7918 Bob vs ATM

Due to shortage of ATMs, a bank decided to convert some arcade gaming machines into ATMs. Unfortunately, due to a bug, the converted machine will only dispense cash if the withdrawer can win the game. The game is played over a valid bracket sequence. Formally a valid bracket sequence is defined as:

1. An empty string is valid
2. If A is a valid bracket sequence, then `(`A`)` is also valid
3. If A and B are valid, AB is also valid

In each move a player can erase such a non-empty valid sub-string of the remaining brackets such that they cannot be formed by concatenating 2 valid sequences. E.g. a player can erase `(())` or `(()())` but not `()()` or `(())()`. The ATM always plays first. The player that is not able to make a move will lose the game. Assume that both the players play optimally.

Bob is trying to withdraw money and is frustrated as he keeps losing against the machine. He suspects the machine is rigged and always wins the game. So, he has asked you to help him out. Given a valid bracket sequence, can you determine who wins the game?

Input

The first line of the input contains an integer T denoting the number of test cases. The description of T test cases follows.

Each of the test case consists of a single line containing the string S — the initial bracket sequence.

Output

For each test case, output a single line containing the winner — either '`ATM`' or '`Bob`' without quotes.

Constraints:

- $1 \leq T \leq 10^5$
- $2 \leq length\ of\ S \leq 10^5$
- The sum of all lengths of S will be not more than 10^6
- S will be a valid bracket sequence

Explanation:

Example case 1. The ATM will erase the entire sequence and win the game

Example case 2. The ATM will again erase the entire sequence and win the game

Example case 3. The ATM can erase either the first or the second `()`, Bob will erase the other one and win

Sample Input

```
5
()
(()()(()))
()()
(())()
(()()(()))((()))
```

Sample Output

```
ATM
ATM
Bob
ATM
Bob
```