CSE 1321L: Programming and Problem Solving I Lab

Assignment 6 – 100 points

Lists

What students will learn:

1) **Declaring lists**
2) **Performing common operations on lists**

Overview: Lists are an incredibly powerful thing in computing. Almost every audio file, video file,and image you've ever seen on a computer is stored in a list-like structure. A list is simply adata structure that holds a lot of values. For example, they could hold 50 integers, 100 Booleans, or a million floats.

map() : The map() function in Python is used to apply a given function to every item in an iterable (like a list or a string) and return a map object (which can be converted into a list).

Example:

```
# Convert a list of string numbers into integers
numbers = ['1', '2', '3', '4']
integer_numbers = list(map(int, numbers))  # Applies int() to each
```

   a. Pick a random number between 0 and 1. If the number is greater than or equal to 0.7 you'll add a Treasure 'T' to the next cell of the list. If the number is less than 0.7 you'll add an open 'O' to the next cell of the list.
   b. Keep track of how many treasures you are adding to the board in a separate variable called numberOfUndiscoveredTreasures.
   c. Repeat step (a) until you have a list that is the height the user asked for in step (1).

4) Repeat step (3) until the board is the width the user asked for in step 1.
5) Tell the user how many treasures you have hidden.
6) Next you'll ask the user to guess coordinates, you'll check if they found treasure or not:
   a. Ask the user to enter in a row number (0 to the width of the board -1)
   b. Ask the user to enter in a column number (0 to the height of the board -1)
   c. Check that location to see if it is a "T" (Treasure) or an "O" (Open).
      i. If it's a treasure tell the user they got treasure, change that cell of the board to an "X" to indicate that it was already discovered. Lower the number of undiscovered treasures by one.
      ii. If it's not a treasure, tell the user to try again.
      iii. Keep asking the user to guess locations until the user has discovered all the Treasures, then print out the whole board, and end the game.

Sample Input:

```
Enter dungeon width: 4
Enter dungeon height: 4
Treasures are hidden in 3 locations.


Enter row to check (0-3): 2
Enter column to check (0-3): 3
You found a treasure at (2, 3)!
Enter row to check (0-3): 1
Enter column to check (0-3): 1
No treasure found at(1, 1)
Enter row to check (0-3): 1
Enter column to check (0-3): 2
```

```
You found a treasure at (1, 2)!
```

Sample Output:

```
0 0 0 0
0 0 X 0
0 0 0 X
0 0 0 0
```