Comprehensive Exercise: Connect 4

Group Name:
Section 642, Team 02

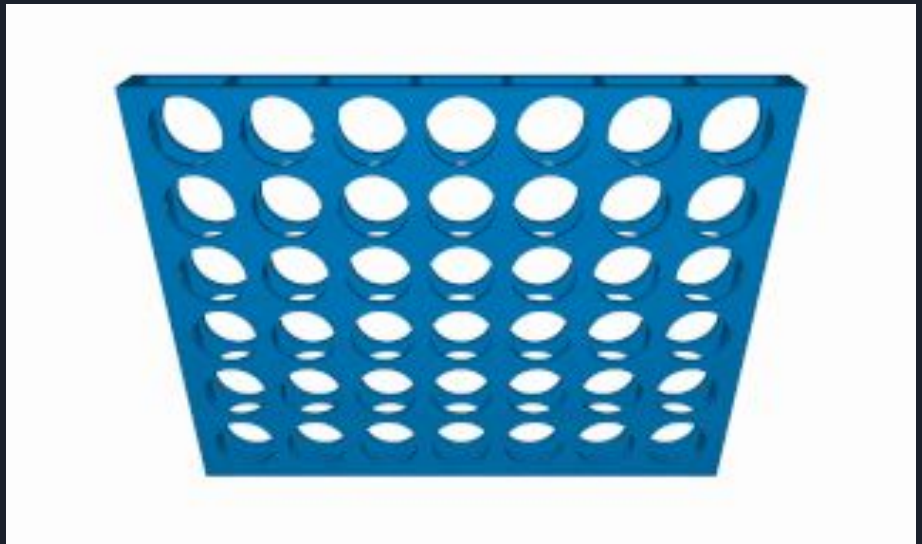
Team Members:
Mustafa Irfan, Mukul Sauhta, Raven Llarina, Ryan Foster

What is Connect 4?

Connect Four is a popular two-player connection game in which the players first select a color then taking turns dropping one colored disc from the top into an eight-column, eight-row vertically hanging grid. The pieces fall straight down, taking up the smallest amount of space in the column. The objective of the Connect Four game is to be the first player to build a horizontal, vertical, or diagonal line using four of one's own discs.

Requirements/Analysis

This program is a Connect Four game that allows 2 players to play against each other from the same computer. The Connect Four match can be won horizontally, vertically, or diagonally. It will be a grid that contains 64 spots. Each user will be a different color, and they choose their spots by clicking an empty spot on the grid. There will be a system where the users cannot place above an empty spot and their chip will fall down to the closest bottom spot.



Requirements/Analysis cont.

Software Requirements:

- ❏ Program must be able to identify a user victory vertically
- ❏ Program must be able to identify a user victory diagonally
- ❏ Program must be able to identify a user victory horizontally
- ❏ Program must be able to identify a draw
- ❏ Program must be able to exit after each game
- ❏ Program must be able to allow forfeits through closing the game
- ❏ Program must be able to identify a full row
- ❏ Program must be able to identify a full column

Requirements/Analysis cont.

Software Requirements:

- ❏ Once game is started, Player 1 is prompted to pick spot first
- ❏ Each player will rotate turns after every click
- ❏ Pieces placed in a spot where there is empty space below will fall down
- ❏ Once a game is won, a window will appear with players color and a “You Win”, and the game will restart

Design

Design One:

- Spot Class: color
- ConnectFourGUI: MINCONNECT, panel, frame, connectGrid, spots, connectFourGrid, redIcon, yellowIcon, emptyIcon, piecesPlayed, playerOneWins, playerTwoWins, inARow

We plan to use design one. We want to use design one because it is a lot more descriptive and in-depth in comparison to design two.

Pros	Cons
More specific, easier to understand	More to code, might take longer
Will be easier to find bugs due to specifics	Might be unnecessary to have to many repeating classes for the different colors

Design cont.

Design Two:

- Yellow Class: X-coord, Y-coord, Turn, Location
- Red Class: X-coord, Y-coord, Turn, Location
- Grid Class: BlankSpace, FilledSpace, ClickedSpot

Pros	Cons
Less parts, making it easier to code and review	Will be harder to find bugs
Less to code, won't take as long as Design One	Errors in one class might lead to multiple errors in others

Demo of Implementation

Methods needed to Implement our Design:

- For Loops: Needed to iterate through the grid and determine who the winner is, and keep track of the maximum chips in a row.
- Calling Methods: We will need to utilize calling methods from other methods in order to maximize efficiency of code, and not create replication of code.
- If/Else Statements: To check conditions and make sure that boundaries are met/not met. They will also be useful for determining the winner of the game.
- Knowledge of objects also helps greatly.

Testing

- The testing phase is defined as the time in the software life cycle when the components of a software product are reviewed and integrated, and the software product is evaluated to determine whether or not requirements have been met.
- The phase where we thoroughly tested our Connect 4 program
- Tested for:
  - A diagonal win by the Red player
  - A draw (The entire grid is filled without having 4 chips of the same color in a row, horizontally or diagonally)
  - Red player wins horizontally in second row
  - Yellow player wins vertically in fourth column
  - Red player more than 4 connects in the bottom row
  - Test full column (space already filled!)

Lessons Learned

Key Takeaways:

- For loops were really important and were used in every method that iterated through the grid. For loops were especially useful when we want to set a cell to empty red or yellow
- Try catch was also useful in determining when a cell was already filled
- Splitting up a GUI into multiple sources is way more efficient than making one big GUI