

**ASIA PACIFIC UNIVERSITY OF TECHNOLOGY AND INNOVATION**

**INDIVIDUAL ASSIGNMENT**

**CT018-3-1-ICP**

**Introduction To C Programming**

**Intake Code: APD1F2209CS(CYB)**

**Hand Out Date:** March 15, 2023

**Hand In Date:** June 14, 2023

**Student Name: Hong Rui Yi**

**Title: APU Programming Café Management System**

**Weightage: 50 %**

Table of Contents

[1.0 Introduction and assumptions 4](#_Toc137426772)

[2.0 Design of the program 5](#_Toc137426773)

[2.1 Flowchart 5](#_Toc137426774)

[2.1.1 Main Logic 5](#_Toc137426775)

[2.1.2 Admin Menu 6](#_Toc137426776)

[2.1.3 Student Menu 7](#_Toc137426777)

[2.1.4 Tutor Menu 8](#_Toc137426778)

[2.1.5 Sub Menu 9](#_Toc137426779)

[2.2 Pseudocode 10](#_Toc137426780)

[2.2.1 Define structure 10](#_Toc137426781)

[2.2.2 Login Function 11](#_Toc137426782)

[2.2.3 Register Function 12](#_Toc137426783)

[2.2.4 Print Function 12](#_Toc137426784)

[2.2.5 Delete Function 13](#_Toc137426785)

[2.2.6 Update Function 14](#_Toc137426786)

[2.2.7 Read from file 15](#_Toc137426787)

[2.2.8 Write into file 16](#_Toc137426788)

[3.0 Additional features 17](#_Toc137426789)

[3.1 Trim Function 17](#_Toc137426790)

[3.2 Check Void of File 17](#_Toc137426791)

[3.3 Check Exist of ID 17](#_Toc137426792)

[3.3 Find Data struct by its ID 18](#_Toc137426793)

[4.0 Sample outputs 19](#_Toc137426794)

[4.1 Confirmation Page 19](#_Toc137426795)

[4.1 Login Page 19](#_Toc137426796)

[4.2 Main Page 19](#_Toc137426797)

[4.3 Add Data 20](#_Toc137426798)

[4.4 Print Data 21](#_Toc137426799)

[4.5 Delete Data 23](#_Toc137426800)

[4.6 Update Data 24](#_Toc137426801)

[4.7 Test plan 25](#_Toc137426802)

[5.0 Conclusion 26](#_Toc137426803)

[6.0 Reference List 27](#_Toc137426804)

# Introduction and assumptions

The goal of C programming Individual projects includes developing an APU Programming Café management system in C programming language. Before beginning the coding phase, the program's design is created using flowcharts. This system is a software solution that is specifically built to adjust to a session arrangement. It provides functional areas for three categories of users: Admin, Tutor, and Student. To create, view, change, or delete a specific task, they must log in to the system using their user id and password. When the task is completed, the data in a text file will be updated.

**Assumption**

1. Acc.txt records all the accounts detail which including admin, tutor, and student information.
2. User\_pwd.txt records all the account id and password.
3. Enroll.txt records the list of sessions and participating students.
4. Session.txt records all the sessions detail and linked with tutor code.
5. The default id and password for the administrator is admin and admin.
6. Every user id is unique and cannot be registered twice.
7. Each session can only be assigned to one tutor.
8. The tutor must be registered before a new session can be assigned.

# 2.0 Design of the program

## 2.1 Flowchart

### 2.1.1 Main Logic

A diagram of a program

Description automatically generated with low confidence

Figure Main Logic Flowchart

### 2.1.2 Admin Menu

A picture containing diagram, technical drawing, plan, line

Description automatically generated

Figure Admin Menu Flowchart

### 2.1.3 Student Menu

A picture containing diagram, technical drawing, line, plan

Description automatically generated

Figure Student Menu Flowchart

### 2.1.4 Tutor Menu

A picture containing diagram, plan, technical drawing, line

Description automatically generated

Figure Tutor Menu Flowchart

### 2.1.5 Sub Menu

A picture containing diagram, plan, technical drawing, line

Description automatically generated

Figure Sub Menu Flowchart

## 2.2 Pseudocode

### 2.2.1 Define structure

A screenshot of a computer program

Description automatically generated with medium confidence

Figure Define Pseudocode

### 2.2.2 Login Function

A screenshot of a computer program

Description automatically generated with medium confidenceA screen shot of a computer code

Description automatically generated with low confidence

Figure Login Pseudocode

### 2.2.3 Register Function

A screenshot of a computer program

Description automatically generated with low confidence

Figure Register Pseudocode

### 2.2.4 Print Function

A screenshot of a computer code

Description automatically generated with low confidence

Figure Print Pseudocode

### 2.2.5 Delete Function

A picture containing text, screenshot, font

Description automatically generated

Figure Delete Pseudocode

### 2.2.6 Update Function

A screenshot of a computer program

Description automatically generated with medium confidenceA picture containing text, screenshot, font

Description automatically generated

Figure Update Pseudocode

### 2.2.7 Read from file

A screenshot of a computer program

Description automatically generated with medium confidence

Figure Read From File Pseudocode

### 2.2.8 Write into file

A screenshot of a computer program

Description automatically generated with low confidence

Figure Write In File Pseudocode

# 3.0 Additional features

## 3.1 Trim Function

A picture containing text, screenshot, font

Description automatically generated

Figure Trim function

This function is to remove all spacing from text and keep single spaces between words. It will detect the error of parameter char is NULL or empty.

## 3.2 Check Void of File

A picture containing text, screenshot, font

Description automatically generated

Figure Check void of file function

It is used in every file access to check if the file content is null or not. If it is not empty, it will return 1, else return 0.

## 3.3 Check Exist of ID

A picture containing text, screenshot, font, software

Description automatically generated

Figure Check exist of ID function

To avoid data duplicated, this function helps in checking the existence of user id in user\_pwd.txt file. Hence, the registration and printing sections can be more accurate and reliable. If ID exists, it will return 1, else return 0.

## 3.3 Find Data struct by its ID

A picture containing text, screenshot, font, software

Description automatically generated

Figure Find data struct by id function

In student and tutor mode, registration and printing sections must be linked with their ID. So that, we require this function to search for and return the data struct pointer by passing its data struct and ID.

(C introduction, n.d.)

(Understanding the basics of linked list, n.d.)

# 4.0 Sample outputs

## 4.1 Confirmation Page

Before entering Login page, system will ask user to press any keys to continue or press q to exit.

A black screen with white text

Description automatically generated with low confidence

Figure Confirmation Page

## 4.1 Login Page

After confirmation, user will see the login page. They need to enter their role by entering 0(admin) or 1(Student) or 2(Tutor). Then, the user needs to enter his unique ID, and system will verify it. If it exists, user needs to key in his password.

A screen shot of a computer

Description automatically generated with medium confidence

Figure Login Page

## 4.2 Main Page

Once the user enters the correct password, the system will display the main page. There will be five options to pick from if the user's role is admin or student. However, only the admin will have access to the sub menu which allows to alter student, tutor, session and enroll sections. Besides, there is only a print function and exit function for tutor menu. If user select an invalid selection, ID or password, it will show an error message. Once the user chooses the exit selection, the system will jump back to the confirmation page.

A screen shot of a computer

Description automatically generated with medium confidence

Figure Main Page

A screen shot of a computer

Description automatically generated with low confidence

Figure Submenu

A black screen with white text

Description automatically generated with low confidence

Figure Tutor Menu

A black screen with white text

Description automatically generated with low confidence

Figure Error Message

## 4.3 Add Data

Under admin role, user allow to add data into certain role’s data according to sub menu. If user press 1, it will require him to insert unique ID, full name and gender, and the default password will be generated automatically.

A black screen with white text

Description automatically generated with low confidence

Figure Admin registered student data

If user press 2, it will require him to insert unique ID, full name and course, and the default password will be generated automatically. A black screen with white text

Description automatically generated with low confidence

Figure Admin registered tutor data

If user press 3, it will require him to insert unique ID, title, day, start time, location, and tutor code. The day input range is from 1 to 7 which represent Monday to Sunday. Start time format is 24-hour notation.A screen shot of a computer

Description automatically generated with medium confidence

Figure Admin registered session data

If user press 4, it will require him to insert registered student ID and registered session ID, the student’s name, session location and tutor name will be linked automatically. A black screen with white text

Description automatically generated with low confidence

Figure Admin enrolled session

Under student role, user allows to add session to his list. The student ID will be provided automatically, and he will just need to input the session ID.

A black screen with white text

Description automatically generated with low confidence

Figure Student enrolled session

## 4.4 Print Data

Under admin role, user allow to print data from certain role’s data according to sub menu. If user press 1, it will show student details.

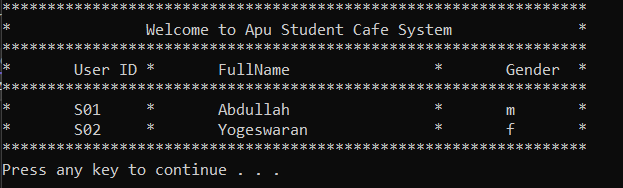


Figure Admin printed student data

If user press 2, it will show tutor details. A screen shot of a computer

Description automatically generated with medium confidence

Figure Admin printed tutor data

If user press 3, it will show session details.

A picture containing screenshot, line

Description automatically generated

Figure Admin printed session data

If user press 4, it will show enrolled session details.

A screenshot of a computer program

Description automatically generated with low confidence

Figure Admin printed enrolled session

Under tutor role, it will only show the login user’s enrolled session details.

A screen shot of a computer

Description automatically generated with low confidence

Figure tutor printed his data

Under student role, it will only show the login user’s enrolled session details.

A screen shot of a computer program

Description automatically generated with low confidence

Figure student printed his data

## 4.5 Delete Data

Under admin role, user allows to delete data from certain role’s data according to sub menu. If user press 1, it will show student details. He needs to input a student id, after which it will be removed from the table. The same procedure will be followed to erase data from other roles such as student, tutor, session, and enrolled session.

A screen shot of a computer

Description automatically generated with medium confidence

Figure Display student data



Figure Input student id

A screenshot of a computer screen

Description automatically generated with low confidence

Figure Display remaining data

Under student role, user can only delete his enrolled session data. He needs to input an enrolled session id, after which it will be removed from the table.

A screen shot of a computer program

Description automatically generated with low confidence

Figure Show enrolled session



Figure Input session id

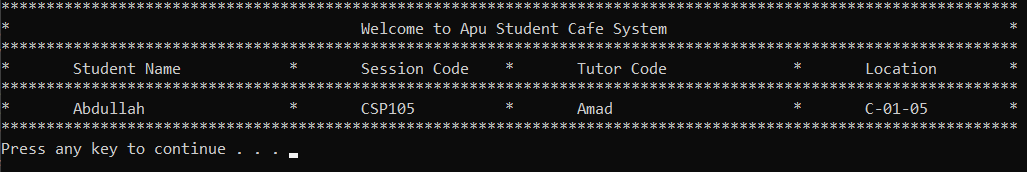


Figure Display remaining session

## 4.6 Update Data

Under admin role, user allow to update data from certain role’s data according to sub menu. If user press 1, it will show student details. He needs to enter a student ID, after which an updated item sub menu will be displayed. If the user selects 1 and enters a modified item detail, the table will be updated. The same procedure will be followed to update data from other roles such as student, tutor, session, and enrolled session.

A screen shot of a computer

Description automatically generated with medium confidence

Figure Display student data



Figure Input student id

A screen shot of a computer screen

Description automatically generated with low confidence

Figure Submenu

A screenshot of a computer screen

Description automatically generated with medium confidence

Figure Updated data

## 4.7 Test plan

|  |  |  |
| --- | --- | --- |
| **Test Case** | **Input** | **Output** |
| Enter any key in confirmation page | A space. 1 | Login page |
| q | Quit program |
| Enter invalid ID and password | Admin, 123 | Fail (invalid password) |
| T05, 123 | Fail (user ID not found) |
| admin, admin | Success |
| Enter option in main page (admin) | 6 | Fail |
| 0, 1, 2, 3, 4 | Success |
| Enter option in main page (student) | 4, 6 |  |
| 0, 1, 2, 3, | Success |
| Enter option in main page (tutor) | 2 | Fail |
| 0, 1 | Success |
| Enter option for submenu (admin) | 0,5 | Fail |
| 1, 2, 3, 4 | Success |
| Enter gender while adding student data | M, m, F, f | Success |
| A | Fail |
| Enter day while adding session data | 1 | Success (Monday) |
| 8 | Fail |
| Enter start time while adding session data | 14:30 | Fail (02:30 pm) |
| 25:00 | Fail |
| Enter Tutor Code while adding session data | T01 | Fail (Tutor full) |
| T08 | Fail (Tutor not found) |
| T03 (Registered) | Success |
| Enter student id to delete/update the data | S01 | Success |
| S03 | Fail (Student ID not found) |
| Enter session while enrolling, delete, update session (student) | C001 | Fail (Session not found) |
| PYP101 | Success |

# 5.0 Conclusion

In a nutshell, a student café management system has been successfully constructed using the C programming language. All basic functions, such as adding, viewing, removing, and changing data through student, tutor, and session data, have been completed. The data structure that handles dynamic data elements throughout the program is the linked-list data structure. When each task is executed, it provides dynamic size and efficient insertion and deletion. The code is divided into two sections, with the menu user interface code written in c and the main function code written in the header. Even though the system is stable and error-free, there is always room for development, such as better data file storage and code searching. Furthermore, this system is unable to erase sessions that have already expired. As a result, I hope to increase my skills and knowledge in order to create a management system that is more advanced, efficient, and functional in the future.

# 6.0 Reference List

*C introduction*. (n.d.). Retrieved from Tutorials: https://www.w3schools.com/c/c\_intro.php

*Understanding the basics of linked list*. (n.d.). Retrieved from GeeksforGeeks: https://www.geeksforgeeks.org/what-is-linked-list/