COP 3014 Honors: Spring 2017
Homework 8

Total Points: 200
Due: Tuesday 04/25/2017 11:59:59 PM

1 Objective

The purpose of this assignment is to test your familiarity with loops, two dimensional arrays and classes. You'll also need to do fundamental algorithmic error checking. These programs have to be tested on linprog before they are turned in.

Email your files `birds.cpp` and `shots.cpp` to `jayarama@cs.fsu.edu`

2 Program 1 - The Walking Red

This program makes use of loops and 2 dimensional arrays. The implementation details are left up to you.

A long time ago, in the planet of Zamenim, which is pretty far far away, a bunch of Star Trek Redshirts beam down to explore the strange new world. They encounter several strange life forms that have freaky mutations which make them look as if they just escaped from the nearest ToysRUs, such as fear inducing aquatic creatures with multiple tentacles(which they immediately repressed from memory) and giant hairy spiders that would shoot string cheese webs. During this bizarre trek, they come across a strange "warrior" in white plate armor. The warrior, who is still somewhat far, warns them that "Imperial Stormtroopers are extremely precise shots" and asks them to stay away from the shield generator, shooting some "warning shots" to let them know that he means business. The Redshirts are well aware of their expendable status, but since the "warning shots" didn't even come close, they decide to approach anyway.

They spread out to form a line and cautiously approach the Stormtrooper, who responds by opening fire. This turns out to be quite the battle. You are a journalist in the area, and you decide to take upon yourself the monumental task of documenting the result of this skirmish. You mentally divide the space between the groups into a square grid. The number of squares on one side is equal to the initial number of Redshirts. The Redshirts advance one square for every shot the Stormtrooper takes. The Stormtrooper usually misses, but if there happens to be a Redshirt in any of the 8 squares surrounding the square where the shot lands, the Redshirt dies anyway. Of course, the Redshirt dies if there is a direct hit too. You also notice that the Stormtrooper counts the initial number of Redshirts and takes an equal number of shots. The Stormtrooper wins if, after the smoke clears, all the Redshirts are dead. If even one Redshirt manages to reach the Stormtrooper, the Redshirts win.

Your first input is the initial number of Redshirtse, followed by a list of coordinates (1 1 refers to the first location), where the Stormtrooper's shots land. The first coordinate refers to the row, while the second refers to the column. Your conclusion should list who wins. If the Stormtrooper wins, all you need to print is "Stormtrooper". If the redshirts win, you need to print "Redshirts", followed by the last square where each of the Redshirts still alive are standing.

Sample Run 1

```
Enter the number of Redshirts: 4
Enter the coordinates of the 4 shots:
1 2
3 1
2 4
4 3
Stormtrooper wins.
```

Sample Run 2

```
Enter the number of Redshirts: 6
Enter the coordinates of the 6 shots:
5 6
4 3
3 2
6 1
3 4
5 2
Redshirts win. Ones at (6,4), (6,5) and (6,6) are still satnding.
```

3 Program 2 - Birdemic 3: Object Orientation

“Acclaimed” director James Nguyen has hired you to help write the script for the final film of the wildly popular Birdemic trilogy. Since he has a background in Computer Science (he was a software salesman before Hollywood came calling), he is willing to let you do this in code. You have to design 3 classes to determine how the birds behave on screen.

Specifications

- Create a class called Bird with the following elements: (5 +10 + 10 = 25 points)
  - type - string
  - wingspan - integer
  - Constructor that takes 2 parameters and initializes the data elements.
  - function called print, that prints the above details
- Create a class called GifBird that contains the following elements: (5 +10 + 10 = 25 points)
  - bird - Object of Bird class
  - willExplode - double
  - Constructor that takes 3 parameters and initializes the data elements.
  - function called print, that prints the above details
- Create a class called AnimatedGifBird that contains the following elements:(5 +10 + 10 = 25 points)
  - gifbird - Object of GifBird class
  - canTrack - character
  - Constructor that takes 4 parameters and initializes the data elements.
  - function called print, that prints the above details

- In the main function, create an array of 3 AnimatedGifBird objects. Give them values of your choice. You ca hard code these. (10 points)
- Call the print function for each of the objects to print the details.(10 points)
- You need not check for any errors.
- Please include comments wherever appropriate. (5 points)

Sample Run

```
Bird 1:
Type - Vulture
Wingspan - 10 ft
willExplode? - 0.8
canTrack? - N

Bird 2:
Type - Eagle
Wingspan - 12 ft
willExplode? - 0.76
canTrack? - Y

Bird 3:
Type - Prehistoric Raptor
Wingspan - 16 ft
willExplode? - 0.92
canTrack? - Y
```

4 General Guidelines

1. Please make sure you're only using the concepts already discussed in class. That is, please try and restrict yourself to input/output statements, variables, selection statements and loops, functions, arrays, strings, pointers, structures, files and classes.
2. Each program is worth 100 points.
3. Please make sure that you're conforming to specifications (program name, print statements, expected inputs and outputs etc.).
4. Please make sure your code is readable.
5. Please make sure you've compiled and run your program before you turn it in. Compilation errors can be quite costly.
6. The functions you have been asked to write are very standard things. Which means, you will find them on the Internet. I will be watching for this. so please do not get into trouble be copying stuff off the Internet or your friends.