ITEC 120: Principles of Computer Science I

Homework 10 – Shooting Gallery

>>>>>>>> **Due Date: April 6th at 10:00 PM via Desire 2 Learn**   <<<<<<<

In doing this homework, remember to abide by the RU Honor Code.


Problem 1
30 points

You are an employee of Casual games Inc., which is in need of advertising revenue. Competitors are using mini-flash games to lure people to websites and your company is losing market share because they don't have something similar. Your boss has tasked you to write the engine for a shooting gallery game that allows score keeping. The art and testing department will produce a finished product, but they need a working engine before they can get started.

The game consists of several square targets that appear on the screen. When the user clicks the screen, an X and Y position will be sent to your program, and you will have to determine if a target has been hit. There will be no more than 100 targets on screen at any point in time. You should use an array with a tombstone or marker to keep track of how many targets are in the system.

The x and y position are at the upper left corner of the target. The x value corresponds to the horizontal axis and the y value corresponds to the vertical axis. Your program must include the capability to determine if a shot has hit the target.

Example target in the x,y coordinate system:
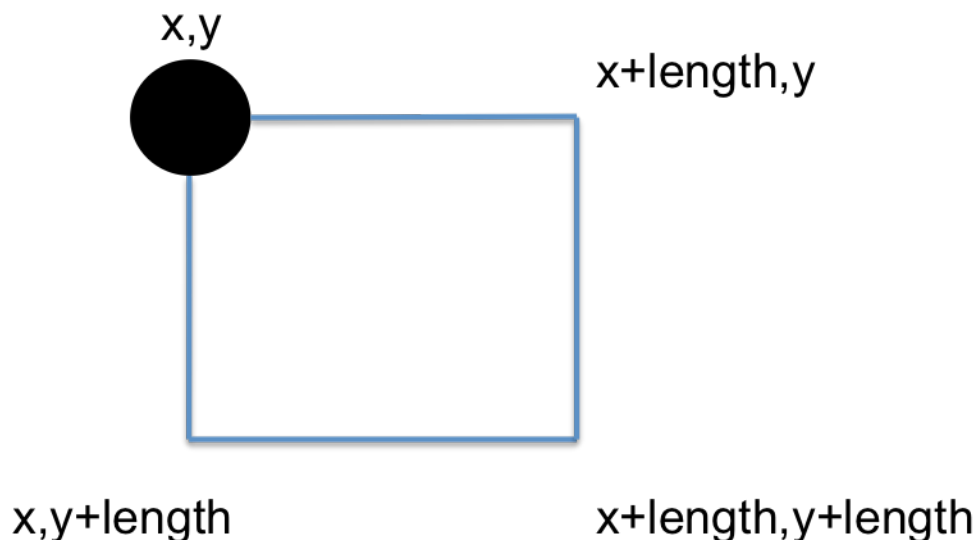
Scoring

When a user shoots a target (i.e. specifies a x,y position inclusive of the square mentioned earlier), one point should be added on the scoreboard. Targets can only be shot once.

Commands

Your program must read and respond to several commands from the scanner. The first command is addTarget, which adds a target to the system. The second command is moveTarget, which moves a specific target to a new location. The third command is shoot which determines if the user hit a target. The fourth command is score, which prints out the user's score. The last command is quit, which stops the program from executing.

addTarget: The add target command reads in an integer value for the x and y values of the target, and the length of the square. All values are on separate lines.

moveTarget: Reads in the current x and y location of a target and then reads in the new x and y location of the target. All values are on separate lines.

shoot: Reads in the x and y location of the shot and potentially updates the score of the game. When a user hits a target, print out a message saying that they hit a target.

score: Print out the score that the user has achieved by shooting targets.

Input

Input to your program will happen through the Scanner. You should output a prompt to the user that tells them what information to enter, i.e. Enter command>. An example input case is presented below:

```
addTarget
0
0
5
moveTarget
0
0
1
1
shoot
1
1
score
quit
```

The score for the input case is 1.

Testing note: your program must allow for a file to be redirected as input to the program. If it does not, then you will receive a 0 on this program. Note: you must create one and only one scanner in your program for his to work.

The reference solution for this project is 99 lines of code without comments. Feel free to use more or less code in your solution. This number is provided to help you gauge the difficulty of the assignment.

Constraints:
You must model the target as an object. The target object should store its location and length along with whether or not it has been hit. The target must also have a function that determines if a shot at a particular location hits the target.

You must also submit a test case that is similar but not identical to the previous usage scenario. By doing so, you demonstrate that you have tested your program beyond the basics.

Submission requirements:
You must submit the .java file containing your program to Desire 2 Learn under the Homework #10 assignment. If your submitted file does not compile, it will receive a 0. You can demo the homework the next school day after it is due, or it will be graded automatically. If you choose not to demo your homework you cannot contest the grade you receive.

Grading Rubric
3 Points – Is the target properly modeled as an object?
10 Points – Does it properly detect hits and report the correct score?
5 Points – Is the array of targets and tombstone implemented properly?
5 Points – Was a reasonable test case submitted with your program?
7 Points – Is the program commented using inline and javadoc comments?