Anna Maria Ferrante
11/18/21
Updated 12/1/2021
15-112 Term Project Proposal

Project Description

The name of my project is "Outsmarting Blackjack". The goal of the application will be to teach users how to count cards (a method of keeping a count based on the cards that have been seen that allows blackjack players to have a better understanding of whether there are high or low cards left in a deck), and to allow the user to test their new skills in an online casino version of blackjack.

Competitive Analysis

I found an online version of a counting cards trainer in which five hands of blackjack are dealt. Then, the user has 40 seconds to guess the "count". If they run out of time, or the count is wrong, a box pops up with the correct count and asks if the user wants to play again. Because I already have a separate blackjack playing aspect of my project, and because counting cards works better further into a deck than just one hand, I decided to go a different direction with my counting card aspect of my project. Instead of having the cards dealt as if there was going to be a blackjack game, I will be having cards dealt at a specific speed until the deck runs out or until the user is asked to guess the count at a specific time (depending on the setting chosen). I do like the idea of the correct count popping up if the answer was wrong and asking if I would like the user to play again.

The online version of blackjack that I found only has the user play against the dealer. You can choose how much to bet, whether to hit, stand, double down, or split. The count is displayed and the bet is displayed. The user is told whether they win or lose to the dealer. I like the idea of displaying the count. However, I want to add an aspect to my game that allows the computer to play against the dealer as well. Not only will this allow the user to use the method of counting cards more to their advantage, but it will also add complexity to the code and to the game.

Structural Plan

##
Main Screen
##

- Code for the main screen (where the name is displayed and the username is taken) will be written under this mode.

##

Home Screen

##

- Code for the home screen (where the user can choose blackjack or counting cards) will be written under this mode.

##

Blackjack

##

- Code for blackjack game
- Functions include starting the game, timerFired, redrawAll, keyPressed, mousePressed, etc.

##

Counting Cards

##

- Code for counting cards game.
- Functions include starting the game, timerFired, redrawAll, keyPressed, mousePressed, etc.

At the very bottom of the code there will be an "appStarted" function as well as a function that creates a list with card images and values.

Algorithmic Plan

The most complex part of my project will most likely be when I have the computer play against the dealer as various players at the table. I want to create a recursive backtracking function that allows the computer to learn multiple strategies for winning blackjack. I will create a dictionary with keys "Hit", "Stand", "Double Down", "Split". The function will be given a random hand and dealer's up-card. I will go through the potential options and see if they are legal. If it is, and the hand wins, then I will add that [[hand], [dealer's up-card]] list to the dictionary. If it is not, I will go onto the next hand and dealer up-card pair. I will go through an entire deck of cards (possibly more) until the function can get thirteen total winning hands. If not, I will backtrack and add the cards in the dictionary back to the list. I will create multiple dictionaries this way in order to have multiple versions of play. I will potentially go through more decks for the feedback for the user.

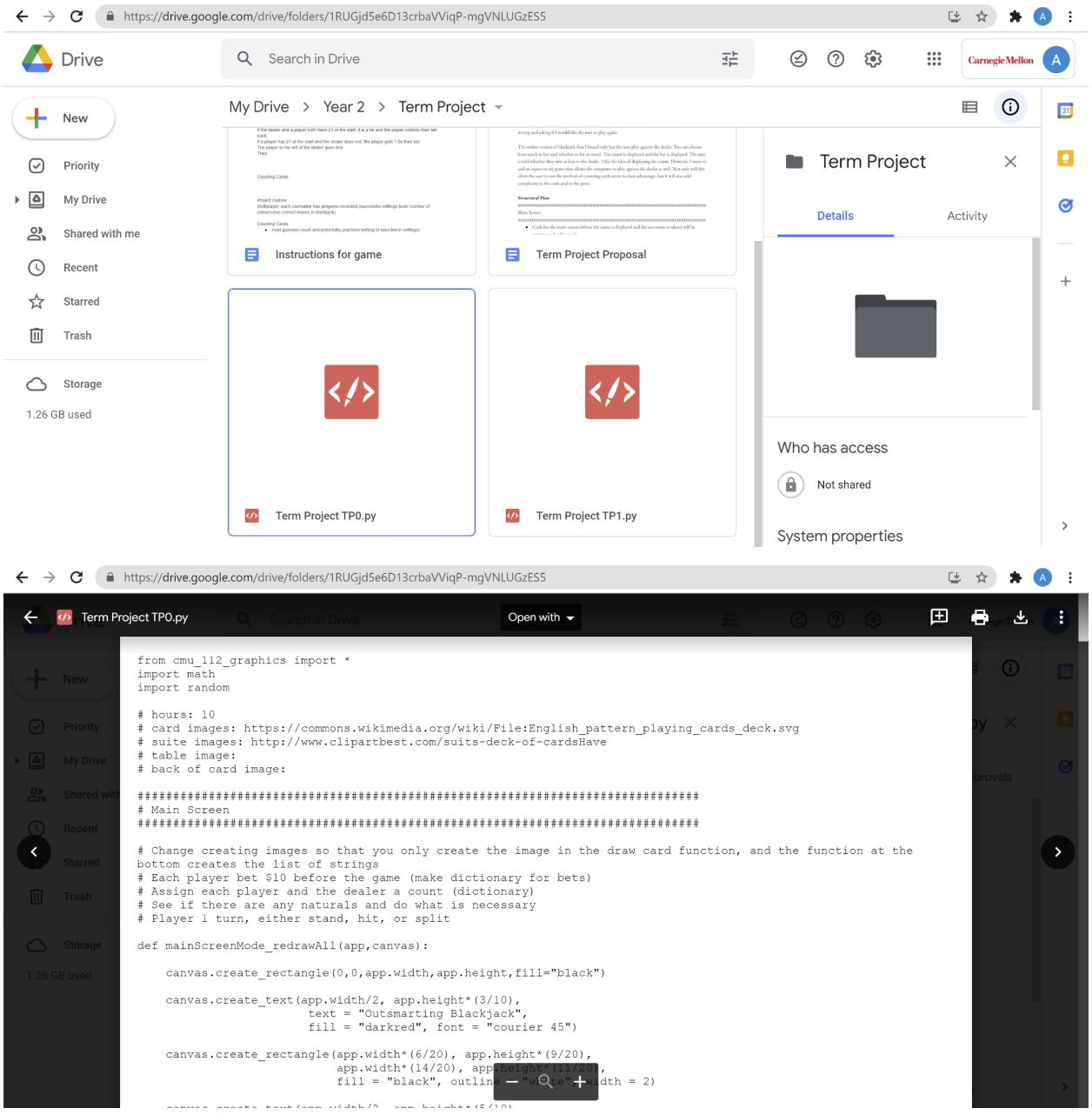
Timeline Plan

Thursday: One-player version of blackjack.
Friday: Blackjack involving AI.
Saturday: Multiplayer blackjack
Sunday: Counting cards simulator

Monday: Feedback on betting and play based on card counting and complex algorithm described above. (blackjack)
Tuesday: Allow user to know their progress

Version Control Plan

I am saving my code to a folder named "Term Project" in my google drive which saves to the cloud.



TP2 Update

Renamed “21|12”

Instead of the algorithmic plan explained above, I decided to use Monte Carlo methods to run through black jack hands thousands of times to determine whether hitting, standing, splitting, or doubling down has the higher probability of winning, and to create a policy based on those probabilities.

Additionally, I created an algorithm for the computer to know how much to bet based on the count (of the cards already dealt in the game), on their amount of money left, and on a random risk factor assigned to each player. (to add a human aspect). I believe the next most challenging parts of my game algorithmically may be incorporating the split option into the Monte Carlo, making the game multiplayer, and adding progress reports to the counting cards simulation.

New timeline:
Tuesday: Incorporate settings, Multiplayer mode
Wednesday: Blackjack variant
Before deadline: Counting cards progress, adding images/other detail

TP3 Update

I made no updates to my overall design.