Chloe Nguyen – Phuong Nguyen

COM 110 – Final Project

May 14, 2022

**PROBLEM SPECIFICATION AND DESIGN DOCUMENT**

**a) (3 points) clear statement of the project: what it is, what it does, etc… this could be very similar to the proposal, but just a final version of it**

* **Project title:** Flappy Bird Game
* **Outline of program:**

+ Import modules and pygame library

+ Initiate pygame, define the sounds that are used in the game

+ Let the player choose the difficulty level and the character

+ Create bird object with the given position and speed

+ Create pipes with spaces and speed according to the difficulty level the player chose

+ Start the game loop and the game is over when the bird is collided with the pipes or out of the window.

+ Main function will keep track of score, and player’s name and writes this information to the Leaders file.

* **Description of functionality:**

**+ What will it do?** This program will let users play Flappy Bird. It will get the user’s space click and convert it into the bird’s move. The goal of the bird is to fly as far as it can while avoiding the obstacles(pipes). Besides, the program also adds some new features which are let the player choose the difficulty and the character.

**+ How will it work?** For each successful move past the obstacle of the bird, users will get some points (1 for easy level, 2 for medium and 3 for hard). The points will continue adding until the bird fails and dies. This is an infinite game without an end point. The points therefore keep adding until the game is over. The score will be updated in the top right corner of the main window and a score board, which is included the medal and the final score, will be shown in the main window after each game.

**b) (6 points) description of the major components of the design, particularly the classes and methods (including parameters and return types)**

1. **Main program**

* File imported: Pipeclass, Birdclass, buttonClass, pygame (for sound effect only)
* Methods:
* ***styleText(text, gwin, point, size = 14, color = "black"):***

+ Helper function to create style Text. This method set the default Style of text to bold and default Face of text to courier

+ This method takes 4 parameters: the input text, gwin, point that the text located, size

+ This method creates text objects and returns the text as user input format

* ***clear(win):***

+ This sub-function helps clear the window but still keep the background image

+ This methods undraw the window except for the background and then update the new window

* ***highscore(gwin):***

+ This sub-function displays high scores

+ This method takes the Flappy Bird Leaders.txt file and convert it into the high score board

+ The .txt file is created by appending each player’s name and score to the list. Then, the high score list is sorted in descending order. The result will return 10 highest scores

* ***flappy(win, space, speed):***

+ This sub-function handles the main game

+ In this method, we import music, create bird object and its position, check user’s key and display the score

+ If the game continues, we keep creating Pipe Object and make the bird fly. We also update the score after each iteration.

+ The score is counted as 1 for each Pipe in the easy mode, 2 for medium mode and 3 for hard mode

+ This function will return the user’s name and score

* ***chooselevel(win):***

+ This sub-function handles user’s choice of difficulty

+ This method will let user type in their name and choose Easy/ Medium/ Hard mode.

+ This function will return user’s name and the difficulty they chose

* ***chooseCharacter(win):***

+ This sub-function handles user’s choice of character

+ It will return the character that the user chose (squirrel, fish, or bird)

* ***main():***

+ This is the main function of the program

+ This function will create the playground of the game, display the medal and final score afterwards, and update the high score list

+ After that, the program will redraw the buttons into: “Restart” – “High Score” – “Quit”

1. **Bird class**

* File imported: graphics, time
* Methods:
* ***\_\_init\_\_(self, point, win, speed):*** constructor method

+ This method creates the bird, set up bird’s speed and height, import the bird’s image

* ***fly(self):***

+ This method makes the bird fly by setting up the move and update the position

* ***fall(self):***

+ This method makes the bird fall by setting up the move and update the position

* ***getPos(self):***

+ This method gets the current position of the bird

+ This method sets up bottom and top position of the bird and return them

* ***main():***

+ This method creates the testing window for the flappy bird. If the user hits “space”, the bird will fly, otherwise it will fall

1. **Pipe class**

* Files imported: graphics, time, random
* Methods:
* ***\_\_init\_\_(self, xPos, yPos, win, space, speed):***

+ Constructor methods to create pipes with given space and speed

+ This method takes the center of the space between two pipes and the speed of the pipes

+ It will insert the upper and lower pipe image

* ***move(self):***

+ This method will move the pipes up and down and returns their new generated position

* ***undraw(self):***

+ This method will undraw the pipe

* ***main()***:

+ function to test the class

+ This method creates the testing window for the moving pipes. The center of the space between two pipes will be randomize in range (15,45)

+ Then it will create the default pipe object with speed = 0.03 and space = 7

1. **Button class and graphics.py**

* Normally used class to create buttons and build GUI

**c) (1 points) kinds of testing you have done and modifications you would make for a future version**

* We have tried to calculate the ymax, ymin of the pipes multiples times using the formula:

pipe\_ymin = pipes.yPos - int(space/2)- char\_height/2

pipe\_ymax = pipes.yPos + int(space/2)+ char\_height/2

The problem is that sometimes, the character still died when passing the pipes even though it shouldn’t.

Then we realized the problem is that we rounded space/2, and space are sometimes odd numbers so that’s the problem. Therefore, we decided to delete the int() as even the space is an odd number, the program can still handle .5 as we set the coordinate 1 = 10 pixels

• For a future version, we want to add coins along the way the character moves so that the player can earn some point when the bird eat the coins

**d) please also include a readme.txt file to tell me *how to run the program* (which file to run, etc) so I don’t have to try running every file to figure it out**

* File included in submission