REPLY OFFICES

Reply is made up of a network of highly specialised companies, which support leading commercial groups using new technology.

Since 1996, we've been growing and growing. Back then, we started with one Italian office. Today, we have 37 offices across 36 European cities. But the world is big, and we keep growing – which means we need to open more new offices.

To work efficiently, we need to be as close as possible to our main customers. So our new challenge is to choose the right locations to improve our customer experience. And we need your help.

PROBLEM STATEMENT

Given the map of the world, the terrain features, and the locations of Headquarters of our Customers, you need to build a number of Reply Offices that minimize the overall distance between the Offices and the Customer Headquarters. You can choose to place an office only on areas suitable for construction.

For each Reply Office you are going to build, you need to:

- provide its position on the map
- connect it with one or more Customer Headquarters

The world map is a grid composed by 'terrain cells' of different types. If a cell is **non-walkable**, you cannot build Offices in it and no path can cross it. Every other cell is walkable and suitable for construction. You cannot build Offices in the same cell of a Customer Headquarter.

A path is a series of consecutive cells going from a Reply Office to a Customer Headquarter. From a cell you can only step UP, RIGHT, DOWN, LEFT to the next cell. No diagonal steps are allowed. Each step has a cost based on the type of the terrain of the next cell.

Each reached client generates a reward. The score for a client is computed as the reward of the client minus the cost to reach it. Your total score is computed as the sum of the scores for each connected client.

Each Office built must have a path to at least a Customer.
Given a pair (Office O, Customer C) it can't exist more than one path from O to C.

INPUT FORMAT

Input data will be provided in plain ASCII text file.
On the first line you will find four integers separated by a whitespace character:

- N: the width of the map
- M: the height of the map
- C: the number of Customer Headquarters
- R: representing the maximum number of Reply Offices that can be built

C lines follow, each one built of three integers describing the X coordinate, the Y coordinate, and the reward associated with Customer Headquarter.
Then, M lines follow, describing a row of N terrain cells. The top-left cell is the origin of the map and is thus assigned "(0, 0)" coordinates. In the world map columns are represented by the X coordinate, while rows by the Y coordinate.
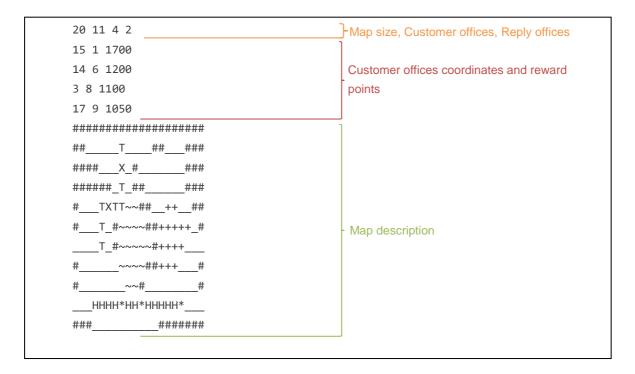
Each cell represents a different kind of terrain, and is rendered with an ASCII character according to the following table:

Character	Meaning	Cost of moving through that terrain
#	Mountains	Non-walkable cell
~	Water	800 points
*	Traffic jam	200 points
+	Dirt	150 points
X	Railway level crossing	120 points
_	Standard terrain	100 points
H	Highway	70 points
T	Railway	50 points

Table 1: Terrain characters and traversal costs

INPUT EXAMPLE

```
20 11 4 2              ⌐Map size, Customer offices, Reply offices
15 1 1700
14 6 1200              Customer offices coordinates and reward
3 8 1100               points
17 9 1050
####################
##_____T____##___###
####___X_#_____###
######_T_##_____###
#___TXTT~~##__++__##
#___T_#~~~~##+++++_#
____T_#~~~~~#++++___    Map description
#_____~~~~##+++___#
#_____~~#_____#
___HHHH*HH*HHHHH*___
###_____#######
```

CONSTRAINTS

$1 \leq R < C \leq 500$

$1 \leq N \leq 2.000$

$1 \leq M \leq 2.000$

OUTPUT FORMAT

For each path going from a Reply Office to a Customer Headquarter, output a single line built of the Reply Office X coordinate, Y coordinate, and a string representing the sequence of steps to reach the Customer Headquarter. Separate each of these three components by a single whitespace character.

The Reply office coordinates are integer numbers.

The sequence of steps is a string composed by the ASCII characters U, R, D, L (UP, RIGHT, DOWN, LEFT).

OUTPUT EXAMPLE

```
2  5  URRRRRUUURRRRDRRRRU
2  5  URRRRRUUURRRRDRRRDDDD
2  5  DDDR
16  7  LLU
16  7  DDR
```

SCORING

Your solution will be checked against the rules described above before being graded. Only valid solutions will receive a score. You can build a number of Reply Offices less or equal the value given in the input file. For each Office you build there must be a path to at least one Customer Headquarter. If each Customer Headquarter has a path to at least one Reply Office, you will get a bonus score.

The score for a client is computed as the reward of the client minus the cost to reach it. The cost to reach a client is the sum of the traversal cost of all the symbols traversed on the path from Reply Office to Customer Headquarter (the starting symbol is not included in a path, while the ending symbol is included).

The total score for an input is computed as the sum of the scores for each connected client. If all Customer Headquarters have a path to at least one Reply Office, add to the score a bonus value equal to the sum of the rewards for all clients.

If the resulting score is negative than the score is automatically assigned as 0. The higher the score, the better.

More formally:

Given a cell $c = (x, y)$ and a direction $d \in (U, D, L, R)$, Let's define the movement function:

$$move(c, d) = \begin{cases} (x, y-1) & if\ d = U; \\ (x, y+1) & if\ d = D; \\ (x-1, y) & if\ d = L; \\ (x+1, y) & if\ d = R; \end{cases}$$

and the cell cost function $cost(c)$ as the traversal cost of the symbol represented at the position $c = (x, y)$ in the map, according to Table 1: Terrain characters and traversal costs. Let's also define the reward function $reward(c)$ as the value of the reward associated to the customer office at coordinates c according to the second section of the

input files, or 0 if there's no office at location c.

Given an initial cell $c = (x, y)$ and a sequence of directions $d = \{d_0 .. d_n\}$, $d_i \in (U, D, L, R)$, let's define the sequence "path"

$$p = \begin{cases} p_0 = move(c, d_0) \\ p_i = move(p_{i-1}, d_i) \end{cases}$$

For each row of the input file, let's call p_r the path that starts at the indicated Reply office coordinates (x, y) and built applying the definition of p with the given sequence of steps.

The score associated with such row is then defined as

$$s_r = reward(p_n) - \sum_{i=0}^{n} cost(p_i)$$

The total score for all rows is calculated as the sum of the scores of each row, or:

$$total = \sum_{r} s_r$$

If all Customer Headquarters have a path to at least one Reply Office, add a bonus calculated as the sum of the rewards for all clients (otherwise, the bonus is 0):

$$bonus = \sum_{n} reward(p_n)$$

The final score associated with each output file is then calculated as the maximum value between total plus bonus and 0:
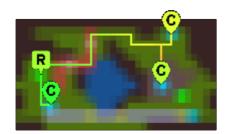
$$score = \max(total + bonus, 0)$$

SCORING EXAMPLE

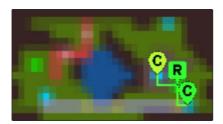All the four Customer headquarters are reached, so the solution is valid.

About the Office built at coordinates (2,5):

- 1st Customer, Path URRRRRUUURRRRDRRRRU: 1700 - (100 + 100 + 50 + 120 + 50 + 50 + 50 + 120 + 50 + 100 + 100 + 100 +100 + 100 + 100 + 100 + 100 + 100)
  = 1700 - 1690 = 10

- 2nd Customer, Path URRRRRUUURRRRDRRRDDDD: 1200 - (100 + 100 + 50 + 120 + 50 + 50 + 50 + 120 + 50 + 100 + 100 + 100 +100 + 100 + 100 + 100 + 100 + 100 + 150 + 150 + 150)
  = 1200 - 2040 = -840

- 3rd Customer, Path DDDR: 1100 – (100 + 100 + 100 + 100) = 700

Final score for the first office: $10 + (-840) + 700 = $ **-130**

About the Office built at coordinates (16,7):

- 2nd Customer, Path LLU: 1200 - (150 + 150 + 150) = 750

- 4th Customer, Path DDR: 1050 – (100 + 200 + 100) = 650

Final score for the second office: $750 + 650 = $ **1400**

As all customer have at least one path, there is a bonus score: 1700 + 1200 + 1100 + 1050 = 5050

Total scoring: $-130 + 1400 + 5050 = $ **6320**

FINAL NOTES

You will be provided with multiple input files, different from one another. You must submit at least one output for each attempt. You're not obliged to submit output files in order, i.e. you can create a submission with just your solution for the second and the fourth problem.

Each submitted file will be validated, and if it's accepted it will receive a score according to the scoring function described later in the document. Each submission will be given a score equals to the sum of the scores of the output files it contains. This score is just a hint on how well you're behaving and doesn't count toward the leaderboard.

The final score of your team, which will be used for creating the leaderboard, is given by the sum of your best scores of each output, even if you scored such maximum on different submission. So don't be shy and try as many submissions as you can!

Good luck, and enjoy the Challenge!