2021

James Scott

N9934618

5/10/2021

CAB301 Assignment 1

CAB301 – Algorithms and Complexity

Software Application for a Tool Library

Table of contents

[Introduction 2](#_Toc72680211)

[Design and Analysis of Algorithms 2](#_Toc72680212)

[Algorithm Design 2](#_Toc72680213)

[Algorithm Analysis 2](#_Toc72680214)

[Software Test plan and results 2](#_Toc72680215)

[References 3](#_Toc72680216)

## Introduction

I have attempted to implement each of the required functionalities in both the Staff and Member menus. I made many difficult design choices on how to approach the problems posed by the interfaces, I will explore the most notable choices here.

Firstly, I was not able to come up with an eloquent solution to the problem of MemberCollection storage. To get things working for submission I have used a MemberCollection at the program.cs level and an internal MemberCollection within ToolLibrarySystem.cs. This decision later posed challenges with updating/using the correct MemberCollection, but I could not think of a cleaner alternative within the bounds of the assignment.

Furthermore, as my Jagged Arrays of ToolCollection were stored as private members of my ToolLibrarySystem class, a lot of the user input and console logic has been implemented within the ToolLibrarySystem class. Obviously it would be ideal to extrapolate this logic out to the program.cs level, but I have tried to mitigate this as best I can with plenty of utility functions to clean up the console logging logic.

I have made no changes to or deviations from the interfaces, and all discovered bugs have been ironed out before submission.

## Design and Analysis of Algorithms

#### Algorithm Design

The algorithm design is a big component so this is essentially the pseudocode and a discussion of the data structures used in the algorithm including the reasons for their choice.

#### Algorithm Analysis

The Algorithm analysis should include a theoretical analysis for the determination of the algorithm complexity and I would recommend including an empirical analysis either using time or counts of basic operations. The empirical analysis can support your theoretical analysis and provides a well rounded and complete analysis of your Top 3 algorithm.

## Software Test plan and results

Finally the Software Test plan providing the screenshots of all the required functionality and a single screenshot of how your application appears at the end of each process (e.g. adding a member, displaying tools) - I would also include in the test plan some evidence of input validation to show that you have considered this as part of your implementation so numbers out of range, wrong data type inputs with the appropriate screen shots.

|  |  |
| --- | --- |
| **Task** | **Expected Outcome** |
| **Main Menu** | |
| Access staff menu | When correct credentials are supplied, success. |
| Access member menu | When correct credentials are supplied, success. |
| Exiting the program |  |
| **Staff Menu** | |
| Add a new tool to the system |  |
| Increase the quantity of a tool |  |
| Decrease the quantity of a tool |  |
| Register a new member |  |
| Search for a members contact number |  |
| Exit to main menu | Success |
| **Member Menu** | |
| List all the tools of a tool type |  |
| Borrow a tool |  |
| Return a tool |  |
| Display all of my currently borrowed tools |  |
| Display the three most frequently rented tools |  |
| Exit to main menu | Success. |

## References

[NearHuscarl](https://stackoverflow.com/users/9449426/nearhuscarl). (2020, September 26). *ag-Grid React forgets gridApi after it has been set.*  
<https://stackoverflow.com/questions/64071586/ag-grid-react-forgets-gridapi-after-it-has-been-set>