

12294 RPG battles

In many typical RPG games, you battle with bad guys, creatures, monsters or ghosts etc. all the time. After each battle, you may get magic potions that power you up, so you'll get stronger and stronger, and finally defeat the big boss. In this problem, we only consider two kinds of potion: *stronger* and *double power*. If you drink a bottle of *stronger potion*, your power increases by 1; if you drink a bottle of *double power potion*, your power doubles.

How long each battle lasts depends on how powerful you are. Each battle is described by six parameters: p_1 , p_2 , t_1 , t_2 , w_1 , w_2 . That means, if your power is less than p_1 , you will be defeated; if your power is greater than p_2 , you'll need t_2 seconds to defeat all the enemies. If your power is between p_1 and p_2 (inclusive), the time needed for the battle decreases linearly from t_1 to t_2 . For example, if $p_1 = 50$, $p_2 = 75$, $t_1 = 40$, $t_2 = 15$, and your power is 55, then the battle lasts for 35 seconds. Note that the time needed for battles may be non-integers. The last two parameters, w_1 and w_2 , represent the number of bottles of stronger potion and double power potion you got after winning the battle. Note that you don't have to drink these potions immediately. You can drink them later if that'll decrease the total time. You cannot drink potions during a battle, so the time needed for battles is not affected by the potions.

Given the list of battles, your task is to find a strategy to win all the battles as quickly as possible. Note that you must enter the battles in order. You cannot redo or skip any battle.

Input

There will be at most 25 test cases. Each test begins with two integers n and p ($1 \leq n \leq 1000$, $1 \leq p \leq 100$), the number of battles and your initial power. Each of the next n lines contains 6 integers p_1 , p_2 , t_1 , t_2 , w_1 , w_2 ($1 \leq p_1 < p_2 \leq 100$, $1 \leq t_2 < t_1 \leq 100$, $0 \leq w_1, w_2 \leq 10$). The input is terminated by a test case with $n = p = 0$, you should not process it.

Output

For each test case, print the shortest time (in seconds) to win all the battles, to two digits after the decimal point. If you cannot win all the battles, print 'Impossible' (without quotes).

Sample Input

```
1 55
50 75 40 15 10 0
2 55
50 75 40 15 10 0
50 75 40 15 10 0
3 1
1 2 2 1 0 5
1 2 2 1 1 0
1 100 100 1 0 0
1 7
4 15 35 23 0 0
1 1
2 3 2 1 0 0
0 0
```

Sample Output

35.00

60.00

41.00

31.73

Impossible