BINUS UNIVERSITY
BINUS INTERNATIONAL

Assignment Cover Letter

(Individual Work)

Student Information:		**Surname**	**Given Names**	**Student ID Number**
	1.	**Heriyanto**	**Kevin Dimas**	**2201798156**

Course Code	**: COMP650**	**Course Name**	**: Introduction to Programming**
Class	**: L1BC**	**Name of Lecturer(s)**	: 1. Monica Hidajat
Major	**: CS**		
Title of Assignment (if any)	: Connect 4		
Type of Assignment	**: Final Project**		
Submission Pattern			
Due Date	**: 20-11-2017**	**Submission Date**	**: 20-11-2018**

The assignment should meet the below requirements.

1. Assignment (hard copy) is required to be submitted on clean paper, and (soft copy) as per lecturer's instructions.
2. Soft copy assignment also requires the signed (hardcopy) submission of this form, which automatically validates the softcopy submission.
3. The above information is complete and legible.
4. Compiled pages are firmly stapled.
5. Assignment has been copied (soft copy and hard copy) for each student ahead of the submission.

Plagiarism/Cheating

BiNus International seriously regards all forms of plagiarism, cheating and collusion as academic offenses which may result in severe penalties, including loss/drop of marks, course/class discontinuity and other possible penalties executed by the university. Please refer to the related course syllabus for further information.

Declaration of Originality

By signing this assignment, I understand, accept and consent to BiNus International terms and policy on plagiarism. Herewith I declare that the work contained in this assignment is my own work and has not been submitted for the use of assessment in another course or class, except where this has been notified and accepted in advance.

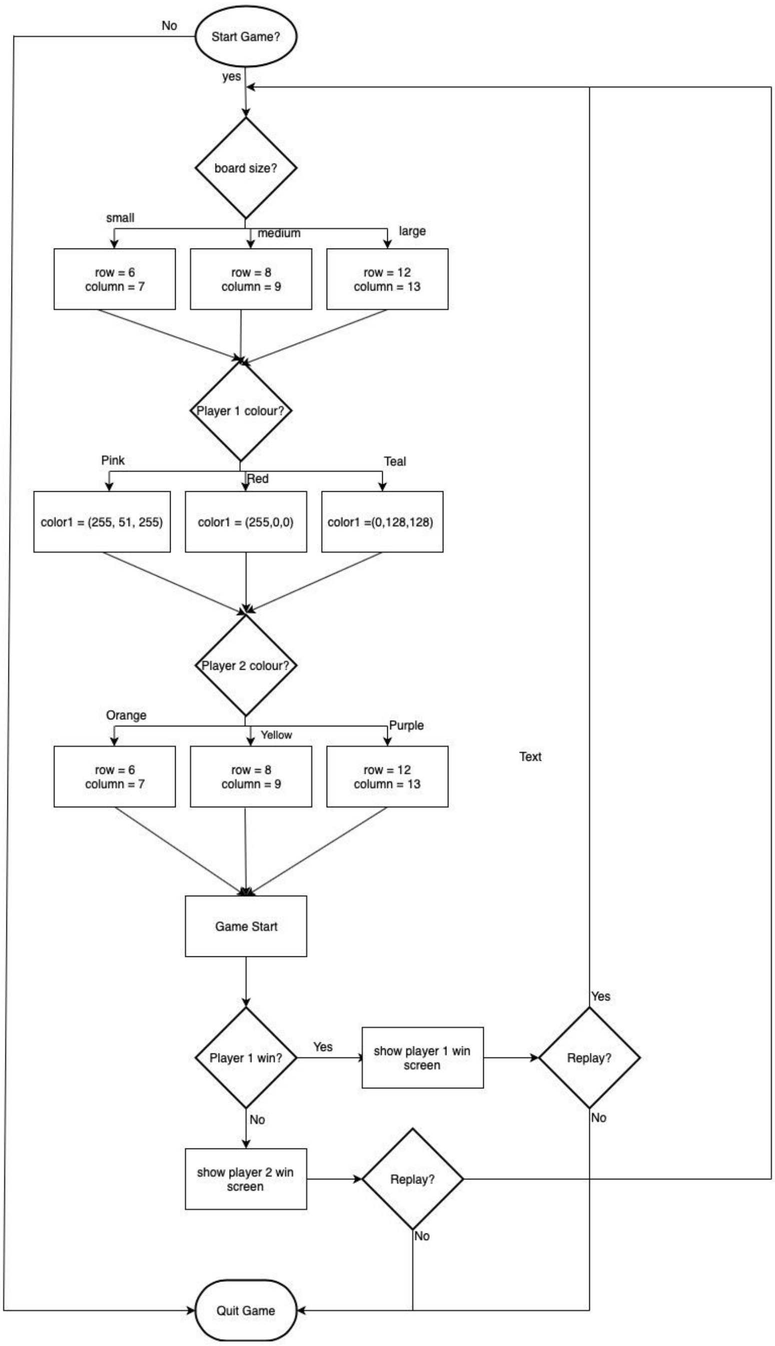
Signature of Student: (Name of Student)

1. Kevin Dimas

1. Description

   This program a computerized version of a board game called connect 4. Connect 4 is a simple board game that includes a vertically aligned board and circular pieces. The way the game works is that there will be players/sides, and each turn, said player/side would drop a piece into the board, and the first player/side to align 4 pieces horizontally, vertically, or diagonally would win the game. Some aspects I added to the program that wouldn't traditionally be available are the selection of board size and the selection of the piece's color for each player.

2. Design



3. Dicsussion:

The things that were included in the program were PyGame, NumPy, and Math.

PyGame is used as the main tool in order to create the program. PyGame is responsible for the creation of the graphics and functions in the program. This includes the board, the circular pieces, and the menu of the program. PyGame also makes the program able to produce sound i.e. when a button is pressed, and the in-game background music.

Numpy is used to make the adjustments in the cursor x value so that when divided by the size of the each "SQUARESIZE" so that the program knows which row the circular pieces need to be placed.

Math is used to make the initial "board" (can be seen in the repl) to have values "0" to be replaced by the values of the players (1/2).

4. How it works

As soon as the program starts, you would be greeted by the start screen, giving you the option to start the game or simply quit the program. If you choose press enter on the "quit" option, you will quit the game. However, if you choose the "start" option, you will be headed to the next option, which is the choosing the size of the board.

The first option is to choose the size of the board. There are 3 options, small, medium, and large. The small board is a 6 x 7 board, the medium is a 8 x 9 board, and the large board is 12 x 13 in size.

After choosing the size of the board, each player will take turns choosing their circular piece's colors. The first player gets to choose between the colors pink, red, and teal, while the second player gets the option to have the colors orange, yellow, and purple.

If both players are done with choosing their respective colors, the game will start. The game begins with the first player choosing the place where said player wants their circular pieces to drop in. The player would hover the cursor in the position of the board, hovering from left to right corresponding to the position of which row the player wants the piece to drop in.

When the player decides which row they want the piece to drop in, they would perform a left click action on their mouse to drop the piece into the board. Then, the next player would do the same thing taking turns between the players.

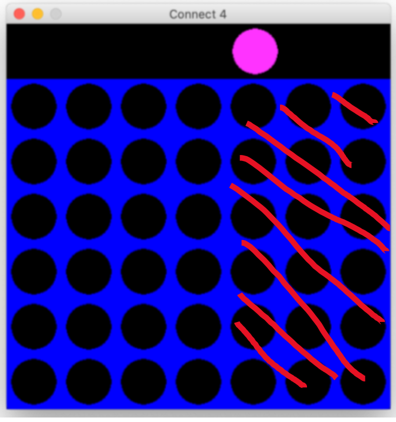
The game will end when a player manages to have four pieces in a row in a horizontal, vertical, or diagonal position. After the game ends, they would have to choose between replaying or quitting the game. If the decide to press the “replay” button, the players would be directed to the size-choosing menu, while if you choose the “quit” menu, the program will exit and quit.

5. Winning Algorithm

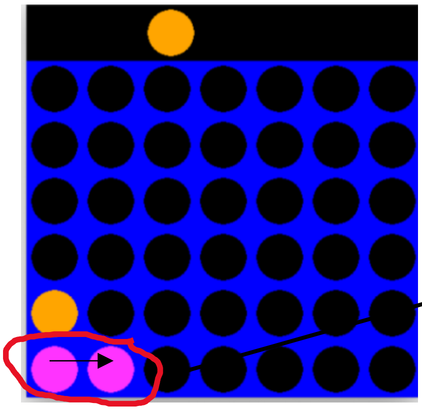
To win the game, one of the players needs to have a sequence of four circular pieces in a row whether it being in a horizontal, vertical, and diagonal position. The game has an algorithm built in to check for such sequences.

There are 4 validation processes in which the game checks for after every single move is made.

a. Vertical win

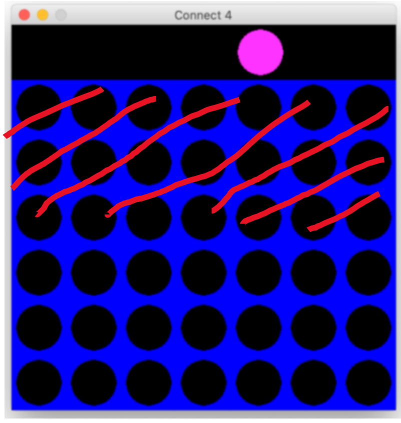


It checks for row – 3 because there can’t be any sequence of 4 circular pieces in a vertical position starting from

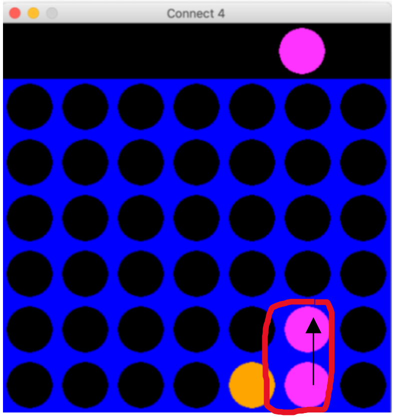


The algorithm checks for vertical winning by checking from the left to the right of the board

b. Horizontal win

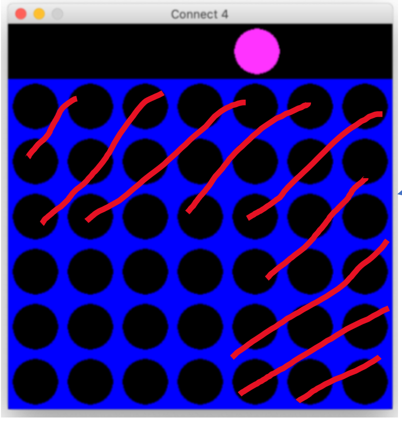


It checks for column – 3 because there can’t be any sequence of 4 circular pieces in a horizontal position starting from that point

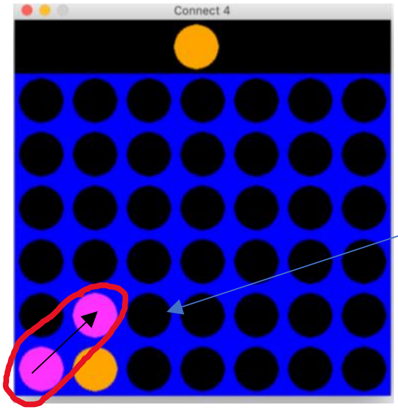


The algorithm checks for horizontal winning by checking from the bottom to the top of the board

c. Right diagonal win

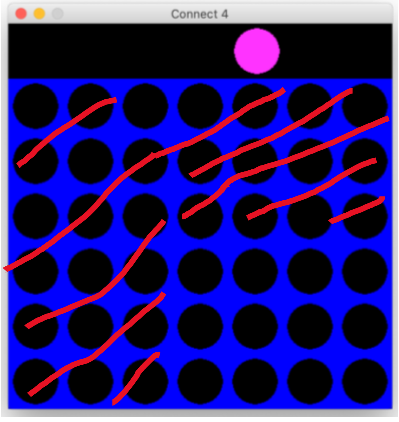


It checks for column – 3 and row – 3 because there can’t be any sequence of 4 circular pieces in a right diagonal starting from that point

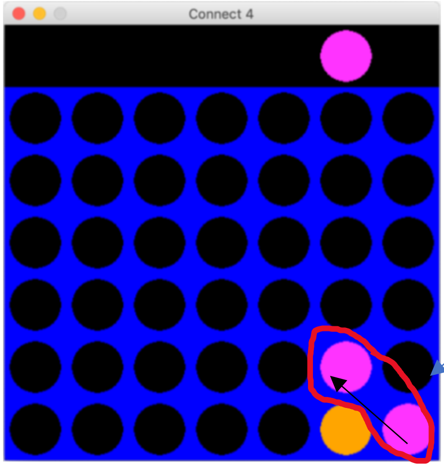


After that, it checks for 4 sequence in a right diagonal position starting from the bottom left to the top right

d. Left horizontal win



It checks for column – 3 and row – 3 because there can’t be any sequence of 4 circular pieces in a right diagonal starting from that point



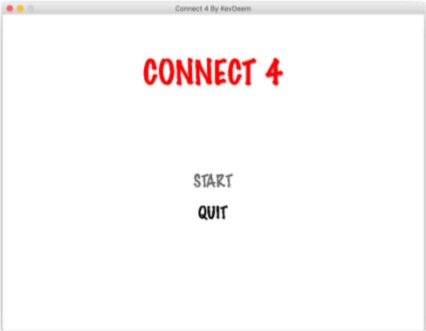
After that, it checks for 4 sequence in a right diagonal position starting from the bottom left to the top right

6. Class Explanation

The config class in the program is used to store the color RGB values, the screen size values, and the sounds of the game.

The author realizes that this is the wrong implementation of a class, and there could be more substances in the class that could make it more useful for the program.

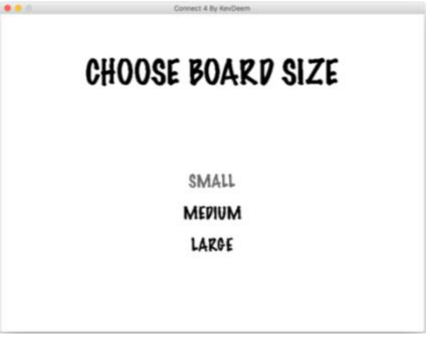
7. Evidence



CONNECT 4

START

QUIT

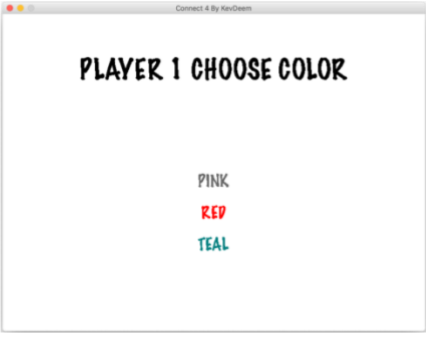


CHOOSE BOARD SIZE

SMALL

MEDIUM

LARGE

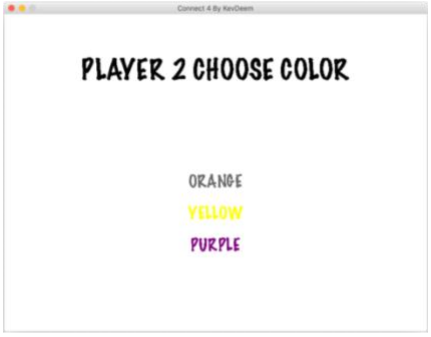


PLAYER 1 CHOOSE COLOR

PINK

RED

TEAL

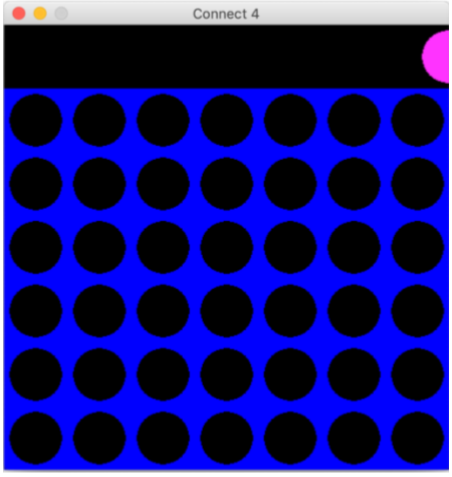


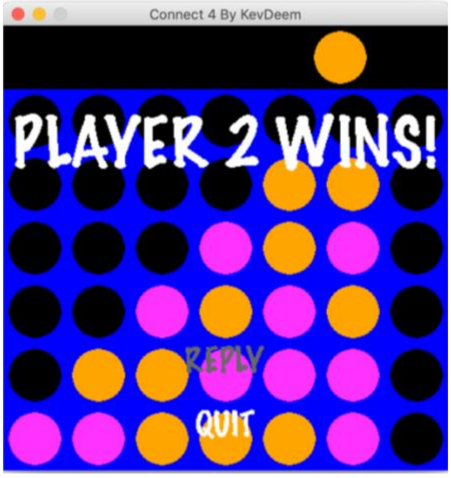
PLAYER 2 CHOOSE COLOR

ORANGE

YELLOW

PURPLE





8. Resources

Free source code, tutorials and articles. (2018). *Python – Pygame Simple Main Menu Selection.* [online] Available at: https://www.sourcecodester.com/tutorials/python/11784/python-pygame-simple-main-menu-selection.html [Accessed 17 Nov. 2018].

Galli, K. (2018). *Connect 4 Game Python - YouTube*. [online] YouTube. Available at: https://www.youtube.com/playlist?list=PLFCB5Dp81iNV_inzM-R9AKkZZlePCZdtV [Accessed 17 Nov. 2018].