
CS29003 ALGORITHMS LABORATORY

ASSIGNMENT 4

Date: 9th Sep, 2021

Important Instructions

1. **Input:** To be taken from command line
 2. **Format fo files to be submitted to Moodle and HackerRank:** ROLLNO_A4.c/.cpp
 3. Take inputs from command line in the specified format
 4. You are to **stick to the file input output formats strictly** as per the instructions.
 5. Submission through **.zip files are not allowed.**
 6. Write your name and roll number at the beginning of your program.
 7. Do not use any global variable unless you are explicitly instructed so.
 8. Use proper indentation in your code.
 9. Please follow all the guidelines. Failing to do so will cause you to lose marks.
 10. **There will be part marking.**
-

Graph Traversal

Problem Statement

Suppose you are in the Pokemon multiverse, and also have a Pokemon (maybe Mewtwo) with which you can beat any gym leader of any universe. So basically you can rule any of the Pokemon universe easily, but too early to celebrate, there are some rules that every player need to obey, so you are also bound to follow those–

1. To become the champion of a particular Pokemon universe you need to beat all the gym leaders in it.
2. You can't challenge any random gym leader and start fighting, you need to first defeat all the players that he/she defeated previously and it is recursive.

Good things are–

1. As you have the most powerful Pokemon, you will always be a winner irrespective of the opponent.
2. You will be provided with a full list of previous fights of a particular universe that you want to attack, clearly mentioning the winner and loser for each of the fights (there will be no tie).
3. Also for simplicity, between any two opponents there will be at most two fights and the outcomes will also be different. So if A and B are two players, there could be possibilities-
 - ▷ A and B have not faced each other yet.
 - ▷ Single match has been played among A & B.

▷ Two matches have been played and A & B won one each.

Given all these rules and fighting history of any of the Pokemon universe, you need to decide the following points-

1. Is it possible to be the champion of this particular Pokemon universe?
2. If yes, provide the sequence of players you need to face to become the champion (remember the rules). There may be multiple possible sequences, reporting any one of them would suffice.
3. Suppose you have found an universe that you can win (after analysing its previous match history). But the problem is that you need a significant amount of time to reach there from your universe. So it is possible that before you reach there, the players will play some more matches among them and it may close the door for you to win the universe. So now the question is what is the maximum number of matches you can afford them to play without destroying your dream of winning the universe? You also need to print the match details.

Input constraints:

$$1 \leq \text{Num of players} \leq 10^5 \quad (1)$$

$$1 \leq \text{Num of matches} \leq 10^5 \quad (2)$$

Note: Solve it with BFS and queue.

Templates: Use the templates below to implement a queue. **Use of STL is strictly prohibited.**

```
typedef struct {
    int *Arr ;
    int queue_size, start_id, end_id;
}QUEUE;
void init(QUEUE *qP, int size)
int isempty(QUEUE qP)
void enqueue(QUEUE *qP, int data)
int dequeue(QUEUE *qP)
void display_queue(QUEUE *qP)
```

Test Case 1

Input

FILE: *input-part1.txt* _____

2 1
0 1

The first line represents total number of players in a particular Pokemon universe, say n (players are represented by indices 0 to $n-1$), number of match history say m , separated by space. Following these are the m lines representing two player ids between $[0, n-1]$ separated by a space denoting the first player (id) beaten by the second player (id).

Output

FILE: *output-part1.txt* _____

0,1
0

Explanation: There are a total of 2 players to take on. To play with 1 you should have to defeat player 0 first. So the correct course order is [0,1]. And maximum number of matches that can still be played without changing the outcome for you is 0, because there is only one match possible with 0 beating 1, but that will create a cycle and it will become impossible for you to win the universe.

Test Case 2

Input

FILE: *input-part2.txt* _____

4 4
1 0
2 0
3 1
3 2

Output

FILE: *output-part2.txt* _____

3,1,2,0
2
3 0
1 2

Explanation: There are a total of 4 players to take on. To play with player 1 or 2 you need to beat player 3 first, and to play with player 0 you need to beat both player 1 and 2. So 3,1,2,0 or 3,2,1,0 are the correct sequences. And maximum number of matches that can still be played without changing the outcome for you is 2, 0 beating 3 and 1 beating 2 or 2 beating 1 will still leave the passage for you to win the universe. But anymore matches will block the path for you.

Test Case 3

Input

FILE: *input-part3.txt* _____

2 2

0 1

1 0

Output

FILE: *Output-part3.txt* _____

unconquerable

Explanation: There are a total of 2 players to take on. To take play with 1 you should have to defeat 0 first, and to play with 0 you should have to beat 1 first. So it is impossible.

To test your own code, take the input from command line similar to formatting specified in the test cases. HackerRank will have some test cases so you can directly submit the code in Hackerrank to run on those test cases. HackerRank test cases are only provided to check the correctness of the code partially. You have to create additional test cases to handle corner cases (if any) and check the efficiency of the code.