UVa Online Judge

10841 Lift Hopping in the Real World

"I think problem descriptions should withstand the reality check, otherwise they are misleading."
`little joey`

A skyscraper has no more than 100 floors, numbered from 0 to 99. It has n $(1 \leq n \leq 50)$ elevators which travel up and down at (possibly) different speeds. For each i in $\{1, 2, \ldots n\}$, elevator number i takes T_i $(1 \leq T_i \leq 100)$ seconds to travel between any two adjacent floors (going up or down). Elevators do not necessarily stop on every floor. What's worse, not every floor is necessarily accessible by an elevator.

You are on floor 0 and would like to get to floor k as quickly as possible. When you first arrive to floor 0, each of the n elevators could be parked on any of the floors accessible by that elevator. You can take any of the elevators that stop on floor 0 and you can switch elevators on any floor, f, if both of them stop on floor f. Assume that it takes 5 seconds to exit one elevator and push another elevator's button. You are forbiden from using the staircase. No one else is in the building, so you don't have to stop if you don't want to, and no one else will request an elevator. Each elevator has its own button, and the system has built-in security to prevent abuse by evil teenagers – once you press the button calling for some elevator E, all the buttons calling for other elevators will be disabled until elevator E arrives at your floor. In other words, you can only call for one elevator at a time.

Calculate the number of seconds required to get from floor 0 to floor k in the worst case if you do not know the initial positions of the elevators. Passing floor k while inside an elevator that does not stop there does not count as "getting to floor k". See the examples below.

Input

The input will consist of a number of test cases. Each test case will begin with two numbers, n and k, on a line. The next line will contain the numbers $T_1, T_2, \ldots, T_n$. The next n lines will contain sorted lists of integers – the first line will list the floors visited by elevator number 1, the next one – those visited by elevator number 2, etc.

Output

For each test case, output one number on a line — the number of seconds required to get to floor k from floor 0 in the worst case. If floor k is inaccessible from floor 0, print '`IMPOSSIBLE`' instead.

Discussion:

In the first example, the worst case is when elevator 1 is on floor 99 and will take 990 seconds to reach floor 0. You will then take it to floor 13 in 130 seconds, spend 5 second exiting and calling elevator 2, which is on floor 30. It will take 170 seconds to reach you and 170 seconds to take you to floor 30. The total is 1295 seconds.

In the second example, the only sensible way to get to floor 30 is to ride the first elevator all the way. Switching to elevator 2 and then back will only make things worse because elevator 1 must take at least 300 seconds to reach floor 30. In the worst case, elevator 1 starts on floor 30.

In example 3, the security system will not allow you to push all 3 buttons and see which elevator comes first. Instead, you should push elevator 2's button. Unfortunately, it is on floor 30 and takes 1500 seconds to get to you. You board it and it takes you to floor 20 in 1000 seconds. You spend 5 seconds pushing the button for elevator 3, which is on floor 50. It takes 3000 seconds before it reaches floor 20 and 3000 more seconds to take you to floor 50. The total is 8505 seconds. If you take elevator 1 first, your worst case time will be 8710 seconds.

In the last example, the one elevator does not stop at floor 1.

Sample Input

```
2 30
10 5
0 1 3 5 7 9 11 13 15 20 99
4 13 15 19 20 25 30
2 30
10 1
0 5 10 12 14 20 25 30
2 4 6 8 10 12 14 22 25 28 29
3 50
10 50 100
0 10 30 40
0 20 30
0 20 50
1 1
2
0 2 4 6 8 10
```

Sample Output

```
1295
600
8505
IMPOSSIBLE
```