magical pattern, either lies on the path from 1 to u OR lies at a distance 1 from some node on this path?

- **Input**
  - The first line contains two integers N and M — the number of rooms and the number of patterns.
  - The next $N - 1$ lines each contain two integers a and b ($1 \leq a, b \leq N$, $a \neq b$), describing a corridor connecting rooms a and b. It is guaranteed that the corridors form a tree.
  - The next M lines describe the magical patterns. For the i-th pattern:
    * The line begins with an integer k_i ($1 \leq k_i \leq N$) — the number of rooms in this pattern.
    * Then follow k_i distinct integers $v_{i,1}, v_{i,2}, \ldots, v_{i,k_i}$ ($1 \leq v_{i,j} \leq N$) — the rooms included in this pattern.
- **Output**

  Print M lines. For each pattern, print: "YES" if such a path exists, or "NO" otherwise.
- **Constraints**
  - $2 \leq N \leq 10^5$
  - $1 \leq M \leq 10^5$
  - $\sum_{i=1}^{M} k_i \leq 2 \times 10^5$
- **Sample Input**

```
10 6
1 2
1 3
1 4
2 5
2 6
3 7
7 8
7 9
9 10
4 3 8 9 10
3 2 4 6
3 2 1 5
3 4 8 2
2 6 10
3 5 4 7
```

- **Sample Output**

```
YES
YES
YES
YES
NO
NO
```

- **Explanation**

  Consider the first pattern $[3, 8, 9, 10]$. The path from the entrance (1) to room 10 covers rooms $1, 3, 7, 9, 10$. Rooms $3, 9, 10$ lie on this path, and room 8 is directly connected to 7, which is on the path — so the answer is `YES`.

  Similarly, the second pattern $[2, 4, 6]$ works for the path from 1 to 2, while the fifth and sixth patterns have no such valid path.

Question 3. (100 points) **Problem 3 - The Sinking Isles**

After solving the mystery of the Whispering Paths, King Horace journeys to the castle's outer grounds — a vast garden of floating **islands**. The garden is represented as an $m \times n$ grid, where each tile is either **land** or **water**. The islands drift over years, forming strange shapes.

A tile with value 1 represents land, and 0 represents water. Two land tiles are considered connected if they share an edge (up, down, left, or right). An **island** is a maximal group of connected land tiles.

An old monk tells King Horace that this garden is **cursed** and whoever steps on the island dies. However, the garden is cursed only if there is exactly one island. Otherwise, the curse will have no effect, i.e., the garden is **safe**. To remove the curse, King calls his army of engineers.
Each day, the King's engineers may sink one piece of land — turning a single land tile (1) into water (0). The King wishes to know: how many days (minimum) are needed before the the garden is **safe** — that is, it no longer contains exactly one island.

- **Input**
  - The first line contains two integers m and n — the number of rows and columns in the garden.
  - The next m lines each contain n integers (either 0 or 1), describing the grid.
- **Output**
  Print a single integer — the **minimum number of days** required to make the garden safe.
- **Explanation**
  The garden is safe if:
  - it has more than one island, or
  - it has no land at all.
- **Constraints**
  - $1 \leq m, n \leq 2500$
  - Each cell is either 0 or 1
- **Sample Input 1**
```
3 4
0 1 1 0
0 1 1 0
0 0 0 0
```
- **Sample Output 1**
```
2
```
- **Explanation**
  Initially, the garden forms a single island. After sinking two land tiles (for example, at positions $(1, 2)$ and $(0, 2)$), the garden splits into two islands. Therefore, at least 2 days are required.
- **Sample Input 2**
```
1 2
1 1
```
- **Sample Output 2**
```
2
```
- **Explanation**
  Turning one land tile into water still leaves one island. Only after both are sunk does the garden become fully water — hence safe. Thus, the answer is 2.