M4 Team Project: Proposal

David Spenziero

Jacob Morris

Hassan Ul Haq

Data Structures & Algorithms 2021FA1-CSCI-3015-01 : M4 Team Project: Proposal

Professor: James Iannibelli

November 21, 2021

Contents

[**Background** 3](#_Toc88432478)

[**Problem Description** 4](#_Toc88432479)

[**Solutions for Implementation and Evaluation** 4](#_Toc88432480)

# **Background**

In this first project assignment we as a group are required to present a formally written project proposal which includes a problem description and the team's choice of three possible solutions for implementation and evaluation. The project to be proposed should analyze the pragmatic differences in how three different algorithms and/or data structures solve the selected problem.

Following is to be achieved to complete the project assignment by the end of this course:

* Self organize into teams (no more than 6 students per team)
* Choose a worthy programming problem that could by resolved using three different data structures and/or algorithms.
* Properly code and present the three possible solutions.
  + Compare them based on efficiency, speed, and accuracy
* Develop and perform a  meaningful test plan and data.
* Analyze the results.
  + Be sure to show the differences between using the various data structures and/or algorithms in creating your solutions.
  + Present all source code, classes, test data, and results.

# **Problem Description**

We are planning to develop a simple employee database management system that takes user input and sorts the database entries by choosing one of the three sorting algorithms. As a final step, the runtime and space complexity of the sort algorithms and/or data structures will be quantitatively analyzed.

# **Solutions for Implementation and Evaluation**

As a first step,  we will generate a dummy database of employees by using a random database generator. Our plan is to create the database (DB) generator from scratch. For each entry, the DB generator will randomly generate Employee Name, Date of Birth, Employee-ID, Department, Designation etc. from a list of names, departments, designations (combinations from a list already provided). Once our database is created, we will develop three different searching algorithms to sort the database for any particular DB entry (for example sorting the database according to Employee ID or Surname). For the sorting part, the user of the application will also decide on the sort algorithm to be used to sort the DB, for example, merger sort on Employee-ID. We are planning to use Selection Sort, Insertion Sort, and Merge Sort in this problem.

Finally, for all three sorting algorithms we will quantitatively analyze and compare their performance to each other.  The results found will be tabulated and graphed to demonstrate our understanding. We also hope to experiment with various data structures to see if the choice of the ADTs affects the runtime complexity of the final solution.