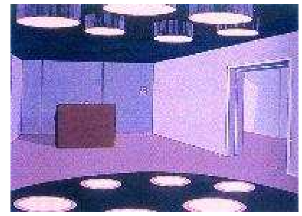
Problem I- Quantum teleporters



It's the future, and all public transit has been replaced with a network of Quantum Teleporters^{tm}! Unfortunately they have a peculiar feature- each teleporter can be in one of two states (let's call them A and B), and you won't know what the states are until you take the entire trip (quantum!).

The states affect the time it takes to be teleported from one place to another. The travel time between two connected teleporters depends on the states they ended up in: AA, AB, BA or BB. Four possible states mean four possible travel times. You need to get to a programming contest and since you absolutely cannot be late, you need to find the path with the smallest guaranteed total time.

Input Specification:

The first line of the input contains a nonnegative integer $T \leq 20$, the number of test cases to follow. Each test case starts with a line containing two integers: N, the number of teleporters, themselves numbered from 0 to $N-1$ ($2 \leq N \leq 100$), and M, the number of teleporter connections. M lines follow, each containing six integers: U, V, $A_U A_V$, $A_U B_V$, $B_U A_V$, $B_U B_V$; which describe a two-way connection between teleporters U and V, with travel times that depend on the states of U and V. Travel times are integers in the range 0 to 10 inclusive.

In the data, there will be at most one connection between each pair of teleporters and no teleporter will be connected to itself. You start at teleporter 0 and the programming contest is at teleporter 1. There will always be a route from 0 to 1.

Output Specification:

For each test case, output the smallest time you can guarantee the trip will take. Output one case per line.

Sample Input:

```
1
3
2
0 2 1 10 1 10
2 1 9 9 1 1
```

Sample Output:

```
11
```