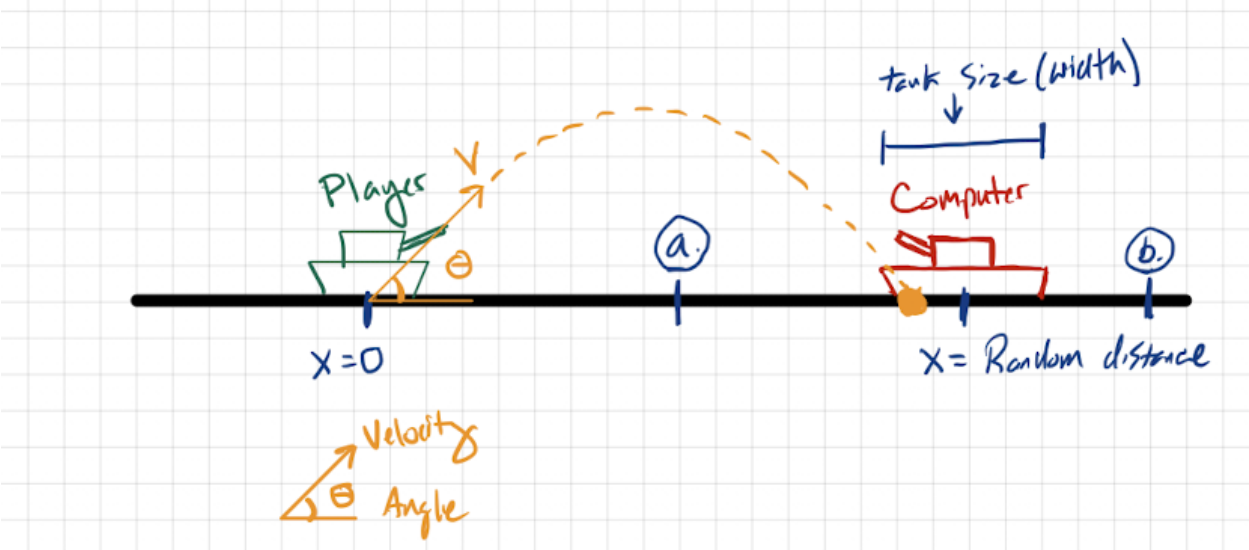
CPSC121 Assignment 5 - Artillery

Your task is to create an artillery game. There are two players, a human player and a computer player. The game plays as follows:



- Ask for the player's name, this may or may not include spaces.
- Ask for the desired difficulty; easy, medium, or hard.
- An enemy tank is placed at a random distance from the player, you choose a good range for this value. It must be placed within a reasonable distance to be able to play the game, meaning it's always placed to the right of the player tank which is located at x = 0, and between some limits denoted by a.) and b.) above.
- The enemy tank is placed at a random distance within [a, b] at the beginning of each game, not each round. The tanks do not move once the game has started.
- There are three rounds
- Each round the player is asked for a velocity and an angle to shoot their artillery at the enemy. The angle and velocity is shown in yellow above. The input from the player must be in degrees, but the equation to calculate the projectile's trajectory is in radians. You must convert from degrees to radians then plug into the equation below to find how far the projectile traveled.

  $$distanceTraveled = velocity * cos(angle\ in\ radians)$$

  $$radians = degrees * \frac{\pi}{180}$$

- If the player hits the enemy, the game is over. The player wins!
- If the player misses the enemy, the game will report to the player if they undershot or overshot and by how much.

- If the player doesn't destroy the enemy at the end of the three rounds, the enemy wins!
- There are three difficulties: Easy, Medium, Hard
    - A "Hit" is registered if your projectile lands within the 'width' of the enemy tank:
        - Easy: enemy tank width = 20 units
        - Medium: enemy tank width = 10 units
        - Hard: enemy tank width = 5 units

For example: If the enemy tank is placed at x = 50, and the game mode is set to easy, then if the player shot lands in [40, 60] it is registered as a hit. If the tank is placed at x = 75 and the game mode is medium, a shot landing in [70, 80] will register as a hit.

- At the end of the match, print out a personalized message to the player, including their name (which may include spaces), that says something about being defeated or victorious. Feel free to have fun with this, if the player wins on the hardest difficulty you could be more celebratory than if they win on easy, etc.

Notes:

- You may assume we are getting valid inputs, except for difficulty. If the player enters a string that represents an invalid difficulty, print a message stating this and terminate the program gracefully via a return statement.
- Print out the round number and difficulty level at the start of each round
- Print out your shot location each time you fire.
- If you hit the tank, print out the shot location and the tank location.
- Format doubles to two decimal places for output
- It would be a good idea to print a debug statement showing the location of the enemy tank while testing, then remove the debug statement before submitting your finished work.

Sample runs of the program:

(Note the tank position debug statement, remove this for your final product but good for testing!)

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 _____           _      ____        _   _   _
|_   _|_ _ _ __ | | __ | __ )  __ _| |_| |_| | ___
  | |/ _` | '_ \| |/ / |  _ \ / _` | __| __| |/ _ \
  | | (_| | | | |   <  | |_) | (_| | |_| |_| |  __/
  |_|\__,_|_| |_|_|\_\ |____/ \__,_|\__|\__|_|\___|
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~


Enter your name:
Bryan Fischer
Choose your difficulty: easy, medium, hard
easy
DEBUG: Tank position is: 120

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Round number: 1 Difficulty: easy
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Enter an angle (in degrees):
45
Enter a velocty:
100
Your shot landed at: 70.71 and was too short!

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Round number: 2 Difficulty: easy
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Enter an angle (in degrees):
45
Enter a velocty:
200
Your shot landed at: 141.42 and was too far!

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Round number: 3 Difficulty: easy
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Enter an angle (in degrees):
45
Enter a velocty:
130
Your shot landed at: 91.92 and was too short!
You failed to destroy the enemy tank in time. Bryan Fischer, you are defeated! (It takes talent to lose on EASY)
```

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 _____           _      ____        _   _   _
|_   _|_ _ _ __ | | __ | __ )  __ _| |_| |_| | ___
  | |/ _` | '_ \| |/ / |  _ \ / _` | __| __| |/ _ \
  | | (_| | | | |   <  | |_) | (_| | |_| |_| |  __/
  |_|\__,_|_| |_|_|\_\ |____/ \__,_|\__|\__|_|\___|
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~


Enter your name:
Bryan Fischer
Choose your difficulty: easy, medium, hard
easy
DEBUG: Tank position is: 199

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Round number: 1 Difficulty: easy
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Enter an angle (in degrees):
45
Enter a velocty:
100
Your shot landed at: 70.71 and was too short!

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Round number: 2 Difficulty: easy
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Enter an angle (in degrees):
45
Enter a velocty:
300
Your shot landed at: 212.13 and was too far!

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Round number: 3 Difficulty: easy
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Enter an angle (in degrees):
45
Enter a velocty:
280
Your shot landed at: 197.99
HIT!
Enemy tank was located at: 199
Congratulations Bryan Fischer, You win! (pffft but it was only on easy)
```

Submission details:

- A .zip folder named lastname_firstname_assignment4.zip
    - Inside this folder should contain the following:
        - Artillery.cpp
        - Makefile (unchanged from what was given to you)
        - a results.txt file that contains at least 5 test runs of your program
        - a reflection.txt file that contains your answers to the following questions:
            - What was the most difficult part of this assignment for you?
            - What was the easiest?
            - If you were to expand this idea and add more features, what would you add and why?
            - Having completed the assignment, if you were to redo it, what would change in your approach/strategy? How would your implementation be different from your first attempt? Would it be easier or harder the second time?

Rubric:
60 points

Code compiles without errors or warnings	10 points (non-compiling code will be given 0 points)
Program functions properly for each case, easy, medium, and hard. Correctly detects hits and misses Gives proper feedback to the player about shots being too far or too short Ends after 3 rounds Easy medium and hard game modes change the range of victory conditions Output roughly follows the example given, and adheres to the written requirements above	30 points
Program catches invalid difficulty and exits gracefully	5 points
All files are included in the .zip and named correctly	5 points
Reflection.txt and results.txt are completed	10 points