Team Name: M&M___

Team Members: Richard Mercado and Mariana Molano_

Project Title: Candidate Selection Database

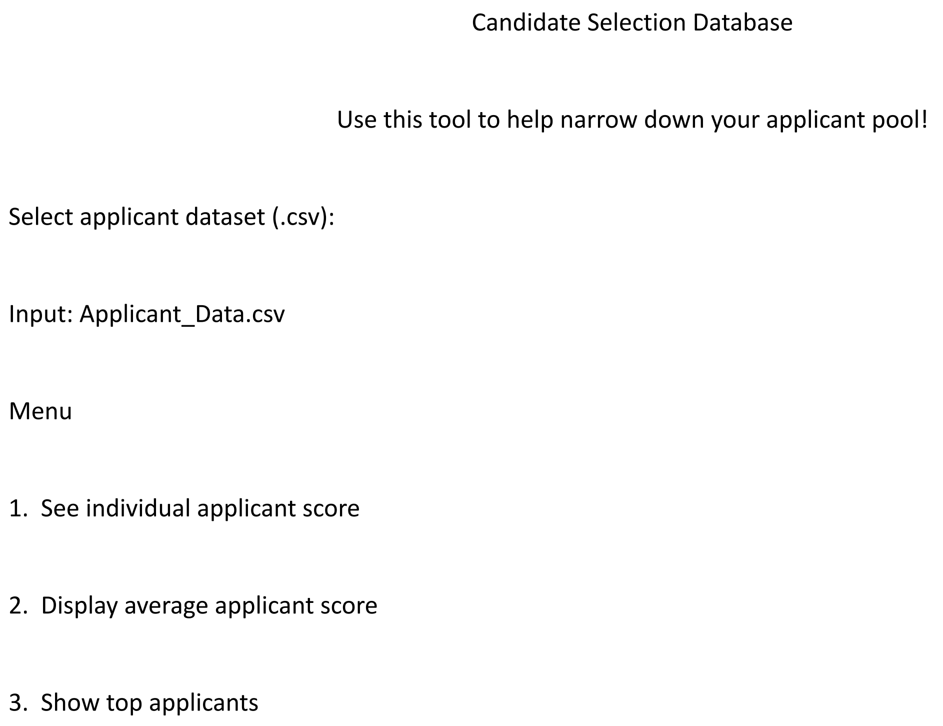
GitHub URL: https://github.com/marianacmolano/project3_cop3530

Video URL: https://youtu.be/n0F7Nw7xvEQ

1. **Problem: What problem are we trying to solve? [0.25 point]**
   1. We will be organizing internship applicants based on whether or not they meet multiple qualifications. Each candidate will be given a score based on their results.
2. **Motivation: Why is this a problem? [0.25 point]**
   1. Selecting superb employees is integral to the foundation of a business. Choosing the most qualified internship candidates can be difficult for any recruiter, however, most companies have a minimum standard of qualifications for potential employees. Therefore, a software that would assist in the narrowing down of candidates would be a great asset to any company.
3. **Features: When do we know that we have solved the problem? [0.25 point]**
   1. Our team will know that we have solved the problem when candidate pools have been accurately downsized. For example, in a data set of 100,000 candidates, the program will have narrowed them down by a significant amount. Recruiter will input the dataset of applicants, and only applicants with the highest scores will be added to the trees.
   2. Another feature will be candidate lookup, which will return their individual score.
   3. Printing top candidates (traversing through tree).
4. **Data: (Public data set we will be using and the link to the public data set) or (Schema of randomly generated data - i.e. what are the different columns in our dataset and the respective data types) [0.25 point]**
   1. We will be using randomly generated data through a pandas data frame in python. This code is accounted for in the GitHub repository. The columns in our data set are applicant ID (integer

0-100000), number of degrees(0-5), GPA (float 1.5-4), amount of work experience (integer 0-9), whether or not they had a cover letter (integer 1 or 0), application completion time (integer in hours, 1-730), interview score (integer 1-10), and a diversity point (integer 1 or 0). The file name is Applicant_Data.csv, also found in the GitHub repo.

5. **Tools: Programming languages or any tools/frameworks we will be using [0.25 point]**
    1. We will be using C++, Microsoft Excel, GitHub, Google Docs, and JupyterLab.
    2. C++: We will be coding our program in C++ in order to best implement the B+ and Red-Black Trees from Scratch.
    3. JupyterLab: We randomly generated our data with the help of a pandas dataframe in python.
    4. GoogleDocs: To collaborate in the creation of this report.
    5. GitHub: To facilitate collaborating on the code.
    6. Excel: To view our randomly generated data and ensure it meets our standards.
6. **Visuals: Wireframes/Sketches of the interface or the menu driven program [0.25 points]**
    1. Text-based menu driven program.



Candidate Selection Database

Use this tool to help narrow down your applicant pool!

Select applicant dataset (.csv):

Input: Applicant_Data.csv

Menu

1. See individual applicant score

2. Display average applicant score

3. Show top applicants

7. **Strategy: Preliminary algorithms or data structures you may want to implement and how would you represent the data [0.25 points]**
    1. The two data structures we are implementing are a red-black tree and B+ tree. We'll assign scores to applicants depending on their qualifications. The overall scores will be sorted into each tree- a red-black tree, and B+ tree. Then we will compare the performance of each tree.
8. **Distribution of Responsibility and Roles: Who is responsible for what? [0.25 points]**
    1. Andre: Develop the B+ tree, the Red-Black tree and their functionality. Measuring the differences in time complexities.
    2. Mariana: Create randomly generated data. Developing the text-based user interface, taking the data from a csv file, and parsing it into our program. Create the applicant class and assign scores.
    3. Joint: Creating the final presentation walk-through video, Group report, and GitHub Repository.

ANALYSIS

- Any changes the group made after the proposal? The rationale behind the changes.
    - We made a few changes after the proposal. Firstly, we lost one of our group members because they dropped the class. Secondly, we changed some of the qualifications that would be included in our data.
- Complexity analysis of the major functions/features you implemented in terms of Big O for the worst case
    - ReadFile() - O(number of applicants)
    - BuildTree() - O(number of applicants * log(number of applicants))
    - CalculateScore() - O(number of applicants * log(number of applicants))
    - CalculateAverage() - O(1)
    - SearchForApplicantByID() - O(number of applicants)
    - TopNApplicants() - O(number of Applicants + the number of top applicants)

REFLECTION

- As a group, how was the overall experience for the project?
  - We found this project to be a valuable learning experience. It was much more practical than previous projects where we could decide which features to implement, and input/output set up. The greater creative freedom that came with this project was refreshing and a fantastic motivator for starting personal projects.
- Did you have any challenges? If so, describe.
  - One of the brief challenges we have encountered is how to use the same nodes for both trees.
  - Another problem which we encountered was how to structure the trees to fit the problem we were trying to solve, especially the B+ tree.
  - Another challenge we encountered was generating random values for applicants. We were able to do this but we had to get rid of the name aspect of the Applicant because it was hard to generate them randomly.
- If you were to start once again as a group, any changes you would make to the project and/or workflow?
  - We think that finding a third teammate would have been beneficial and helped us a lot in dividing the work.
- Comment on what each of the members learned through this process.
  - Mariana: I have learned a lot through this process, especially how to incorporate other tools in order to accomplish a goal. There are different programming languages for a reason. Each language is made to specialize in certain tasks and python was of great assistance in creating our randomized data.
  - Richard: What I learned about was how to complement complex data structures to fit the problem which we needed to solve. B+ trees and Red and Black Trees have complicated implementations and being able to manipulate these implementations to fit our applicant class was a great accomplishment.

References

https://www.geeksforgeeks.org/introduction-to-red-black-tree/

https://www.geeksforgeeks.org/introduction-of-b-tree/

https://www.analyticsvidhya.com/blog/2021/08/python-tutorial-working-with-csv-file-for-data-science/

https://kanoki.org/numpy-to-generate-random-number-between-0-and-1

https://www.programiz.com/dsa/red-black-tree

https://www.programiz.com/dsa/b-plus-tree