Krish Kalla
Mrs. Brodsky
CSC 590 – G.A. 2
3 April 2023

1. What problems are you currently pondering/working through? If you feel near finished, is there some aspect of your program that you could make more efficient or accurate?

   Earlier, during class on Friday, I had the issue where I was unable to get any mouse-clicks registered through the program. If I hovered over a column and clicked, the program did not register the mouse click, and there was no update on the interface. That problem has been resolved now, and currently, I am facing the issue where the program is working, but the visual is reversed, where the program acts as if the bottom of the board is the top and vice versa. The logic works completely, but the visuals are flipped 180 degrees.

2. In groups of 2 or 3, talk through the problems that you and others have. What solutions might you have come up with? (These are of courses not tested yet, but you will try them. Feel free to include pseudo code or just a description of the idea.)

   On Friday, Jonny, Edie, and Miss Brodsky helped me diagnose the issue, giving me the idea to put print statements in the code to check whether the problem was truly no register of the mouse. After that was confirmed, I was given the idea to try and use pygame's inbuilt function with mouse buttons by Ethan, as he is also using python.

3. Who were your teammates and what problems are they working through? Were you able to help? How? Did thinking about their problems also help your approach?

   My teammates were Jonny and Edie. I helped Jonny fix his problem where his coin colors were rapidly changing in the same box when the mouse button was held down by giving him the idea of using a checker to see whether the place the coin is trying to be placed is open, and if it is not, then move to the next box up. I also gave him the idea of using nested if statements where if the mouse button was clicked down, then check once it is clicked up, before exiting the if statement and placing the coin. This, however, was a less-useful method as java has a method that checks this same thing inbuilt. I also helped Edie by giving the idea of using hidden buttons behind the columns to act as the way to get an input from the mouse while it is clicking on a column. This also gave me ideas for my project, as it gave me the idea of using hidden position boxes to act as the way to place my coins.

Krish Kalla
Mrs. Brodsky
CSC 590 – G.A. 2
3 April 2023

4. Stage 1: Planning and Requirements Gathering – Please include all that you have for this state. You likely submitted this last weekend; copy-and-pasting that exact submission is expected. Do not add more now.

   Customer – Michael Ngai
   Language: Python using VS Code*

   * is Additional Requirements by Me

   Requirements:

   - Menu Screen with start, game history, and settings to change colors of “coins” and background – tentative based on time constraints*
   - Start – Use “S” key to start the Game
   - Pause & Resume – Use “Space” key to Pause and Resume toe Game*
   - End/Forfeit - Use “F” key to End the Game*
   - Black Holder/Board (Background - Default)
   - Red & Blue “Coins” (Default)
   - Display at top to show who’s turn it is*
   - Restart Button - Use “R” key to reset*
   - Scoreboard
     - Simple Scoreboard Table will be visible at top Right Hand Corner of Screen*
   - Game History - Use “H” key to display game history, with a small picture of the end result of each game - scroll through past games - automatically pauses the current game if in one*
   - Click to Place “Coin” in a specific place*
   - 1 vs. 1 game - two player

Krish Kalla
Mrs. Brodsky
CSC 590 – G.A. 2
3 April 2023

5. Stage 2: System Design – Please include all that you have for this phase, too. Do not add more now. This might be in the form of bullet points, sketches, and/or pseudo code

   Pseudo Code:

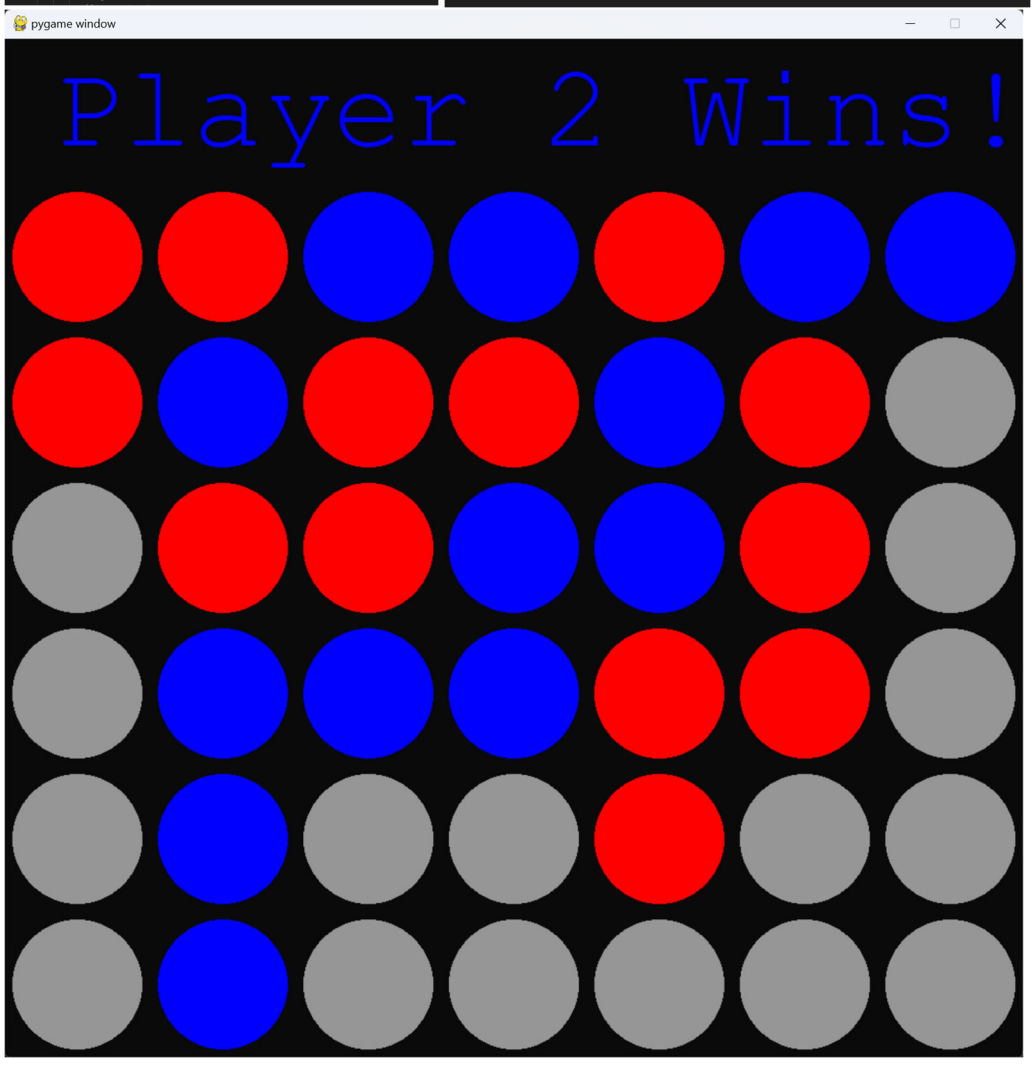
   - Create Board
   - Render Board
   - While Game is Running
     - Player 1 Move
       - Check if Valid Drop
         - Drop Piece
       - Check If Winning Move
         - Win Game – End While Loop
         - Else Continue
     - Player 2 Move
       - Check if Valid Drop
         - Drop Piece
       - Check if Winning Move
         - Win Game – End While Loop
         - Else Continue
   - Restart While Loop
   - Who has won?
     - If Player 1 – Add +1 to score
     - If Player 2 – Add +1 to score
   - Play Again?
     - If Yes – restart
     - If No – Quit and Close Window

6. Stage 3: Development – I do not ned to see all of your code at this point. However, I'd like to see your progress. Please include screenshots of what your game currently looks like. Include as many as you wish, showing as much functionality as possible.

Krish Kalla
Mrs. Brodsky
CSC 590 – G.A. 2
3 April 2023

```python
def renderBoard(BOARD):
    for c in range(board.columnCount):
        for r in range (board.rowCount):
            pg.draw.rect(window, BLACK, (c*SIZE,r*SIZE + SIZE, SIZE, SIZE))
            pg.draw.circle(window, GRAY, (int(c*SIZE+SIZE/2), int(r*SIZE+SIZE+SIZE/2)), RADIUS)

    for c in range(board.columnCount):
        for r in range(board.rowCount):
            if BOARD[r][c] == board.p1:
                pg.draw.circle(window, RED, (int(c*SIZE+SIZE/2), int(r*SIZE+SIZE+SIZE/2)), RADIUS)
            if BOARD[r][c] == board.p2:
                pg.draw.circle(window, BLUE, (int(c*SIZE+SIZE/2), int(r*SIZE+SIZE+SIZE/2)), RADIUS)

    pg.display.update()

BOARD = board.createBoard()
board.printBoard(BOARD)
isOver = False
turn = board.p2

pg.init()

SIZE = 150
RADIUS = int(SIZE/2-8)

width = board.columnCount * SIZE
height = (board.rowCount+1)*SIZE
windowSize = (width, height)

window = pg.display.set_mode(windowSize)
renderBoard(BOARD)
pg.display.update()

font = pg.font.SysFont('monospace', 120)

while not isOver:

    for event in pg.event.get():
        if event.type == pg.QUIT:
            sys.exit()

        if event.type == pg.MOUSEMOTION:
            pg.draw.rect(window, BLACK, (0,0, width, SIZE))
            x = event.pos[0]
            if turn == board.p1:
                pg.draw.circle(window, BLUE, (x, int(SIZE/2)), RADIUS)

            if turn == board.p2:
                pg.draw.circle(window, RED, (x, int(SIZE/2)), RADIUS)

        pg.display.update()

        if event.type == pg.MOUSEBUTTONDOWN:
            pg.draw.rect(window, BLACK, (0,0, width, SIZE))

            if turn != 0:
                if turn == board.p1:
                    turn = board.p2

                elif turn == board.p2:
                    turn = board.p1

                #Player 1 Move
```

```python
import numpy as np
import math

gameOver = False
turn = 1

empty = 0
p1 = 1
p2 = 2

rowCount = 6
columnCount = 7

def createBoard():
    board = np.zeros((rowCount, columnCount))
    return board

def dropPiece(board, r, c, piece):
    board[r][c] = piece

def isValid(board, c):
    return board[rowCount - 1][c] == empty

def getOpenRow(board, c):
    for r in (range(rowCount)):
        if board[r][c] == empty:
            return r

def printBoard(board):
    print(np.flip(board, 0))

def winningMove(board, piece):
    #Check Horizontal Win
    for c in range (columnCount - 3):
        for r in range(rowCount):
            if board[r][c] == piece and board[r][c+1] == piece and board[r][c+2] == piece and board[r][c+3] == piece:
                return True

    #Check Vertical Win
    for c in range(columnCount):
        for r in range(rowCount - 3):
            if board[r][c] == piece and board[r+1][c] == piece and board[r+2][c] == piece and board[r+3][c] == piece:
                return True

    #Check Diagonal Win(s)
    for c in range(columnCount - 3):
        for r in range(rowCount - 3):
            if board[r][c] == piece and board[r+1][c+1] == piece and board[r+2][c+2] == piece and board[r+3][c+3] == piece:
                return True

    for c in range(columnCount - 3):
        for r in range(3, rowCount):
            if board[r][c] == piece and board[r-1][c+1] == piece and board[r-2][c+2] == piece and board[r-3][c+3] == piece:
                return True
```



Krish Kalla
Mrs. Brodsky
CSC 590 – G.A. 2
3 April 2023

7. State 4: Testing – What is your plan for the testing phase? How have you tested so far? What will you do to test your final product?

   Currently, I am performing an Open Beta testing, having people in my dorm play the game, albeit being oriented upside down. I will continue to use this to help check for any logic errors, while still getting multiple people's inputs on the game and if there are any other errors they think should be fixed. For the Final Products, I will also use this method, as most companies that do not mind their program coming out into the wild find it the most useful way of testing due to the amount of tests that can be run at the same time by others while work can still be performed on the program.

8. Reflection – Reflect on this SDLC process. How has it gone for you? What SDLC model do you think this followed? Why? Should you have put more or less time into any phase(s)?

   I believe this was the waterfall method we had talked about earlier due to its linear style of working. Each phase had its own time frame and continued to flow through each phase without going backwards. This has gone well, and I have found myself to be quite productive following this model. However, I would say I should have put more time into the programming phase, as I would like to make everything more efficient before turning it in, as currently it seems slightly bloated and there are some if, elif, else, and while loops in the program that could be avoided. But overall, this was very useful and went well for me.