# 1. Proposal:

## Overview:

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

|  |
| --- |
| I want to create a chess AI game / program. The features / products I want to create are a chess board, chess pieces, moving pieces, possible move logic, special move logic, green shade on possible tiles, and the logic for a simplish chess bot you can play against. This project is rather ambitious but I think it’s doable in the next month / few weeks. |

## Objective:

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

|  |
| --- |
| Completing this project would improve my skills at creating larger more ambitious projects. Specifically, it would build my skills in more complex data structures and algorithms like linked lists and possible move and depth searching algorithms. |

## Connections:

Describe how this project connects with two other CTS courses that you have taken (these can be specific one-credit courses within Computer Science):

|  |
| --- |
| The project will implement most key elements of computer science 30. The project will very likely use things like recursion, algorithms, linked lists, and potentially others along the way. Recursion is very necessary for creating a chess AI / move decision algorithm as the algorithm will recursively call itself to go deeper in searching for moves. I think using a linked list would be a good way to store certain types of data in the project, for example, I could store the pieces in a linked to improve efficiency since they can remove elements faster. For algorithms, the 2 main algorithms would be a possible move checker and an optimal move algorithm for the chess bot. |

## Resources:

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

|  |
| --- |
| I think I would mostly use google, stack overflow, YouTube, and some java graphics libraries. |

# 2. Design

## Visual

Provide a storyboard for your product. For console-based projects, provide a “script” for how the program would accept input and provide output. You can either create a mock-up (very primitive version of product) and paste screen captures, or you can hand-draw.

|  |  |
| --- | --- |
| **Not that much to write here as it’s chess but here is what I have so far:** | **Also, after determining possible moves I want to redraw the board to color the possible move tiles a shade of green to make it look nicer. (obviously not all tiles would be green)** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

## Computation

What are the major computational challenges that you need to solve? Provide any required algorithms and/or general computational strategy.

|  |
| --- |
| Probably the biggest computational challenge would be creating the logic for optimal moves. I think the general / optimal way to do it is to scan the board for all possible moves for your side and then show all possible responses to these moves and have a point system for the gain of these moves. For example a pawn might be worth 100 points, a rook worth 300, a queen worth 800, a king worth infinity etc. Then you can redo this to whatever depth, probably 2 or 3 layers deep to increase speed. There is some quarks and methods you can use to improve performance a lot and ways of determining how deep to search on different branches without going to the bottom like alpha beta pruning. Currently, I don’t know very much about these algorithms and features but I think it will be interesting to learn.  A smaller but also big challenge would be to determine the legal moves for different pieces. I would also like to implement the special cases like castling, pawn to queen promotion, and en pessant if I have time. To determine possible moves, I could  I don’t think I will implement the logic of being in check since it’s not super necessary to the game and is also pretty complex and intensive to determine. |

# 3. Approval

At this point, submit this document to your teacher to obtain feedback and approval on your proposal

# 4. Planning

**Time Appreciation**

Provide estimates for how long each phase of the project will take. You have already completed the first two phases, which is to create a proposal and a design. A project “should” take between 25 and 50 hours to complete.

|  |  |
| --- | --- |
| Phase / Component | Estimated  Hours |
| Proposal | 1 |
| Design | 2 |
| Planning | 2 |
| Implementation - Graphics | 1 |
| Implementation - Logic | 14 |
| Journal | 2 |
| Debugging | 8 |
| Documentation | 1 |
| Reflection | 2 |
| Other | 1 |
|  |  |
| Total Hours | 34 |

# 5. Process

## 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 have you learned about you as a learner?
    - What is working well?
    - What is not?

*Note 1: It is strongly recommended that you use a template for each entry: For an example, use the template provided on the next page. You can copy and paste new entries as you go.*

*Note 2: If this is a team project,* ***each member is expected to provide a separate journal.***

|  |  |  |  |
| --- | --- | --- | --- |
| Date | 3/10/2023 | Hours: | 2 |
| **Work Completed:** | Created simple chess board, with dynamic size for board that fits user resolution. | | |
| **Victories Achieved:** | Basically same as above. Chess board looks good and code is elegant. | | |
| **Resources used:** | Used some java graphics library and a YouTube video of someone who made something similar. | | |
| **Working well:** | The board works well. | | |
| **Not working well:** |  | | |
| **NEXT STEPS:** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Date |  | Hours: |  |
| **Work Completed:** |  | | |
| **Victories Achieved:** |  | | |
| **Resources used:** |  | | |
| **Working well:** |  | | |
| **Not working well:** |  | | |
| **NEXT STEPS:** |  | | |

# 6. Documentation

## Controls

(if applicable)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Key | Action | Mouse | Action | Other | Action |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

## Deployment / Execution

Provide instructions on how your product can be deployed (installed) and executed (run) on another computer.

* If you have used the templates provides in class, then this should be straight forward:
  + Import into Eclipse
  + run <main class>
* If you are using another platform (language, development environment, etc):
  + Note any dependencies and platform requirements
  + Be precise on how to run the project. Use screen captures if necessary
  + If you cannot explain this adequately, then your project likely does not meet the intended outcome, which is to create a stand-alone product.

|  |
| --- |
|  |
|  |

# 6. Reflection

*Note 1: Don’t be constrained by the size of the answer box!*

*Note 2: If this is a team project,* ***each member is expected to provide a separate reflection.***

1. Provide an updated time tracking chart, reflecting the hours actually spent on each category.

|  |  |
| --- | --- |
| Phase / Component | Estimated  Hours |
| Proposal |  |
| Design |  |
| Planning |  |
| Implementation - Graphics |  |
| Implementation - Logic |  |
| Journal |  |
| Debugging |  |
| Documentation |  |
| Reflection |  |
| Other |  |
|  |  |
| Total Hours |  |

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

|  |
| --- |
|  |

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

|  |
| --- |
|  |

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

|  |
| --- |
|  |

5. What worked well in your process?

|  |
| --- |
|  |

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

|  |
| --- |
|  |

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

|  |
| --- |
|  |

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!

|  |
| --- |
|  |

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

|  |
| --- |
|  |