Connor Goldsmith

CSC 4263

Individual Project Report

# What is the game

Pentago is a two-player board game that consists of a 6x6 board that is broken up into four 3x3 quadrants. The players take turns one by one, each placing a piece in an available slot, then rotating one of the quadrants by 90 degrees either clockwise or counter-clockwise.

# Goal of the game

The goal of the game is to get 5 of your pieces in a row at the end of a turn. This can be vertically, horizontally, or diagonally. You cannot win the game between a placement and a rotation, though you can win by your opponent rotating 5 of your pieces together.

# How I made the game

Previously, for CSC 4444 (Artificial Intelligence), I was part of a group whose project was an AI for pentago. To do this, we made a command line version of the game, where the user manually entered things like x coordinate, y coordinate, quad for rotation, and direction. We were able to get both a heuristic and a neural net working for this, but people did not like playing it due to the tediousness of taking a turn. Therefore, for this project, I made a video game version that actually utilized a GUI and take events like mouse clicks to make it easier to enjoy the game.

I started with the terminal version that I wrote for the group project, which can be found here: <https://github.com/cgolds2/pentago-ai>

I then made the assets that would be used for the actual game. For the project, I modeled every asset except for the table and the text board. I previously had some experience in animation, but mainly modeling for 3D printing, so while I have experience modeling, I don’t know much about texturing and materializing. The assets that I did make I used Blender to create. I then imported the models into unity, and because the process of taking the AI project and converting it into a unity game. The biggest hurdle with this was converting it from a mainly sequential game, to a more non-deterministic game. I had to make major changes to the game to counter things like the user being able to click at any point, having to wait for events, and other things that you do not have to factor into a purely terminal game. For writing the scripts I used Visual Studios because of the unity debugger plugin.

# How to play the game

The game is played purely through mouse clicks. You start from the title menu, and have options to either play against another player, or 2 computer options: heuristic and neural net. To play against the neural net more setup is required (see the section below). Selecting an option will bring you to the game screen. On a players turn, they must first select one of the holes to place a marble into. Then, they must select one of the arrows to rotate the corresponding quadrant. The game will end when either player gets 5 in a row, and a button will pop up announcing the winner. Clicking this button will bring the player(s) back to the main menu, upon which they can then start a new game

# Using the neural net

For this you must have conda installed, as tensorflow is needed to run the network (<https://www.anaconda.com/download/>). It is recommended that you run this on OSX, as windows runs of the net were not supported by the original project. You then must point the game (main menu -> options) to the python file that runs the net. If you are cloning the pentago-ai repo, this would be pentago-ai/4444 AI Proj/single\_move.py.