You live in a small town with R bidirectional roads connecting C crossings and you want to go from crossing 1 to crossing C as soon as possible. You can visit other crossings before arriving at crossing C, but it's not mandatory.

You have exactly one chance to ask your friend to repair exactly one existing road, **from the time you leave crossing 1**. If he repairs the i-th road for t units of time, the crossing time after that would be $v_i a_i^{-t}$. It's not difficult to see that it takes vi units of time to cross that road if your friend doesn't repair it. You cannot start to cross the road when your friend is repairing it.

Input

There will be at most 25 test cases. Each test case begins with two integers C and R ($2 \leq C \leq 100$, $1 \leq R \leq 500$). Each of the next R lines contains two integers x_i, y_i ($1 \leq x_i, y_i \leq C$) and two positive floating-point numbers v_i and a_i ($1 \leq v_i \leq 45$, $1 \leq a_i \leq 5$), indicating that there is a bidirectional road connecting crossing x_i and y_i, with parameters v_i and a_i (see above). Each pair of crossings can be connected by at most one road. The input is terminated by a test case with $C = R = 0$, you should not process it.

Output

For each test case, print the smallest time it takes to reach crossing C from crossing 1, rounded to 3 digits after decimal point. It's always possible to reach crossing C from crossing 1.

Sample Input

```
3 2
1 2 1.5 1.8
2 3 2.0 1.5
2 1
1 2 2.0 1.8
0 0
```

Sample Output

```
2.589
1.976
```