Problem TUBEMARATHON: Tube Marathon

James is on vacation and bored. He thus decided to participate in the famous London tube marathon. Rules of this marathon are simple: given a tube map participants are required to visit all tube stations in lexicographical order as fast as possible. Next to several different tube lines participants are also allowed to use other means of public transport. However, James is reluctant to use any cycle hire service and thus only uses buses as alternative to the tube service. He thinks it looks ridiculous him riding bikes. Of course he wants to win and thus asks you to determine a strategy resulting in the fastest possible time to complete the marathon.

Further rules of the competition are as following. Once participants arrive at the next tube station to visit they are required to get a stamp at a checkpoint. While you can assume that getting the stamp takes zero time, it takes C_u minutes to get from any tube platform to the checkpoint and C_c minutes to get there from a bus station. Times for getting from the checkpoint to a tube platform or to a bus station, analogously. Checkpoints exist only at stations that are part of the tube system. The competition starts and ends at the checkpoints of the (lexicographically) first and last tube stop respectively. The competition starts at time 0 and ends at time E, finishing at E is fine. For simplicity departure times are given with respect to the starting time of the competition. Times are given in minutes. Participants are allowed to pass/ride through any stations in order to visit the next station on the list.

You are given an underground map consisting of U different tube lines with $N_i^u, i \in \{1..U\}$ tube stations each. You are also given a bus map consisting of B different bus lines with $N_i^b, i \in \{1..B\}$ bus stations each. For each tube/bus line you are given the departure time F_i of the first train/bus, the interval I_i between departing trains/buses and the time L_i of the last train/bus. Trains and buses always depart in opposite directions both at the first and last station of a line on the map. The journey time between tube stations is given as T_u and between bus stations as T_b. Journey times between stations are the same for all lines.
Changing between tube services, bus services or between tube and bus service is possible, but only at stops having an identical name. Changing platforms between tube lines takes C_u minutes, between bus services takes C_b minutes and between corresponding tube and bus stops C_c minutes.

Input

The first line contains the number of test cases T ($1 \leq T \leq 100$). Then follows for each test case:

- A line containing five integers: the end time E ($1 \leq E \leq 1\,440$), the number of tube lines U ($1 \leq U \leq 10$), the number of bus lines B ($0 \leq B \leq 10$), the journey time between tube stations T_u ($1 \leq T_u \leq 10$), and the journey time between bus stops T_b ($1 \leq T_b \leq 10$).
- A line containing three integers: the transfer time between tube platforms C_u ($1 \leq C_u \leq 25$), between bus lines C_b ($1 \leq C_b \leq 25$) and between tube platform and bus stop C_c ($1 \leq C_c \leq 25$). It holds $max(C_b, C_u) \leq C_c$.
- Then follow $2U$ lines, each pair of lines describing one of the tube lines. The first of each pair of lines contains four integers: the departure time of the first train F_i ($0 \leq F_i \leq 1\,440$), the interval between trains I_i ($1 \leq I_i \leq 1\,440$), the departure time of last train L_i ($1 \leq L_i \leq 1\,440$) and the number of stops on this tube line N_i^u ($1 \leq N_i^u \leq 20$). It holds $L_i \geq F_i$ and $L_i - F_i$ is a multiple of I_i. The second line contains N_i^u distinct station names. Each station name consists of between 1 and 25 lower case characters.
- Then follow $2B$ lines describing each of the bus lines. The input format is equivalent to the format of the tube lines.

Output

For each test case output a single line with the minimum time in minutes to visit all tube stations on the map in lexicographic order. If it is not possible to visit all stations within the given time E output "`IMPOSSIBLE`".

Sample Input 1

```
4
100 1 0 9 9
6 7 8
10 7 80 3
a c b
563 2 0 10 7
3 2 5
100 20 600 5
a b c d e
100 20 600 5
aa bb c dd ee
500 2 1 5 5
5 5 5
0 10 100 10
a b c d e f g h i j
0 10 300 10
k l m n o p q r s t
100 10 200 4
j jj kk k
1000 2 0 5 5
5 5 5
0 10 200 2
a b
0 10 200 2
c d
```

Sample Output 1

```
60
563
310
IMPOSSIBLE
```