**HW 8: Hare and Hound** **– Basic GUI Display with User Interaction**

Problem Description:

You are to create a program that simulate the launching of two projectiles, Hare and Hound. These projectiles will have two different movement methods. The Hare will be launched first and will fly until it hits the ground. Once the Hare lands, the hound will be launched. The Hound is computer controlled which will determine when the Hound needs to burn fuel. Once Hare and Hound both have landed, the program will calculate their distance. Based on their distance, the program shall declare a winner.

Projectile motion equations:

|  |  |
| --- | --- |
|  | X: x displacement  X0: initial x displacement  Y: y displacement  Y0: initial y displacement  Vo: initial velocity  t: time  : initial angle  g: gravitational constant |
| Keep in mind that the X and Y displacement is the distance from the projectiles origin at a given time stamp. See examplein HW 3. | |

Program Requirements:

1. The program shall use the first quadrant of a 2D coordinate plane.
2. The program shall have a max height of y = 2500 meters.
3. The program shall have a max width of x = 2000 meters.
4. The program shall have 2 projectiles named Hare and Hound.
5. The Hare projectile shall follow projectile motion equations.
6. The Hare shall have a random initial angle ranging from 0 to 90 degrees.
7. The Hare shall have a random initial velocity ranging from 70 to 120 meters per second.
8. The Hare shall have a random initial height ranging from 400 to 1000 meters.
9. The Hare shall increase its X position by 10 meters every 1.5 seconds during flight.
10. The Hound shall have a constant vertical velocity of -9.81 meters a second.
11. The Hound shall have a constant horizontal velocity of 11.3 meters a second.
12. The Hound shall have a starting height of 600 meters.
13. The Hound shall have the ability to “burn fuel” which increases its current height by 11 meters.
14. The Hound shall be controlled by an AI which will determine when the Hound needs to “burn fuel”.
15. The Hound AI shall attempt to land the Hound as close to the Hare as possible.
    1. **[IMPORTANT] This AI can range from randomly hitting “burn fuel” to calculating when it should “burn fuel”. Thus, the AI as smart or as dumb as you want.**
16. The Hare shall stop moving once it hits the ground.
17. The Hare shall stop moving once it hits the max height.
18. The Hare shall stop moving once it hits the max width.
19. The Hound shall stop moving once it hits the ground.
20. The Hound shall stop moving once it hits the max height.
21. The Hound shall stop moving once it hits the max width.
22. The program shall calculate the distance between the Hare and the Hound once they have both hit the ground.
23. The Hound shall win when the distance between the hare and hound is under 7 meters.
24. The Hare shall win when the distance between the hair and hound is greater than or equal to 7 meters
25. The program shall display the winner.
26. The program shall start both Hare and Hound on the left side of the screen (x = 0).
27. The program shall define ground as the following:
    1. A series of 20 plateaus.
    2. A single plateau shall have a random height between the ranges of 100 to 350 meters.
    3. The width of each plateau shall be the 100 meters.

Homework Requirements:

1. The program shall run one iteration.
2. An iteration shall be started by the user by use of a JavaFX button (or something similar)
3. An Iteration shall consist of:
   1. Setting up the Hare and hound (see Program Requirements 6-11)
   2. Setup the ground ((see Program Requirements 29).
   3. Launching the Hare to follow projectile motion.
   4. Once the Hare hits the ground, launching the Hound.
   5. Once the Hound hits the ground:
      1. The program shall display via JavaFX the distance between the Hare and the Hound.
      2. The program shall display via JavaFX who won the game.
      3. These can be displayed in simple labels or other text objects.
4. The program shall have 5 classes HHGUI, Manager, Projectile, Hare, and Hound.
5. The Projectile class shall handle the location and movement.
6. The Hare class shall handle the movement of the hare by extending the Projectile class.
7. The Hound class shall handle the movement of the hound by extending the Projectile class.
8. The Manager class shall handle the 1 iteration using 1 Hare and 1 Hound objects.
9. The HHGUI class shall handle the graphic display of the program, as well as the user interaction.
10. Feel free to decorate your GUI as much or little as you want.

Time estimate: 3-8 hours

Rubric:

|  |  |  |
| --- | --- | --- |
| **Deliverable** | **Points** | **Awarded** |
| Code compilation | 5 |  |
| Fulfilled the requirements | 25 |  |
| Totals | 30 |  |