Final Project
My name: Zuobijia Sun
Project 4: Plane Shooter

Summary

What I design is a plane shooting game called Plane Shooter. There are three pages in total - prestart page, game page, and gameover page respectively.
In the prestart page, with simple animation like loading bar and moving stars, players will find a start button after the game loads successfully. Clicking the button, they will enter into the game page. In the game page, there is a removable shooter at the bottom and planes flying in the sky. Players will have 10 bullets and use the mouse to control the shooter to shoot the floating airplanes. When hitting one plane, they will get 10 points and one more bullet for reward. At the upper right corner, they can also view their remaining bullets and scores. The more points they get, the faster the planes fly, and the harder the game becomes. At this time, players can use the key “i” and “d” to increase/decrease the speed of planes. When running out of bullets, the game’s over. In the gameover page, players can view their final scores and press the key “r” to restart the game.

Instruction

1. Clicking the start button to start the game when the game loads successfully,
2. Using the mouse the control the removable shooter at the bottom.
3. Getting 10 points and 1 more bullet when hitting one plane.
4. Pressing the key “i” or “d” to increase/decrease the speed of planes.
5. Pressing the key “r” to restart the game when game’s over.

Murphy’s Law Plan

- (6/12-14) General setups: Variables, setup functions(fonts, background), draw functions(background images, properties of text, button), mouseclicked(switching pages).

  Risks:
  The start button doesn’t show on the 1st page.
  The texts doesn’t show on the proper position and without the given properties.
  It cannot switch to the next page by each clicks.

- (6/14-6/16) Creating 3 arrays and classes of shooters, bullets, and airplanes with properties such as positions, floating speed, displays, draw functions(display of three items), mouseclicked(sets the clicking area to the button).

Risks:
Planes and shooters don't show with the given properties.
Planes don't move in the sky.
The shooter doesn't move with the mouse.
When clicking on the start button, it cannot switch to the next page.

- (6/16-6/19) the setups of classes and draw functions when planes are hit, shooter fires, and bullets are shoot

  Risks:
  There are no transformations on the three items:
  Planes will not crash if being hit.
  Score and remaining bullets will not be incremented if hitting planes.
  The planes will not fly at the given speed. The speed will not increase if hitting planes.
  When the number of remaining bullets is zero, it will not switch to the gameover page.

- (6/19-6/21) Key and mousepressed setups, Living code

  Risks:
  Pressing "i" and "d" cannot increase/decrease the speed of planes.
  Pressing "r" cannot restart the game.
  Bullets cannot be fired successfully, and the number of remaining bullets doesn't change

- (6/21-23) Simple animations and Living code

  Risks:
  The music cannot play successfully
  The loading bar cannot loaded successfully
  Even though it loads successfully, the start button doesn't show under the loading bar.
  Multiple stars shown in the background don't move with the mouse.

- (6/23-6/26) Checking and executing to make sure everything work well.

  Blue Sky Plan

- (6/12-14) General setups: Variables, setup functions(fonts, background), draw functions(background image, properties of text, button), mouseclicked(switching pages).

  Ideal outcomes:
  Creating three pages:
  the prestart page with the game name and a start button
  the game page with the number of remaining bullets and score
  the gameover page with final score

It will switch pages by each clicks.

- (6/14-6/16) Creating 3 arrays and classes of shooters, bullets, and airplanes with properties such as positions, floating speed, displays, draw functions(display of three items), mouseclicked(sets the clicking area to the button).

  Ideal outcomes:
  There is a removable shooter at the bottom and floating planes on the 2nd page.
  Planes will fly from left to right. When one plane fly out of the window, there will be another plane fly from the left to right again.
  The speed of planes and bullets are constant.
  It will switch to the game page only clicking the start button.

- (6/16-6/19) the setups of classes and draw functions when planes are hit, shooter fires, and bullets are shoot

  Ideal outcomes:
  Planes will crash if being hit.
  Score will be incremented 10 and remaining bullets will be incremented 1 if hitting one plane.
  The speed of planes will also increase if hitting planes.
  When the number of remaining bullets is zero, the game's over. It will switch to the gameover page.

- (6/19-6/21) Key and mousepressed setups, Living code

  Ideal outcomes:
  Pressing "i" and "d" can increase/decrease the speed of planes.
  Pressing "r" can restart the game.
  Bullets will be fired from the bottom and the number of remaining bullets will decrease 1 each time.

- (6/21-23) Simple animations and Living code

  Ideal outcomes
  The game will have the background music.
  There is a loading bar on the first page. When it loads successfully, the start button will show under the bar.
  There are multiple stars shown in the background, and they will move with the mouse.
  Updating the comments of code.

- (6/23-6/26) Checking and executing to make sure everything work well.

User Story

As a first-person shooter, I need to find a removable shooter at the bottom and several airplanes floating in the air so that I can shoot these planes and get scores and more bullets, and thereby shoot planes as more as possible.

Technical Information

- User input
    - Creating variables - scores, remains, marks, etc
    - Importing sound files, images, shapes, text fonts
    - The size of the window
    - The display after hitting planes
    - Pressing “i” and “d” to control the floating speed of airplanes
    - Pressing “r” to reset all setups
- Class descriptions
    - Class of plane
        - Constructor for planes
        - Display of planes flying in the sky
        - Display of hitted planes
    - Class of shooter
        - Constructor for shooter
        - Display of shooter fired
    - Class of bullet
        - Constructor for bullets
        - Display when bullets are fired
        - Display when bullets shoot in the sky

Actual Project Plan

(6/12 Tuesday)
I create three 500*700 pages, which respectively are prestart page with the game name and start button, game page with score and the number of remaining bullets, and the gameover page with final scores.

(6/14 Thursday)
I create three arrays and classes of plane, shooter and bullet with constructor and displays. There is a removable shooter at the bottom, and it will move with the mouse. Planes will fly from left to right at constant speed. I also sets the click area to the start button so that it will switch to the next page when I click the button.

(6/19 Tuesday)

I use the variable mark to determine whether bullets are fired and planes are hit, and design the transformations of bullets and planes when bullets hit the planes. Finally, it works that planes will crash and the number of remaining bullets will be incremented when hitting planes. The score will also be incremented. When running out of bullets, it will switch to the gameover page.

(6/21 Thursday)
Pressing the key “i” and “r”can control the speed of planes. Pressing the key “r” can restart the game. In each clicks, bullets will be fired from the shooter, and the number of remaining bullets will also decrease each time.

(6/23 Saturday)
I add some simple animations(loading bar and moving star) and import background music. Music plays successfully. The stars shown on the 1st pages will move with the mouse. When the loading bar loads 100%, the start button will show under it.

The best part of my project

The best part is to use to the variable mark to determine whether planes are hit, bullets are fired and create the displays of each items in classes. The variable mark is also essential in the draw functions, representing in certain circumstances, they will have different displays and transformations.