CS 1410: Asteroids Part 1

Nearly everyone has played or at least heard of the famous arcade game Asteroids. But, if not, play it here. The game involves a player-controlled spaceship that can turn, accelerate forward, and shoot. There is also a variety of asteroids that move through space, potentially on a collision course with the spaceship. If a collision occurs, then the spaceship is destroyed. The objective of the game is to eliminate all of the asteroids, by successfully shooting them with the spaceship, before a devastating collision incident occurs.

Assignment

Your assignment is to recreate a simple Asteroids game using Python and Pygame. The assignment will consist of two sequential parts. For this first part, you are required to implement the following features of the game:

1. A player-controlled spaceship that can turn in either direction, to point in any direction, and accelerate forward. All three spaceship functions should be controlled by keyboard or mouse input from the user. Remember, the ship may only accelerate in the forward direction (relative to the direction that the ship is currently oriented); it may not ever accelerate in the backward direction. The spaceship should be drawn using a simple polygon that resembles a spaceship. Either use the original game for inspiration, or use your own creativity. The polygon should rotate visually on the screen according to the rotation commands given by the player. Also, if the spaceship moves off the screen, then it should instantly reappear on the opposite side of the screen, still traveling in the same direction and at the same speed. This should apply to all four sides of the screen (left/right and top/bottom).

2. A number of asteroids that move freely through space, at a random direction and speed. Asteroids do not necessarily rotate; they simply move along a straight line, in a random direction. The shape and size of each asteroid should also be randomly different. The asteroids should be drawn using a simple polygon that resembles an asteroid (i.e. a rock). Again, feel free to use the original game for inspiration. Just like the spaceship, asteroids should reappear on the opposite side of the screen when they travel off the screen.

For part 1 of the assignment, no additional functionality is required (e.g. shooting and destroying asteroids or colliding with asteroids). You will add these features in part 2. You are welcome to continue working on additional features once you complete the requirements for part 1, but it is your responsibility to complete the requirements for part 1 of the assignment first, and submit it by the due date.

For this assignment, you are required to demonstrate use of the object oriented principles inheritance, polymorphism and aggregation when designing and implementing the classes that you will use to represent the game and its various components. For instance, you might consider having a Movable class (or something similar) to implement the game logic for moving an object (and enabling it to wrap around the screen edges, etc.). Think about how you can use inheritance and polymorphism as you decide which classes you will need, and as you implement them.

If needed, you may download the Pygame starter kit by clicking here.

Hints

- Refer to the Pygame documentation to understand which parameters are necessary when calling each of the Pygame draw methods. Specifically, you should be interested in \texttt{pygame.draw} and \texttt{pygame.Rect}.
- When creating colors, use a helpful tool to determine the RGB values. Here are two good options: \url{color.adobe.com} and \url{colorpicker.com}.

Sample

An example running program:
Add-ons
- Text
- Sound
- Image