

HW8: Love Triangle

Due: 11/21/2021 11:59PM (merry xmas)

You just landed your first tech job in the big city! After moving into your fancy new high rise apartment (which you can afford because you got a bouje job), you are finally ready to get a life! As time goes on you eventually find yourself fond of another human being in the city. Perhaps a neighbor, a co-worker, a friend of a friend, someone you just bumped into, or more realistically in modern times, someone you met on one of those dating sites (hinge, coffee meets bagel, match, farmersonly.com, etc.)

However, in the true nature of a ex-comp sci major, instead of just asking them out on a date, you write a game in an attempt to simulate the chances you have with the Love Interest over your arch nemesis in love, the Rival.



Your new bouje apartment. Rent is \$8,500 a month. Has a professional gas range and jacuzzi tub in the bathroom. Landlord isn't an a**hole but you're somewhat disappointed the place doesn't come with a butler.

Game Rules

- The game starts out with an $N \times N$ grid that represents a city. Conveniently, all cities are exact squares (imagine a universe with good city planning).
- Each turn represents a day. Each day/turn, you will choose a point in the grid and so will your rival. The objective is to choose a point closest to the Love Interest. The location of the Love Interest is concealed until the end of the turn.

- At the end of each turn, the point chosen by the Love Interest will be revealed. Then the distance between the points chosen and the Love Interest will be calculated. The calculated distance will be assigned as points to the respective player.
 - e.g If you choose point (2,3), your Rival chooses point (2,6), and the Love Interest chooses point (5,5). 3.6055 points will be assigned to you and 3.1622 points will be assigned to the Rival.
- **Win Condition:** The person with the lowest score (between you and the Rival) after 5 turns gets a date with the Love Interest/wins.
 - Note: The number of turns may be subject to change. Meaning at one point we may suddenly tell you that the game is now 10 turns.
 - Note: **One** date, don't push it.

Specifications

Most of the game will be within a "Match" class. This class will have:

- an array of Games, should contain previous games played
- a **final_results()** function that will print out the win rate
- a **play()** function that will initiate a game

There will also be a "Game" class that contains

- a 2d dynamically allocated array, for the size of the city
- randomly generated coordinates for the Rival for each turn
- randomly generated coordinates for the Love Interest for each turn
- score of the Player
- score of the Rival
- a **reset()** function
- a **results()** function
- an overloaded << operator for printing out the 2D array
- whatever constructors you need
- a destructor

Of course you can add classes or other functions as necessary.

The program will prompt the player for the size of the city before doing anything else. Your program will then randomly generate all coordinates for each turn, for both the Rival and the Love Interest before the first turn. All the coordinates the Rival will play and all the coordinates of the Love Interest will be generated for all 5 turns before the game begins.

Before turn one, print out the board with numbers on both axes so the user knows what to input. The player will then choose an x coordinate directly followed by a y coordinate. After the player presses enter, the board will be printed again, but this time with the location of the Rival indicated by a 'R', the location of the Love Interest indicated by a 'L', and your location indicated by a 'P'.

This will be followed by two lines indicating how many points were assigned that turn. If any entity chooses the same point, the Love Interest takes first precedence, followed by the Player, and finally the Rival.

E.g if the player and the rival choose the same point, only a 'P' will be displayed. If all entities choose the same spot, only a 'L' will be displayed.

After all has been printed, the next turn will begin and the user will have to choose a location again.

Once the game is finished, use a **results()** function to print out the results of the game.

The program will then ask if the user wants to play again and if the user wants to play in a different sized city.

If the user wants to play with a city of the same size, make a copy of the Game and clear the scores of the new copy with a **reset()** function. When copying the Game, you should use a separate function that should be a constructor of some sort.

If the user wants to play in a different city, create a new Game.

If the user declines to continue this match or if the user has already played 20 times, print the winrate of the Player to the screen. To calculate the win rate, you will have to have a history of past cities, count the winners of each Game, and then divide that by the number of cities the match was played in.

Notes/Tips

- For random numbers, use a seed of 8 when you turn in the assignment.
- Hint: To keep track of the history of Games, you may want to store every single Game used inside of an array inside the Match class.
- When calculating scores, do not round.
- When storing pre-generated coordinates, you can use a static 2D array or 2 static 1D arrays.
- You may need to include the cmath library to calculate the distance between 2 points.
 - `#include <cmath>`
- As in tune with a true comp sci, your understanding of what gets you a date is a distance from reality. So don't actually use this for a real life reference, this is just a game, not a genie.
- In distance calculations use [Euclidean Distance](#)


$$\int_{t=0}^{\infty} \text{heart} e^{-st} dt$$

You're doing it wrong...

Sample Output

Note: This sample is purely for formatting reference, not to compare answers to. Output format does not have to be exact, but input format does. The three dots are ellipses to indicate “stuff happened in between here” and not part of the actual output.

Welcome to the Love Triangle Game! Ready for some heartbreak?

How big is your city?

5

Beginning Game

```
0 |      ,      ,      ,      ,      |
1 |      ,      ,      ,      ,      |
2 |      ,      ,      ,      ,      |
3 |      ,      ,      ,      ,      |
4 |      ,      ,      ,      ,      |
   0      1      2      3      4
```

Turn 1

Choose a coordinate:

1 1

```
0 |      ,      ,      ,      ,      |
1 |      ,  P      ,      ,      ,      |
2 |      ,      ,  L      ,      ,      |
3 |      ,      ,      ,      ,      |
4 |      ,  R      ,      ,      ,      |
   0      1      2      3      4
```

Player Distance to Love Interest: 1.414213

Rival Distance to Love Interest: 2.236067

Player wins this turn!

Turn 2

Player Score: 1.414213

Rival Score: 2.236067

Choose a coordinate:

3 4

0			,		,		,		
1			,		,		,	L	
2			,		,		,		
3			,		,		,	R	
4			,		,	P	,		
		0		1		2		3	4

Player Distance to Love Interest: 3.162277

Rival Distance to Love Interest: 2

Rival wins this round!

.
.
.

Player Total Score: 12.135968

Rival Total Score: 9.138564

Rival Wins! Tough Luck :(Go hit up tinder.

Do you want to continue playing (y/n)? y

Do you want to start in a new city (y/n)? y

Please provide the size of the new city: 3

0			,		,		
1			,		,		
2			,		,		
		0		1		2	

.
.
.

Do you want to continue playing (y/n)? n

Player Win Rate: 0.300000