Traffic Jam is becoming a major problem in the city of 'Faka' these days. Every day, you need to go to work from home through a one-way road that can be modeled as a series of straight line segments (aka piecewise linear) of positive length. To be more explicit, suppose there are one way straight line paths named P, Q and R in your way. P starts at your home & ends at the starting point of Q. Q starts from the ending point of P & ends at starting point of R. Similarly, R starts at ending point of Q & ends at your destination. Two nonadjacent line segments in a road will never intersect with each other.

Now, you start from home for work and immediately get irritated by the heavy and annoying traffic jam. So you wish you could fly to your destination in order to avoid spending the whole day on road. Your new vehicle Aerocar comes as the solution to you. With this car, you can fly from any point on the road (The aerocar has a vertical take-off & landing similar to that of a helicopter) and land back to any point on the road any time. Now, though you would have liked to fly all the way and thus avoiding the traffic, you need to resist the temptation due to the fact that flying requires more fuel than driving. You need 1 unit of fuel to travel every unit distance along the path while F units per unit distance while flying. Now you need to write a program that, given the description of the road, can calculate the minimum possible fuel amount needed to complete your journey.

Input

Input file contains less than 60 test cases. Each test case starts with a couple of integers N ($1 \leq N \leq 25$) & F ($2 \leq F \leq 5$). N is the number of line segments in the road. F is the amount of fuel required for every unit distance traveled while flying. Next ($N+1$) lines each has two integers, x ($-1000 \leq x \leq 1000$) & y ($-1000 \leq y \leq 1000$). Assuming the road is placed on a planar 2d grid, the integers in the i-th line denotes the x and y coordinates of the starting point of the i-th line segment while the end point coordinates by the i-th line segment are given at the $(i+1)$-th line. The last test case will be followed by a case with $N = F = 0$ indicating the end of input. This case should not be processed.

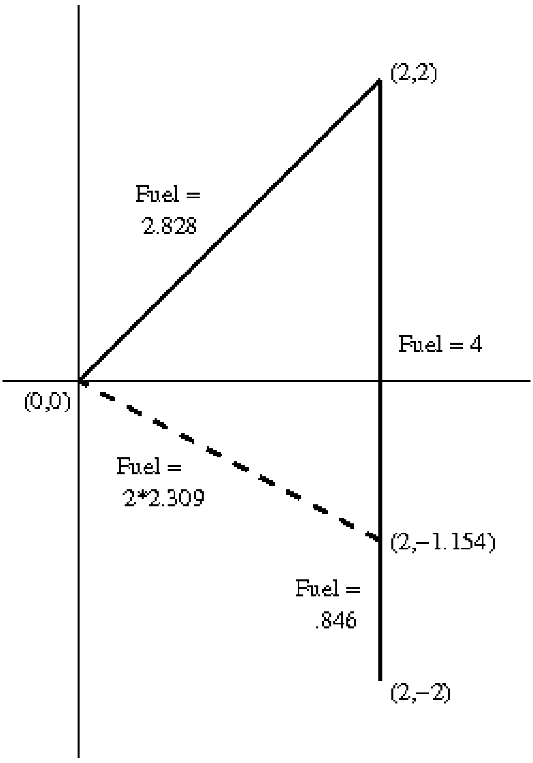
Output

For each test case except the last one, print one line of the form '`Case` X`:` Y', where X is the serial number of output & Y is the minimal possible units of fuel required. Print 3 digits after decimal point for Y. The output will be tested by a special judge program and outputs with precision error smaller than 10^{-3} shall be considered as correct.

Illustration:

The following figure illustrates the second test case of sample input. Your home is placed in the (0,0) point whereas the destination is at (2,-2). If the travel is completed through the roads, the fuel consumption will be 6.828 units (2.828 units to go to (2,2) from (0,0) and 4 for the (2,2) to (2,-2) part).

Alternatively, one can directly fly from (0,0) to (2,-1.154) with the required amount of fuel being 4.618 units (2.309 * 2), and then through the road to (2,-2) consumpting .846 more units of fuel, being the total 5.464 units. So, the best result is achieved by flying a part.



Sample Input

```
2 5
0 0
5 0
10 0
2 2
0 0
2 2
2 -2
3 5
0 0
1 1
-7 0
1 10
0 0
```

Sample Output

```
Case 1: 10.000
Case 2: 5.464
Case 3: 22.283
```