Bob and Ben

by nikasvanidze

Problem | Submissions | Leaderboard | Discussions | Editorial

Bob and Ben are playing a game with forests! The game's rules are as follows:

- The game starts with a forest of n trees.
- Bob always moves first and they take alternating turns. The first player with no available move loses the game.
- During each move, the player removes one node. If the node is *not a leaf*, then the whole tree vanishes; otherwise, the rest of the tree remains in the forest. We define a leaf to be a node with exactly 1 connected edge.
- Both players play optimally, meaning they will not make a move that causes them to lose the game if some better, winning move exists.

We define each tree i in the n-tree forest as follows:

- Tree i is defined by two integers, m_i (the number of nodes in the tree) and k_i (a constant).
- Its nodes are numbered sequentially from 1 to m_i.
- Its edges are numbered sequentially from 1 to $m_i - 1$, and each edge j connects node $j + 1$ to node $\lfloor max(1, \frac{j}{k_i}) \rfloor$.

Given the values of m_i and k_i for each tree in the forest, can you determine who will win the game?

Input Format

The first line contains an integer, g, denoting the number of games. The subsequent lines describe each game in the following format:

1. The first line contains an integer, n, denoting the number of trees in the forest.
2. Each of the n subsequent lines contains two space-separated integers describing the respective values of m_i and k_i for tree i.

Constraints

- $1 \leq g \leq 100$
- $1 \leq n \leq 10^6$
- $1 \leq m_i \leq 10^9$
- $2 \leq k_i \leq 100$
- The sum of n over all games is at most 10^6.

Subtasks

For 50% of the maximum score:

- The sum of n over all games is at most 10^3.
- $1 \leq m_i \leq 10^3$

For 25% of the maximum score:

- $1 \leq n, m_i, g \leq 10$

Output Format

For each game, print the name of the winner on a new line (i.e., `BOB` or `BEN`).

Sample Input

```
2
2
1 2
1 3
1
3 2
```

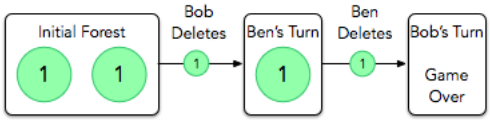
Sample Output

```
BEN
BOB
```

Explanation

Bob and Ben play the following two games:

1. The forest consists of $n = 2$ trees containing one node each, and each tree has no edges as m_1 and m_2 are both 1 (so both trees have $1 - 1 = 0$ edges). The sequence of moves is as follows:

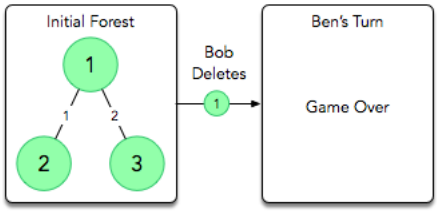


   We then print the name of the winner, `BEN`, on a new line.

2. The forest consists of $n = 1$ tree containing three nodes. We find the $m_1 - 1 = 2$ edges like so:

   - Edge $j = 1$ connects node $j + 1 = 2$ to node $floor(max(1, \frac{j}{k_1})) = floor(max(1, \frac{1}{2})) = 1$.
   - Edge $j = 2$ connects node $j + 1 = 3$ to node $floor(max(1, \frac{j}{k_2})) = floor(max(1, \frac{2}{2})) = 1$.

   The game then plays out as follows:



   We then print the name of the winner, `BOB`, on a new line.

Submissions:173

Max Score:50
Difficulty: Medium

Rate This Challenge:

More

Current Buffer (saved locally, editable)

Java 7

```
1 import java.io.*;
2 import java.util.*;
3 import java.text.*;
```

```java
import java.math.*;
import java.util.regex.*;

public class Solution {

    public static void main(String[] args) {
        /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution. */
    }
}
```

Line: 1 Col: 1

Upload Code as File | Test against custom input | Run Code | Submit Code