Tony. J and
Adish. S

Step 1: Proposal

Overview

Provide a description of what product(s) you intend to create in this project:

A computer version of the classic Connect 4 game but the user gets to play against bot Adish while also having the ability to change Adish's ability to play Connect 4 if Adish is beating the user too often.

Objective:

Provide a description of what skills and understandings you intend to gain in completing this project:

This project will be an introduction to creating algorithms as Adish (the bot) uses math to compute the optimal slot to insert the chess piece which is the foundation to both machine learning and artificial intelligence. Although Python is the optimal programming language for artificial intelligence, using Java will expand our skills in creating AI in different languages. This is the first game that both of us has created so it will be an eye opening experience towards programming an enjoyable game that can be a great way to pass time.

Connections:

Describe how this project connects with two other CTS courses that you have taken:

The algorithm for where Adish is going to place the game pieces is very similar to The Game of Life program where it checks for empty and filled slots and calculates the optimal slot to insert the game piece by calculating the next few steps (that could be changed with difficulty) and assigning an integer value to how good the position is. The GUI is similar to Bank Account GUI as it features event handlers and basic GUI design.

Resources:

Provide a list of resources, outside of what is available in class, that you will require to complete this project:

- YouTube Videos
- Reliable/Well-Known Websites (Ex. StackOverflow)

Risks:

Identify any risks to health and/or safety, and how you intend to mitigate these:

Eye strain - Pomodoro technique (timed 30 minutes of work with 10 minutes of rest)

Indicators for Success:

How will you know that you have achieved your objectives? Provide a rubric that describes what a successful project looks like:

1.   On the left-hand side, list the various components (i.e. parts) of your product (e.g. if you are writing a game, list the GUI, Game Play, Scoring, Sounds, etc). On the right-hand side, describe what these elements *should* look like for this product to receive a Mastered.

2.   Then, also consider the process. What does good work look like? What would you look for in a teammate?
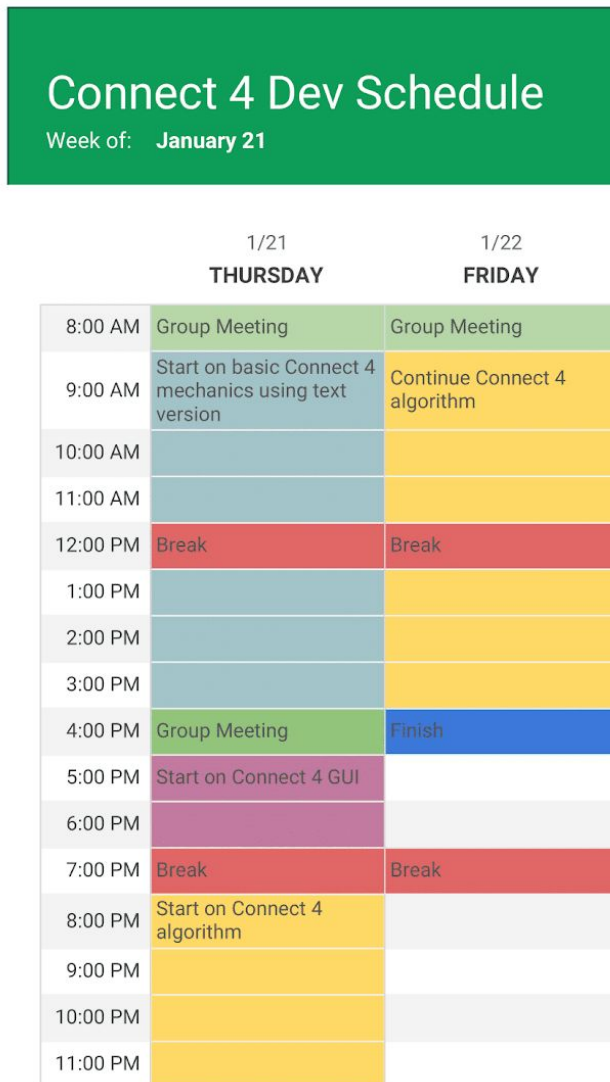
Component	Mastered (5/5)
Game mechanics	- Winner declared when 4 game pieces are aligned. - Tie declared when the board is filled with no winner. - No false winner declarations - Game pieces go into user intended slot
Bot Adish	- Difficulty can be changed - Calculates future moves (difficulty 1 = 1 future step, difficulty 5 = 5 future steps) - Does not place random pieces - Values assigned to each slot can be seen in the terminal (visible calculations) - Bot can win
GUI	- Simple and intuitive GUI - Enjoyable - All buttons are functional - Game pieces and board look like that of a normal Connect 4 - Playing our version will be straight forward for a user with previous experience in playing Connect 4 - Able to input user name

Tony. J and Adish. S

Code Management	Code is commented and easily readable. Code is also aligned with proper formatting and doesn't include random spaces that would make the code confusing to the reader.
Time Management	2 day time frame to finish project
Problem Solving	Asking for help online (Stack Overflow) due to the tight time frame.

Tony. J and
Adish. S

Planning

Gant Chart

Break your project into distinct phases. Then, provide estimates for how long each phase will take. You have already completed the first phase, which is to create a proposal. You are currently in the second phase, which is planning.

Connect 4 Dev Schedule
Week of: **January 21**

	1/21 THURSDAY	1/22 FRIDAY
8:00 AM	Group Meeting	Group Meeting
9:00 AM	Start on basic Connect 4 mechanics using text version	Continue Connect 4 algorithm
10:00 AM		
11:00 AM		
12:00 PM	Break	Break
1:00 PM		
2:00 PM		
3:00 PM		
4:00 PM	Group Meeting	Finish
5:00 PM	Start on Connect 4 GUI	
6:00 PM		
7:00 PM	Break	Break
8:00 PM	Start on Connect 4 algorithm	
9:00 PM		
10:00 PM		
11:00 PM		

Design

Consider your IPO.

·   What are the inputs into your program?

·   What processing needs to be done? i.e. which algorithms do you need?

·   What are the Outputs

For graphical projects, consider what your project would *look* like. How does it interact with the user? Provide a series of mock-ups that clearly show the flow of execution. These can be either drawn or designed using WindowBuilder.

For console-based projects, provide a "script" for how the program would accept input and provide output. These can be text-based but should be thorough. Consider cases (and show examples of) where the user provides improper input. Also, provide an explanation/flowchart/pseudocode of the algorithm(s) that you are using to process the input.

Answer:

How our game works is that the user will be playing a connect4 game against an AI bot (which is Adish in this case). The user will input their name and will choose the difficulty of the AI bot depending on their skill level from 1 to 5. After the user enters their name and difficulty, the program will initiate by showing a picture of Adish saying "Bring it on bud"(output of the program). Then the gameboard will pop up allowing the user to click on a series of buttons at the bottom of the screen to choose where they want to place their game pieces, allowing the game to initiate (input). After the user has chosen where they want to place their game piece, the AI bot will process a bunch of algorithms like "for loops" and "if-else statements" to make his turn as well based on the difficulty that the user has selected (output). Once the game is finished, when someone has connected 4 game pieces together either horizontally, vertically, or diagonally, the system will show a pop message stating that the winner has won the game (output). If the real player won the game, a picture of the AI bot(Adish) will pop up saying "Sorry I am bad at Connect4" otherwise if the AI bot won, then it will say "Get Rekt Loser"(output based on the winning input of the game).

As for the mock-up, we made a short video that shows how the program runs from our computer.

Journal

Provide a journal for the time that you worked on this project. You do not need to provide an entry for each period or session that you worked on the project, but your entries should cover all time spent. This is not meant to be a simple accounting of your time. Rather, this is a living document that records the process of building your projects and demonstrates your learning. Entries can contain your thoughts on

- Work completed
- Victories achieved
  - getting something to work
  - understanding a concept
- Challenges encountered
- Resources used
- Meta-cognition
  - What seems to be working well

Entry 1 - 9:23am 21/01/2021
- Game Constructor not working
- Asked question on StackOverflow

Entry 2 - 9:43am 21/01/2021
- Reply from StackOverflow to do "board = new char [ROWS] [COLS]" instead of "board = new char [] []" as we need to declare the two variables to replicate the board inorder for the game to function correctly.

Entry 3 - 9:32am 21/01/2021
- Game Pieces too small
  - Fixed by math.offset to correct for screen size and resolution

Entry 4 - 11:26am 21/01/2021
- Basic game mechanics seem to be functioning after importing .jar file

Entry 5 - 5:36pm 21/01/2021
- Connect 4 Text surprisingly easy to code

Entry 6 - 8:05am 22/01/2021
- Researched creating basic algorithms (ex: linear algebra and derived multi variable calc (that was hard))

Entry 7 - 9:59am 22/01/2021
- Chose game tree and alpha and beta pruning for algorithm style by suggestion on Reddit and Stack Overflow for simplest algorithm to create
- Adish left...to school

Entry 8 - 12:10pm 22/01/2021
- Finished algorithm for column and row consideration; starting diagonal
- Bot now prints assigned value for strategic value for each slot (larger int = better strategic slot)

Entry 9 - 1:10pm 22/01/2021
- Realized needed to function to store the number of moves carried out by the player to incorporate into the algorithm as the bot does not realize how many pieces are on the board and may start placing game pieces in already occupied spaces.

Entry 10 - 4:10pm 22/01/2021
- Fully functional Adish bot (woooooo hoooooooo BEST DAY OF OUR LIVES!!)
- Started working on GUI and putting memes in

Entry 11 - 7:23pm 22/01/2021
- Good enough
- We are done!!!!

Resources Used Throughout the Project:
- StackOverflow - for asking questions to a community forum
- Discord - used for communication
- GitHub - used to push and pull new commits and changes made to project
- Youtube - watched videos about certain concepts like JAR files for example
- VS-Code - environment used to make the game

Reflection Questions

These are to be completed along with your final submission of your code.

1. Provide an updated Gantt chart

Connect 4 Dev Schedule
Week of: **January 21**

	1/21 THURSDAY	1/22 FRIDAY
8:00 AM	Grouping meeting and decision making	Tony & Adish both work on algorithm
8:30 AM		
9:00 AM	Tony- started programming function to check whether slots are occupied. Adish- started programming game pieces	
9:30 AM		
10:00 AM		
10:30 AM		
11:00 AM		
11:30 AM		
12:00 PM	Break	
12:30 PM		
1:00 PM	Tony - started programming ArrayLists to house game pieces and Connect4State.java. Adish- started programming Connect4Text.java	
1:30 PM		
2:00 PM		
2:30 PM		
3:00 PM		
3:30 PM		
4:00 PM		
4:30 PM	Unproductive time	

5:00 PM		Break
5:30 PM		
6:00 PM		Tony & Adish work on GUI
6:30 PM	Tony- Started Connect4.java. Adish- Started algorithm for Connect4 bot	
7:30 PM		
8:00 PM		
8:30 PM		
9:00 PM		
9:30 PM		
10:00 PM		
10:30 PM		
11:00 PM		

2. Refer to your original specification. How does the planned duration of each task compare with the actual time it took to complete?

The Algorithm turned out to be more complicated than expected to more time was dedicated to it in order to meet the deadline, otherwise, the other component took the time we expected it to take and the GUI even seemed to take less time as our familiarity with GUI design from BankAccount GUI and Triangle Classifier.

3. Did you need to deviate from your original specifications? If so, describe which and why.

We did have to deviate from our original proposal about making the chrome extension to making this connect4 game instead. Although our work on the extension was successful and was working, we believed that we wouldn't be done with the entire extension given the time we had. We will still continue to work on the extension during our spare time and see if we can actually publish it on the chrome store but for now, that project is halted and we made this new project instead.

4.   What would you change about your process (i.e. how you approached the creation of your product)?

Using a more user-friendly IDE would probably be something that we would change about the creation of our game as we used VS-Code. VS-Code is one of the most powerful IDE's out there but you need to add a lot of specific directories and extensions to actually work on it and make project files. This took a lot of time to do and we could have potentially saved this time if we worked on another IDE like IntelliJ for example because IntelliJ is meant for java and we don't need to worry about adding any extensions or other directories.

5.   What worked well in your process?

Once we set up our project on VS-Code, everything else for the most part went smoothly coding wise. Maintaining humor within our project made it very enjoyable and almost made the last 2 days feel like playing a game as the time flew by with the constant voice chat between Adish and I on Discord; communication was also key in meeting the deadline as collaborating on a project from start to finish at home poses its own challenges.

6.   What part of the finished product are you most proud of, and why?

The part where we added in the AI Bot that resembles Adish was the achievement we were most proud of because that by far took the longest time to code as we had to program many algorithms to actually make the AI play the game as it is versing the real player.

Tony. J and
Adish. S

7.   If you were to create version 2.0 of your product, what part of the product would you develop further?

We would add a menu screen that gives the user the option to change some game settings if they wish to do so and we would also add a play button as part of the menu to let users know that the game has started. Also adding in an instructions page would be a good development as well to let the user know how to actually play the game if they don't know how connect4 works.

8.   Are there any specific bugs or functional shortcomings in your product? Be honest – being aware of these and recording them is a good thing!

When the AI selects where it wants to place the game piece after the user has finished their turn, sometimes what would happen is that the game pieces would start to flicker all over the place because the program is calculating where the next game piece should be placed as it tries all the combinations to get the one that will be a step closer to winning the game; this might be seen as it is bugging at sometimes the calculations can take a few minutes due to large amount combination available. Utilizing multi-core processing would significantly reduce the calculation time or even let the calculation become instant but dividing the workload onto multiple cores is beyond the skill level of both of us at this current stage.

9.   Which specific skills and understandings did you gain while working on this project?

So, our program was completely based on the concepts that we learned throughout the course of the semester as our game was entirely based on Java. So, most of the skills and understanding were already learned and served as prerequisites for building this project like making if else statements, for loops, etc. Inserting images however to our code was one new concept that we found was really cool and was very interesting. We implemented this insertion to our program to make our game more humorous for the users that play it. One skill that was new to us and we developed was adding in a JAR(Java ARchive) file in our project. We did give credit to a person named Delos Chang for this but this method allowed us to make a package file that included many Java classes in it which was associated with the game resources and images like the game frame and the graphical view of the game including the game board as well.