Interview Coding Problem

Instructions:

It shouldn´t take you more than 2 hours to solve this problem. This application will be run from the command line with the input file being passed in as a parameter. The goal of this exercise is to get an idea of your coding and software design skills. Your code should be clear, concise, and easy to read.

Expected Solution:

Send all source files as well as instructions on how to run the program in one zip file. Bonus points for submitting the solution as a github or bitbucket repository instead of sending a zip file.

PROBLEM: Destroy as many zombies as possible

You are in charge of writing the software for a missile-launching platform that has to kill as many zombies as possible. Your platform has 3 missiles. As input you will receive a file with the dimensions of the grid you are to shoot the missiles at and the location and number of zombies within each grid square.

The grid area is defined by width and height and will always be a rectangle. Within this area each cell is identified by it's coordinate where the bottom left corner has the coordinate 1,1. Every cell of this area can contain 0-n zombies. In the file that gets passed in you will only receive coordinates that contain at least one zombie.

The input file you will receive will have the width and the height of the grid as first line and then any number of additional entries. Each additional entry will have be of the format:

X Y # of zombies present

Sample Input:

10 8
3 3 4
2 3 3
4 2 2
5 1 3
5 5 6
8 6 4

The kill zone of each of the 3 missiles that your launcher has can be seen here where **X** (the middle) is where the missile was fired, and O is a square where the missile will still kill a zombie. Empty squares are unaffected by the missile impact.

		O		
	O	O	O	
O	O	**X**	O	O
	O	O	O	
		O		

Expected output of your program is the coordinates of where you would fire each of the 3 missiles and the number of zombies each shot killed. For the coordinates above a possible solution is listed below.

Sample Output:

4 3 15
8 6 4
10 3 3