Casino Floor | Advanced Division

Problem F: The Cardinal Quest

Base Program Constraints: *1s, 256 MB*

Clear Reward: *+1 Point*

Stanley *might* be the reason why the PHS Computer Team is now in crippling debt — but hey, it's not every day you get to visit the Cardinal Casino. Why not enjoy the various games? As long as he doesn't gamble away *too* much money, he'll be fine, and it'll make him look like a convincing customer. At least, that's what he's telling himself...

Besides the usual card games and slot machines, the Cardinal Casino also has a selection of arcade games to bet money on! One particular game catches Stanley's eye - **The Cardinal Quest**. In The Cardinal Quest, your character starts with 1 **power token**, and you try to build up as much power as possible during your journey to the *Calculus Castle*.

Along the road to the castle, there are n **junctions**. At each junction, the road branches off into a number of different **paths**, where a_i is the number of paths at the i-th junction. At the end of the junction, all a_i paths rejoin back into the main road. However, each **path** is enchanted with a **magical spell** that applies a mathematical operation to your current number of power tokens:

- `ADD` paths **increase** your power tokens by a fixed positive integer.
- `SUB` paths **decrease** your power tokens by a fixed positive integer.
- `MULT` paths **multiply** your power tokens by a fixed **nonzero** integer.

Once you enter a path, you **cannot** back out of it. In other words, after the path's spell is applied to your power tokens, you must end up on the other side of the junction. However, you **can** move through a path both ways (go backwards through a junction).

Upon reaching the end of the n-th junction, Stanley can choose to leave the road and enter the Calculus Castle. To avoid losing his money, Stanley wants to be as prepared for the final boss of the castle as possible. For some roads, it might be possible for Stanley to gain **infinitely many** power tokens. If not, help him calculate the **maximum** number of power tokens he can have when he enters the castle.

Input

Each test contains multiple test cases. The first line of input contains the number of test cases t $(1 \le t \le 10)$.

The first line of each test case contains an integer n $(1 \le n \le 10^3)$, the number of junctions. Then, for each of the n junctions:

- The first line contains an integer a_i $(1 \le 1 \le 10^3)$, the number of paths at the i-th junction.
- The next a_i lines of input contain an operation (`ADD`, `SUB`, or `MULT`) followed by an integer x $(-100 \le x \le 100)$, describing the magical spell associated with that path of the junction.

It is guaranteed that the sum of n across all test cases does not exceed 10^3.

It is guaranteed that all magical spells follow the constraints set in the problem description (see above).

Output

For each test case, output the **maximum** number of power tokens Stanley can enter the Calculus Castle with.

If Stanley can gain infinitely many power tokens, output `INFINITY` instead. (Formally, for all integers x, if there exists a sequence of moves such that Stanley can enter the castle with more than x power tokens, output `INFINITY`.)

Sample Test Cases

Sample 1 - Input

```
3
1
1
ADD 1
1
2
SUB 3
SUB 2
2
2
SUB 3
SUB 2
1
MLT -1
```

Sample 1 - Output

```
INFINITY
-1
INFINITY
```

Notes

In the first test case, there is one junction with one path, which adds 1 power token each time you cross it. Stanley can keep moving back and forth through this path to gain as many power tokens as he wants.

In the second test case, there is one junction with two paths, where Stanley can choose between losing 2 power tokens or losing 3. Stanley's power token count is maximized if he uses the `SUB 2` path once. Note that going back through this junction is suboptimal.

The third test case is almost the same as the second test case, but contains a second junction with a `MLT -1` path. Now, Stanley can move back and forth through the gates in the first junction, then pass through the second junction to end up with as many power tokens as he wants.