

206 Meals on Wheels Routing System

The Meals on Wheels program has the responsibility for providing hot meals to homebound senior citizens within the city. Volunteer drivers deliver the meals in a timely manner to ensure that they are still hot on arrival. The list of customers for meals and the number of available drivers varies on a daily basis. For each day, management tries to assign drivers routes so that allocation of customers is as even as possible among the routes.

An algorithm for developing the daily routes involves sorting the addresses of the meal customers based on the directions of their locations relative to the Meals on Wheels headquarters and dividing this sorted list among the available drivers. The Meals on Wheels headquarters is considered to be located at the origin on a cartesian grid of square city blocks. Each customer's address has been converted into the number of city blocks in the x and y directions from the Meals on Wheels facility. For example, a customer living at location $(3,-2)$ would be living 3 blocks east and 2 blocks south of the Meals on Wheels headquarters.

Write a computer program to determine routing for several different days. For each day, your program will read in the number of drivers (routes) and number of customers followed by sets of names and locations for the customers. Allocate customers to routes using the following strategy.

- Find the polar coordinate of each customer's location. Consider 0° to be due east and 90° to be due north.
- Sort the sets of polar coordinates by angle and then divide the customers as equally as possible among the available routes starting at the angle of smallest measure.
- Routes with customers at high degree angles should not have more customers than those for customers of lower degree angles.
- If two customers are at the same angle, then assign the customer nearer to the Meals on Wheels headquarters before you assign the one further away.
- The difference in the number of customers assigned to any two routes may not exceed one.

Your program will determine not only the route for each driver but also the total length of each route. The length of any route includes the sum of the distances from the Meals on Wheels headquarters to the first customer, plus the distances between subsequent customers, plus the distance back to the headquarters from the last customer on the route. Note that a block may not be traversed diagonally, and all city blocks are squares.

Input

Input consists of *multiple* data sets in which the first line is a data set ID and the second line contains the number of routes followed by the number of customers. The remaining lines of the data set are arranged in pairs, one pair per customer. The first line of each pair is the customer's name and the second line contains the x and y coordinates of where that customer lives. Assume that no two customers live on the same position and that no customer lives in $(0,0)$. So each data set is arranged in the following manner.

Line 1: data set ID (string — maximum length 50 characters)
 Line 2: $n m$ (number_of_routes number_of_customers — positive integers)

The next $2m$ lines come in pairs:

Line 3: customer name (string — maximum length 25)

Line 4: x -coordinate y -coordinate (x and y coordinates for the preceding customer — integers)

Assume the input is correct and the number of routes does not exceed the number of customers. The end of input is indicated by end-of-file.

Output

For each data set your output should include the data set ID, the number of customers, the number of routes, the routes of customers in order along with their corresponding route lengths, and the total route length for all routes in this data set. Print a row of asterisks between output for successive data sets.

Sample Input

```
Sample Route List 1
4 10
able
1 2
baker
-3 6
charlie
-4 -5
donald
4 -7
eloise
3 4
frank
2 2
gertrude
5 9
horace
-2 -5
inez
5 -3
james
0 1
Sample Route List 2
1 1
charlie
1 1
```

Sample Output

```
Sample Route List 1
Number of Customers: 10          Number of Routes: 4

Route ==> 1
```

Customer: frank
Customer: eloise
Customer: gertrude
Route Length ==> 28

Route ==> 2
Customer: able
Customer: james
Customer: baker
Route Length ==> 22

Route ==> 3
Customer: charlie
Customer: horace
Route Length ==> 18

Route ==> 4
Customer: donald
Customer: inez
Route Length ==> 24

Total Route Length ==> 92

Sample Route List 2

Number of Customers: 1 Number of Routes: 1

Route ==> 1
Customer: charlie
Route Length ==> 4

Total Route Length ==> 4