

# Food Shortage in Byteland

Fanatics from the BBFO blew up all the food factories in the Bytelandian capital! Hurry up! There is still some food left in shops. Some shops are located in the centre, others in the suburbs, so Johnny has to decide which of them are worth visiting. Some shops can be very big and have plenty of food in them, others may be so small that food disappears from them at an alarming rate... So? Help Johnny buy as much food as possible.

There are  $n$  open shops, each of them located at position  $(x_i, y_i)$ , for  $i=1, \dots, n$ , where  $0 \leq x_i, y_i \leq 250$ . The distances between shops are measured using the Manhattan metric (i.e. as sums of absolute values of differences of  $x$  and  $y$  coordinates). Besides, every shop is characterized by a linear time function describing how much food is left in the shop at the moment:

$$f_i = \max\{0, a_i - b_i * \text{time}\}$$

where  $0 \leq a_i \leq 1000000$ ,  $0 \leq b_i \leq 1000$ , while *time* is the time (in minutes) that has elapsed from the moment Johnny left the house (assume that Johnny does not live in the same place as any shop). If Johnny decides to stay in a shop, he can buy at most  $b_i$  units of food per minute. Otherwise, he can move along the orthogonal system of streets of the city at a constant speed of unit distance per minute. Johnny only ever changes the action he is performing at the full minute. Because his family is slowly beginning to starve, he should be back at home not later than  $m$  minutes after he left. Since there are thousands of starving families in the capital, Johnny can't spend more than  $1 \leq c_i \leq 10$  minutes in a shop. Moreover, he will never go into the same shop twice for fear of being lynched...

## Input

The first line of input contains a single positive integer  $t \leq 1000$ , the number of test cases. Each test case begins with the number of shops in the city  $1 \leq n \leq 1000$  and the deadline  $1 \leq m \leq 5000$ . Then the following  $n$  lines consist of four integers  $x_i y_i a_i b_i c_i$  each, describing the position and the parameters of the function for food availability of the  $i$ -th shop. At the end of every test case comes a line with two integers  $p q$  (between 0 and 250), corresponding to the  $x$  and  $y$  coordinates of the position of Johnny's house.

All the input data are integers.

## Output

Process all test cases. The correct output for the  $i$ -th test case takes the following form:

$i$  [the number of the test case, in the input order]

$s m$  [ $s$  is the number of the target shop and  $m > 0$  is the number of minutes spent in it].

At the end of the series of moves you should always write a line consisting of two zeros ('0 0').

All the output data should be integers.

## Scoring

The score of your program is the total amount of food that Johnny bought (summed over all the testcases in which he managed to come back home before the deadline).

## Example

### Input

```
4
2 20
0 0 100 5 5
10 0 200 10 10
5 0
2 20
0 0 180 15 10
10 0 200 20 10
5 0
4 101
0 0 1000 20 5
20 0 200 1 5
0 20 5000 200 5
20 20 300 5 10
10 10
1 15
1 0 10 1 5
5 0
```

### Output

```
1
2 10
0 0
2
1 10
0 0
3
3 5
4 10
2 1
0 0
4
1 5
0 0
```

### Score

Score = 1261