

People use to say that developers are unsociable. To prove that this is not true (well, and for win some money too) Douglas decided to go to the happiness contest in the television: The Price is Correct. The rules of the contest are really easy:

Douglas has to stand on a grid initially off. When the game starts, different squares begin to turn on and off. When a square is on, it shows a number in dollars. Douglas must run over all the grid, standing on the lighted squares; for every square that Douglas reaches before it turns off, he will win the amount of money that the square shows.

Every time that a square is turned on, last a second on and then off. However, it's possible that in the next second the square turns on again, with the same or other value. It is also possible that there are several lighted squares at the same time, and of course Douglas will be able to stand just one at a time.

Unfortunately, Douglas isn't free to run in any direction. Douglas will always begin in the middle of a square and every step he takes will be only to the center of an adjacent square. Developers are really fast to think but they are really slow to move so every step Douglas take will last one second. This slowness will make Douglas lose a lot of prizes, so he wants to know what is the sequence of step that he should take to ensure the largest amount of money. Given the initial position of Douglas, the appearing time and the amount of money of each prize, can you answer Douglas's question?

Input

Input begins with an integer t , the number of test cases. Each case begins with 3 integers N , M and P ($1 \leq N, M \leq 20$, $1 \leq P \leq 500$), that represents the number of rows, the number of columns and the amount of prizes, following that there will be a line with two integers X_o and Y_o ($1 \leq X_o \leq n$, $1 \leq Y_o \leq m$) Douglas's initial position in the grid. After that there will be p lines with 4 integers: X_i , Y_i , T_i , V_i , ($1 \leq X_i \leq N$, $1 \leq Y_i \leq M$, $1 \leq T_i \leq 2*P$, $1 \leq V_i \leq 1000$), the row and the column where the prize is, the second that the prize will appear and the value in dollars for the i -th prize. There will not be two prizes in the same square at the same time.

Output

Prints a single line for test case: an integer number representing the maximum number of dollars that Douglas can win.

Sample Input

```
1
4 3 4
1 1
1 2 1 10
3 1 3 5
3 3 3 15
2 1 4 15
```

Sample Output