CS65: Introduction to Computer Science

Final Project Proposal

Md Alimoor Reza
Assistant Professor of Computer Science

Final Project

- Final project will be due before the end of the semester

Grading and requirements:

- *Programming Assignments (25%).* Homework programming activities.
- *Labs (20%).* Completing programming activities during class.
- *Quizzes (10%). true/false, fill in the blanks,* etc.
- *Midterm (15%).* Paper based exam midway through the semester.
- *Final (20%).* Paper based exam by the end of the semester.
- *Final project (10%).* Your proposed group project (2-3 members).

- The project will also include a final demonstration of the project: 1st week of December (during class time)

- The project proposal is due by **Sunday November 13, 2022**

Final Project Proposal

- This is an opportunity for you to explore a project of your own choosing that uses Python programming

- Submit a **1/2 to 1 page** written project proposal with a storyboard for your final project.
  - A few rough sketches of what your program will look like to help guide the development
  - Your storyboard should be a graphical description of the "flow" of your project
    - *You can use Powerpoint (windows) or Keynote (OS X) to prepare the flowchart or other graphical description)*

Final Project Proposal

- Your project proposal should either be a Word processing document or in a PDF file format and it should include the following information:

1. Your name or team members' names

2. The idea (the game, simulation, visualization, etc. you plan to implement for your final project)

3. Any datasets you plan to use in your project (if applicable)

4. A development plan:

- What will you do first, second, third, etc.?
- What functions will you use or develop?

Final Project Proposal

- Submit the project proposal on Blackboard (10 points)

- One PDF is sufficient for your group as long as it has the names of the members

- The project proposal is due by **Sunday November 13, 2022**

Final Project Ideas

- Text-based games: dice game, tic-tac-toe, etc
- Graphics game
- Simulation
- Data analytics: explore any dataset, visualize properties, compute statistics, etc
- Or anything else of your choice

Final Project Proposal: Example 1

Dataset: US Unemployment Dataset (2010 - 2020) | Kaggle

Project Idea & Summary:

In this project, we will be taking a look at a dataset containing 10 years of unemployment data from the US. The variables included in this dataset are year and month the data were recorded, the unemployment rate based on the qualification of education of adults at that time (primary school, high school, associates' degree, and professional degree), as well as demographic data looking at the race and gender of the people in the US who were observed in the data. This dataset also contains state-wide unemployment rate data for the year 2020.

Specifically, we will be looking at the trends in unemployment across education qualifications over the 10-year period, and comparing the trends to the more recent effects on US unemployment from the COVID-19 pandemic. By performing such analysis, we will be able to better understand how the COVID-19 pandemic has impacted the unemployment rate since 2020, by viewing the trends across race and education qualification since 2010.

With Python, we will be using data visualization functions to create charts and graphs to visualize the dataset and compare trends across the 10-year period. By using such packages in Python, we can create a dashboard that provides our audience with visualizations to better understand the dataset and the conclusions we have arrived at from performing the analysis. Our Python code will also use functions learned in class to help aggregate the data, such as if/elif functions, to compare variables in the dataset and make understandable conclusions.

Final Project Proposal: Example 2

Idea: Ride the Bus / Irish Poker

Ride the bus or as some others know it Irish poker is a game that is played with more than one person. Note, this does not have to be played as a drinking game, that is just most commonly how it is played.

Step 1) Round 1: The game starts with one of the players guessing if the card is going to be black or red, if the guess is wrong the player has to take a sip of his drink and starts over again, if the guess is correct the user does not have to take a sip and moves on to step 2. Note that all the players have to select their choice before the output is revealed.

Step 2) Round 2: For the second part of the game the code will prompt the users with a question of "higher" or "lower". Depending on the user choice the user will be prompted to take a sip and restart at round 1, or move to the next round. For example, if the user selects "higher" and the card has a higher value than the card that we previously pulled the user will pass on without taking a sip, and if the card has a lower value the user takes a sip and restarts.

Step 3) Round 3: In the third round, the player will input "between" or "outside", to guess whether the next card they flip will be within the range or outside the range of the two cards from the first two rounds. If they guess correctly they move on to the final round, if not they restart at round 1.

Step 4) Round 4: In the final round, the player will input "Spade" or "Club" or "Heart" or "Diamond", guessing what the suit of the final card will be. If the suit of the flipped card is correctly guessed, the win the game, if not they restart at round 1.

Datasets We Will Use: For our final project we will use def, while, for, list, random, dictionary.

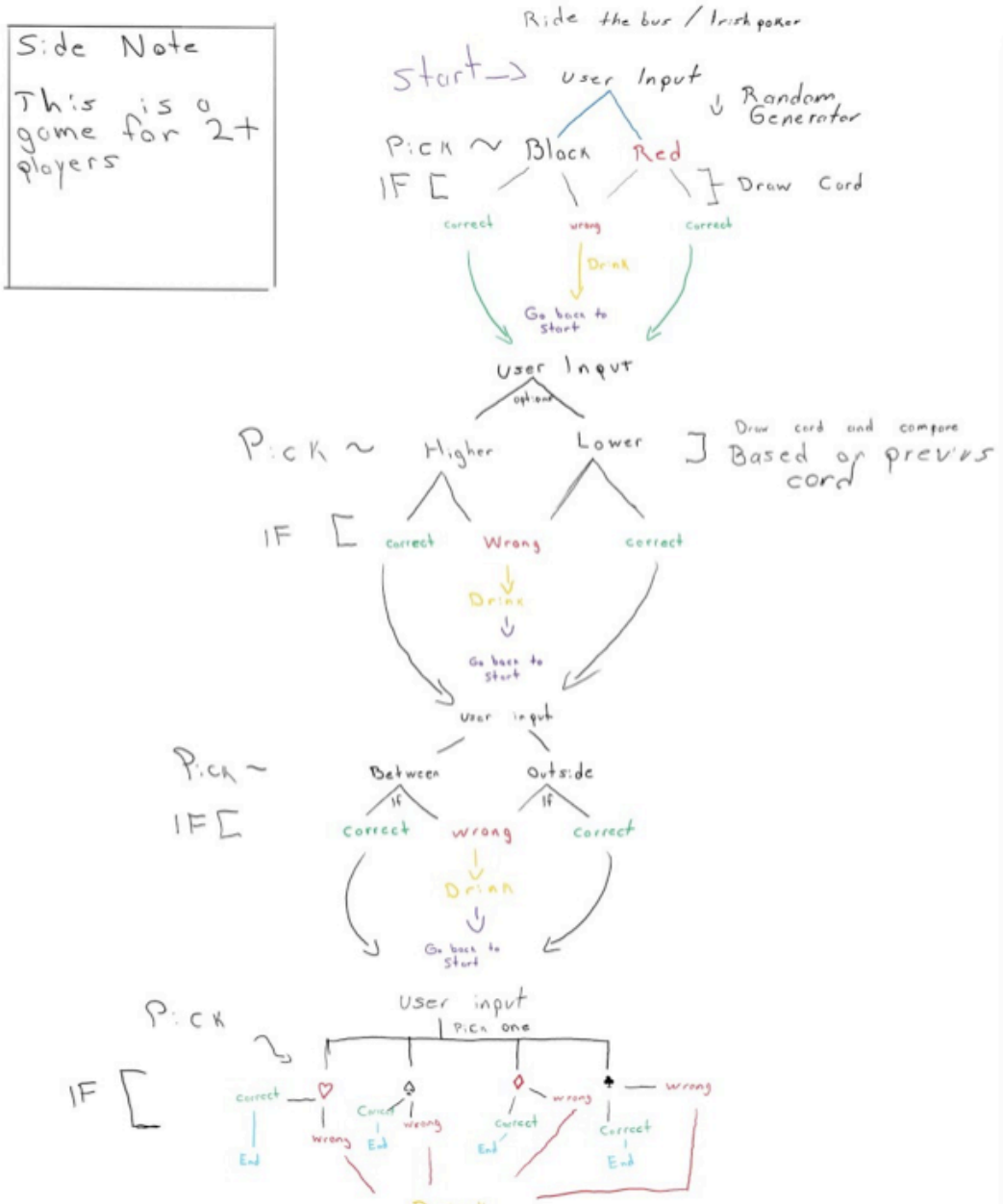
Final Project Proposal: Example 2

Development Plan: We will be dividing work into different parts while also using resources that we have used throughout class. We might also utilize the professor for and tutors for extra guidance. Game will make use of "If" statements and "While" loops primarily, along with random generation of playing cards. The entire game will be within a large "While" loop that repeats the steps from step 1 while step 4 is false, and ends the game when step 4 is true, with smaller "If" functions constituting each round of the game.

Steps of Development:

1) Create a function that draws a random playing card
2) Create a function that displays and image of the drawn playing card
3) Create a menu to explain rules of game and execute functions
4) Create a function for each round 1-4
5) Create skeleton "While" loop for entire game that contains all the rounds, e.g. "While flag == false:", and set round 4 as flag == false, until it is passed
6) Call functions within skeleton "While" loop to assemble game

Final Project Proposal: Example 2



Final Project Proposal: Example 3

Idea: Sorting Algorithm Visualizer

For our project, we plan to create a sorting algorithm visualizer. We intend to use Python's built-in GUI library t-kinter to achieve this. T-kinter will not create the most visually pleasing software however it will create a very functional and fast system that allows for a lot of modifiability in the parameters, options, and display of the algorithms.

The algorithms we intend to use for sorting are bubble sort, selection sort, insertion sort, merge sort, quick sort, random quick sort, counting sort, and radix sort.

We intend to use rectangles to represent the magnitude of a value in the list and then visually display which boxes are being interacted with and where they are being moved to. We intend to simply swap places of the boxes instead of implementing animations as we want to focus on learning how to create GUIs and understand the sorting algorithms behind them.

Development Plan:

1. Learn for ourselves how each sort functions and what it may be used for
2. Figure out the order in which these sorts will be written/done
3. Write the functions for each individual sort
4. Write the main function where the order will be set and/or the user will be able to choose which sort to execute
5. Run main function
6. See visualization of each sort algorithm by run through

- Functions being used or developed: defined, input, int, main(), for loop, nested for loop, while loop, lists, classes, Python libraries

This is a sort of example based off a more complex idea of what we want to do:
https://visualgo.net/en/sorting

Final Project Submission

- Submit the project code and report on Blackboard (100 points)

- For full credit, you should plan on investing several hours
  - *You will be asked during your presentation*

- The final project is due on **December 12, 2022**
  - *You have more than a month so plan accordingly*
  - *Sooner the better*

Final Project Submission