

**TEAM MEMBERS:**

* ‌KHILFI BIN KHAIRUL AMIN (23039065)
* MUHAMAD IRFAN BIN MUHAMAD NASIR (23038820)
* MUHAMMAD IYAS NAUFAL BIN NAZLIM (23039748)

**LECTURER:**

* MS. HANIN BINTI ABDUL RAHMAN

|  |  |
| --- | --- |
| Aesthetic Minimalist Home Office Workspace Desk Person Working Laptop  Computer Stock Photo by ©maximleshkovich 623985092  USCS20  COMPUTER PROGRAMMING GROUP PROJECT |  |

Table of Contents

[1.0 INTRODUCTION 3](#_Toc151915135)

[1.1 PROBLEM DEFINITION 3](#_Toc151915136)

[1.2 FLOWCHART 5](#_Toc151915137)

[2.0 SELECTED CODES 10](#_Toc151915138)

[3.0 SAMPLE OUTPUT 14](#_Toc151915139)

[4.0 USER MANUAL 15](#_Toc151915140)

[5.0 CONCLUSION 16](#_Toc151915141)

# INTRODUCTION

The demand of efficiency and the increase of number of students every year led to development of student database management system. This student database management system is mainly utilized for a systematic student registration process and keep student records in organized manner. Hence, we have created a proper and well-ordered student database system that is easy to use by the students and admins. The program starts of by asking the user whether they already had registered an account or not. If no, user will go for student registration by creating a username and password. If user had registered an account, they will proceed to entered their username and password assigned for themselves. After one of the processes has done, user will be given 5 different options: (1) Browse; (2) Add more information; (3) Edit existing information; (4) Delete their data; (5) Exit the program. Each option has it own process written by us and it integrates well with each other.

## 1.1 PROBLEM DEFINITION

A Hospital is in need of highly efficient student database system that is able to perform several tasks needed by the user. If the user is a first-time user, then they should start by creating their own username and password. If they have a registered account, they should perform the log in process and create. After any one the tasks mentioned is done, they should proceed on choosing which process they wanna go through from 6 given options.

|  |  |
| --- | --- |
| INFORMATION | STUDENT RECORD |
| FORMULA | Options before authorization:  1. **Register** (Add new user)  a. Student ID (Auto-generated)  b. Full name  c. IC Number  d. Age  e. Program  f. Number of subjects  g. Username  h. Password  2. **Login**  a. Username  b. Password  Options after authorization:  1. **View**  a. Student IDs  b. Full names  c. Ages  d. Programs  e. Number of subjects  2. **Edit**  a. Full name  b. IC Number  c. Age  d. Program  e. Number of subjects  f. Username  g. Password  h. Password verification  3. **Delete**  a. User confirmation (yes/no)  4. **Search**  a. Search key  5. **Exit** |
| DATA | **Register**: Student ID, full name, IC number, age, program, number of  subjects, username and password  **Login**: Username and password  **View**: Student IDs, full names, ages, programs and number of subjects  **Edit**: Full name, IC number, age, program, number of subjects, username,  password and password verification  **Delete**: User confirmation (yes/no)  **Search:** Search key |

## 1.2 FLOWCHART

## 

The full flowhcart

## A close-up of a list of students Description automatically generated

Flowchart for choice 1 (view)

## A black background with white rectangles Description automatically generated

Flowchart for choice 2 (Search)

A diagram of a data processing process

Description automatically generated  
A screenshot of a cell phone

Description automatically generated

Flowchart for choice 3(ADD)

Flowchart for choice 4 (EDIT)

## 

Flowchart for Option 5 (DELETE)

# A white rectangle with black text Description automatically generated

Option 6 (EXIT)

# 2.0 SELECTED CODES

List of library:

* fstream used to read and write files.
* String used for declaring characaters of string.
* Algorithm is used for string processing.

A black background with white text

Description automatically generated

We used an array to create the maximum amount of student it can hold.

A black screen with white text

Description automatically generated

This AuthUI() function will as user whether they’re a new user or they already have an account registered.

A computer screen shot of a program code

Description automatically generated

A screen shot of a computer program

Description automatically generated

As for signUp() function, we included loop and ifstream for account registration. Loop is used to check whether the username they typed already existed or not by using ifstream. Ifstream is used to read files that contains username and password. If user typed in existing username in the account.txt file, it will then prompt a message and ask the user for a different username.

In this part MainUI() functions allow user to choose one of the 6 given options that includes view, search, add ,edit, delete and exit. Each operation is separated with their own designated function by using switch case.

A screen shot of a computer program

Description automatically generated

In this picture, we have separated our options by using different function. Each function has its own process and each one of them will be connected to pressEnterToContinue() function ro return to mainUI() to do another operation by user.

A screenshot of a computer program

Description automatically generated

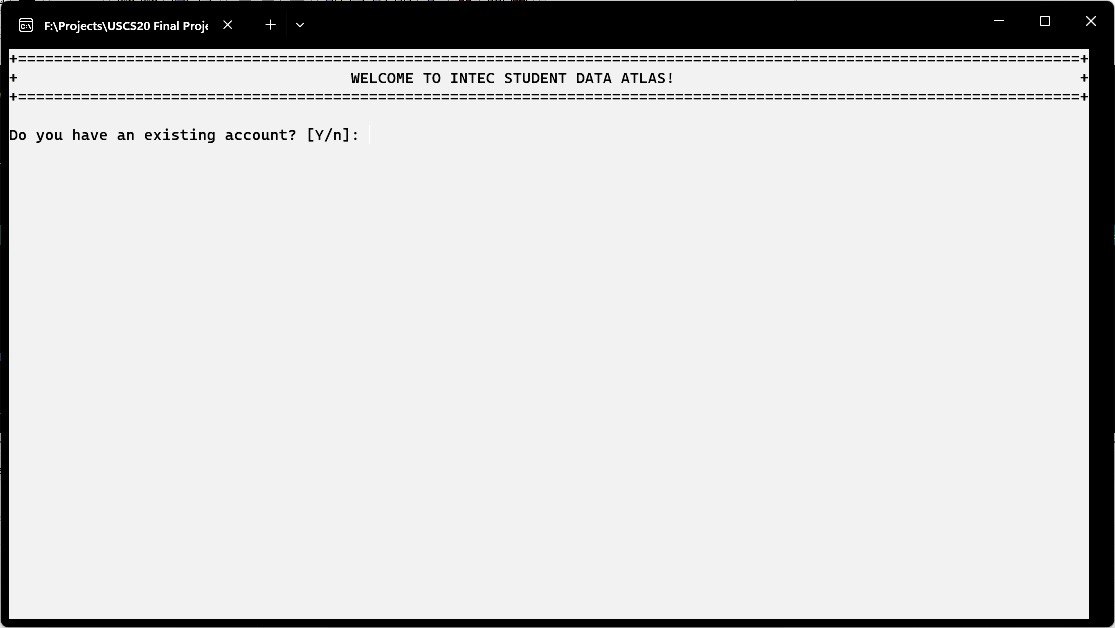
A screen shot of a computer program

Description automatically generatedThe function prototype

# 3.0 SAMPLE OUTPUT

The Welcome Page

User will first see the welcoming page that asks the user whether they have an existing account or not.



If user typed in ‘n’ or ‘N’, the user will be directed to the account registration interface. The system will ask the user to enter a new username and password. The picture below is provided with the input.

A screenshot of a computer

Description automatically generated

If user typed in ‘y’ or ‘Y’, user will be directed into log in page, asking user to input their existing username and password. Input is provided below. (Assume the account has existed)

A screenshot of a computer

Description automatically generated

The Main Menu

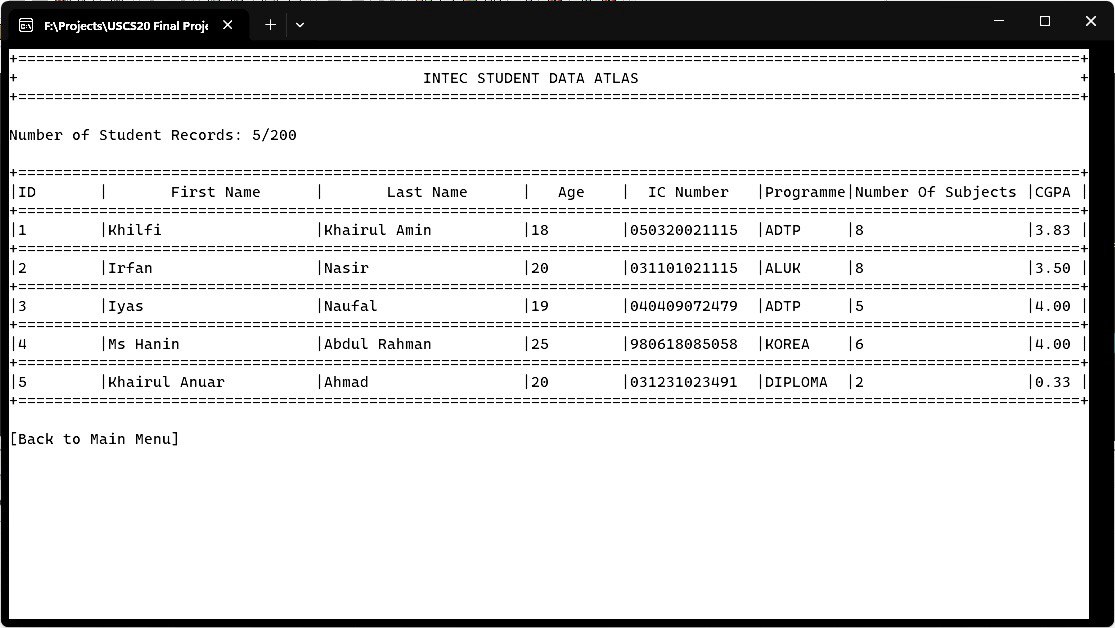
After registration / loggin in, user will have the access to the main menu. In this page, user will be given 6 different options: View, Search, Add, Edit, Delete and Exit. Each option has it designated number in char type assigned to it (1-6).

A screenshot of a computer

Description automatically generated

View Page (Option 1)

All student will be displayed including their first name, last name, age, IC number, programme, number of subjects, and GPA. This page also show the number of students that has registered. The maximum of student that can register is 200 students only.



Search Page (Option 2, ID searching)

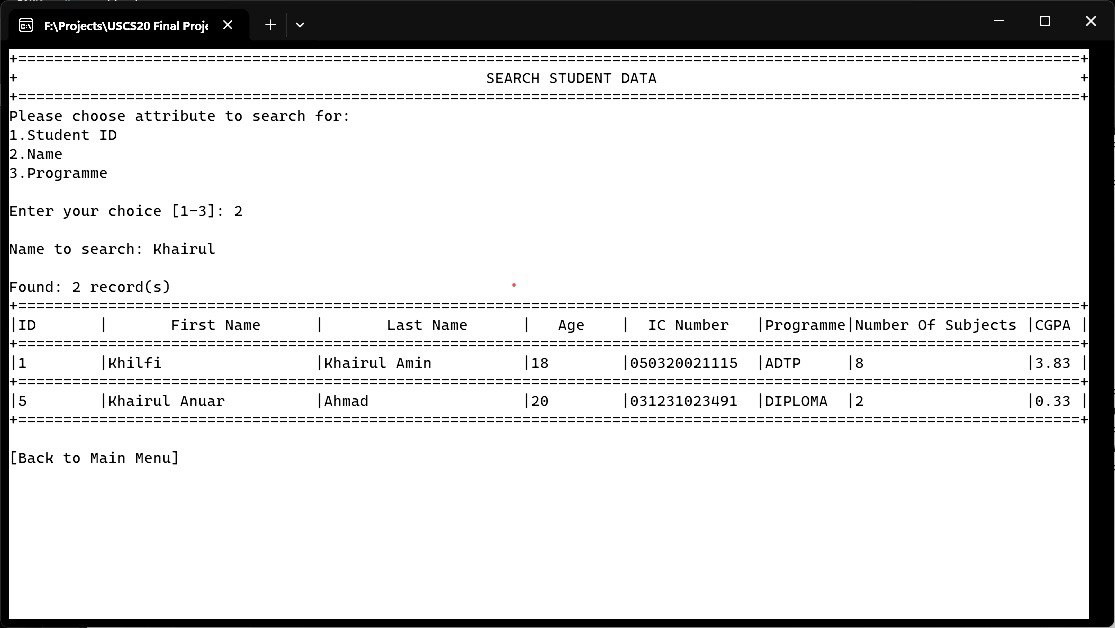
User is able to search their name based on their ID assigned by the system.

A screenshot of a computer

Description automatically generated

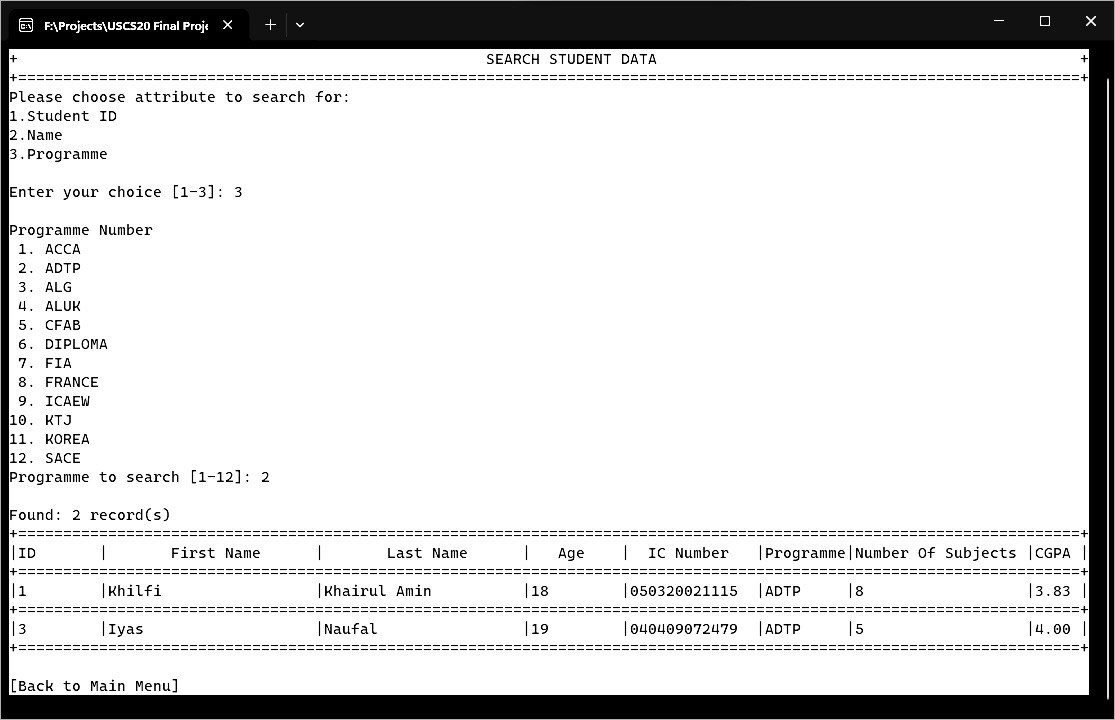
Search Page (Option 2, Name Searching)

User is able to search based on their name. The correct input is needed in order for it to show the desired result.



Search Page (Option 2, Programme Searching)

Lastly, user can also search students according to their programme. A list containing 12 different programmes will be shown. User will be asked to enter one number that matches the programme.



Add Page (Option 3)

On this page, user will be asked to enter first name, last name, age, IC number, Programme Number, number of subjects and CGPA. After all input has been fulfilled, an output containing brand new information will be displayed right away.

A screenshot of a computer

Description automatically generated

Edit Page (Option 4)

User need to enter student ID first before proceed to edit student information.



This then will be followed by displaying current student information. An option will be asked to which information need to be edited.

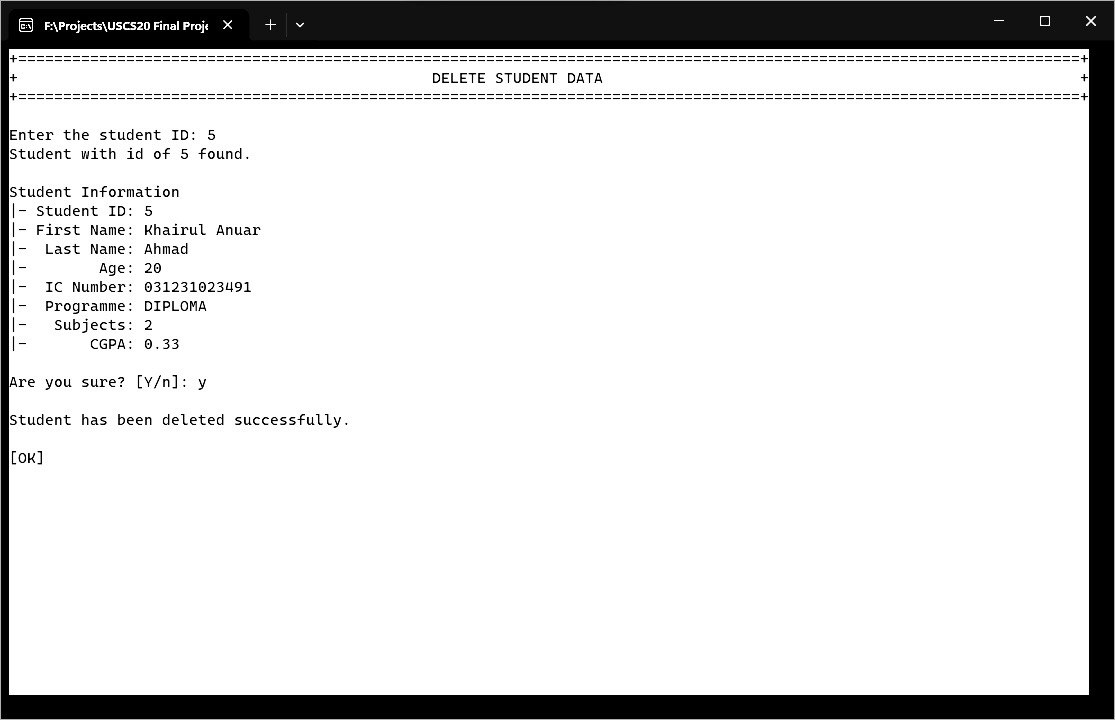
A screenshot of a computer

Description automatically generated

The updated student information will be displayed after editing. User will have the option to continue editing or go back to main menu.

Delete Page (Option 5)

Delete page will first ask user to input student ID. Student information will be displayed if found.



The system will ask the user to confirm deletion by entering ‘y’ or ‘Y’. If ‘n’ or ‘N’ is typed in, the deletion will be cancelled.

# 4.0 USER MANUAL

1. This program will first execute the log in page of the system.
2. After that, a confirmation for existing account is needed to make sure if there is an existing account. If there is no existing account yet so the user will need to enter “N” or “n” and if the user has an existing account, then enter “Y” or “y”

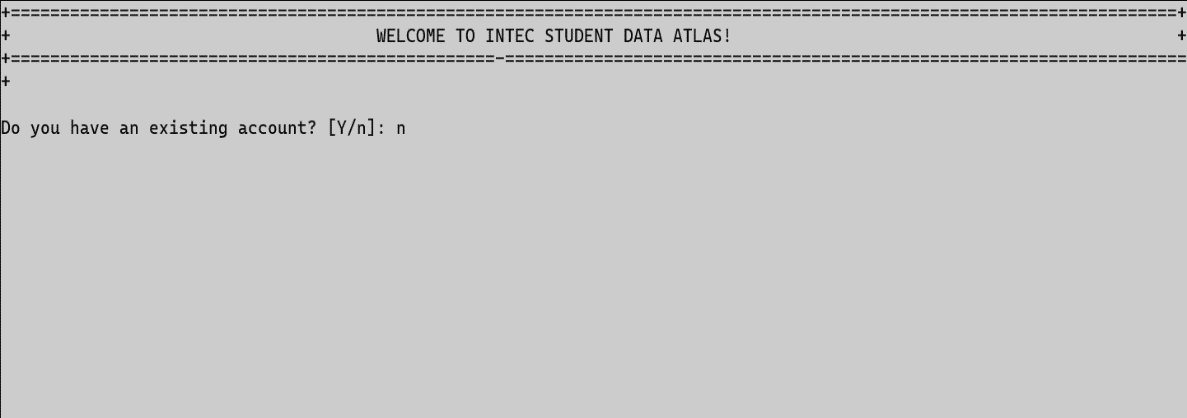


Diagram 2.1 shows the example of input

1. Creating a new account

● First, a username needs to be chosen

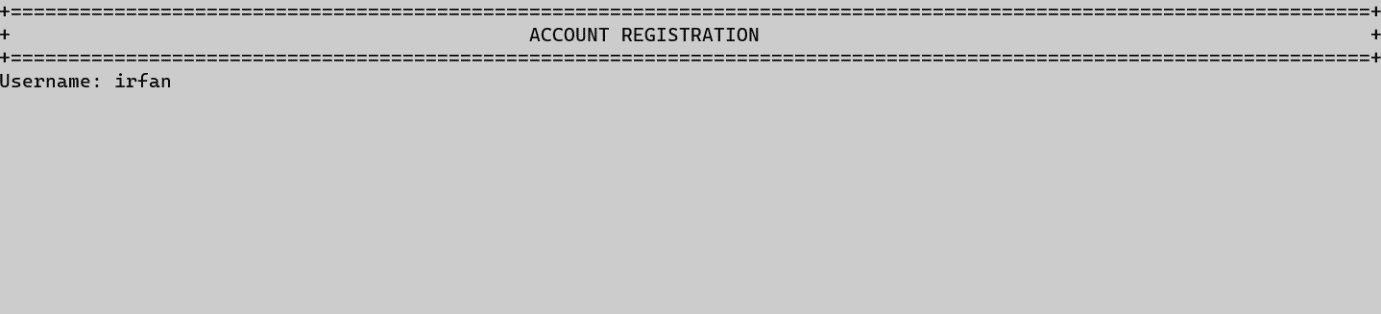


Diagram 3.1 shows example of username

● Next, choose a strong and suitable password

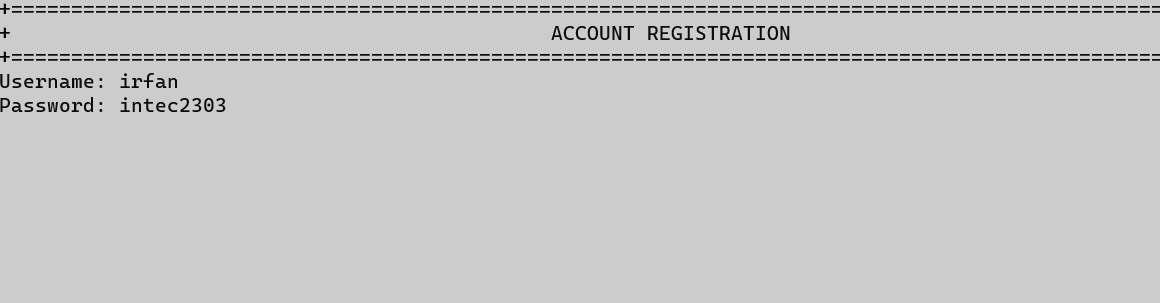


Diagram 3.2 shows example of password

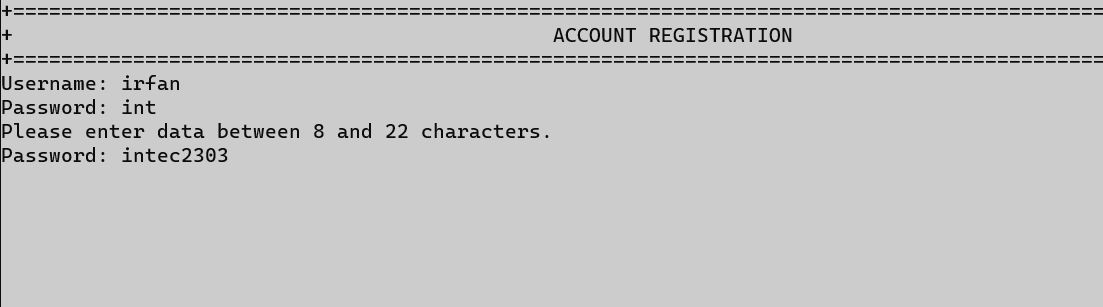


Diagram 3.3 shows example of password that is not valid

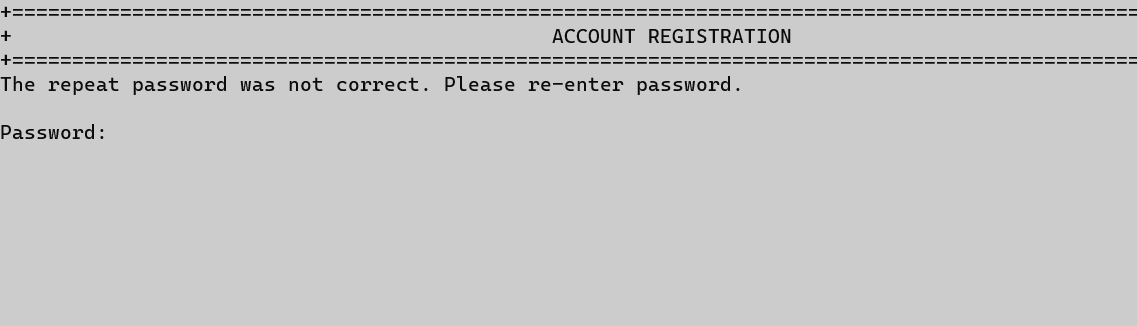


Diagram 3.4 shows example if the password is not matched with the previous one

Note: Please make sure to enter password with 8 to 22 characters and re-enter the password correctly

● User need to make sure a file named accounts is created in the same file location where the main program is saved

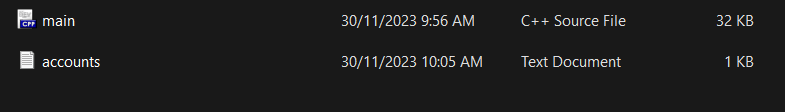


Diagram 3.5 shows the file that has been created named accounts

1. Main menu

● There are 6 options that can be selected from



Diagram 4.1 shows main menu

● Option 1 (View)

- This option allows the user to see all students’ data

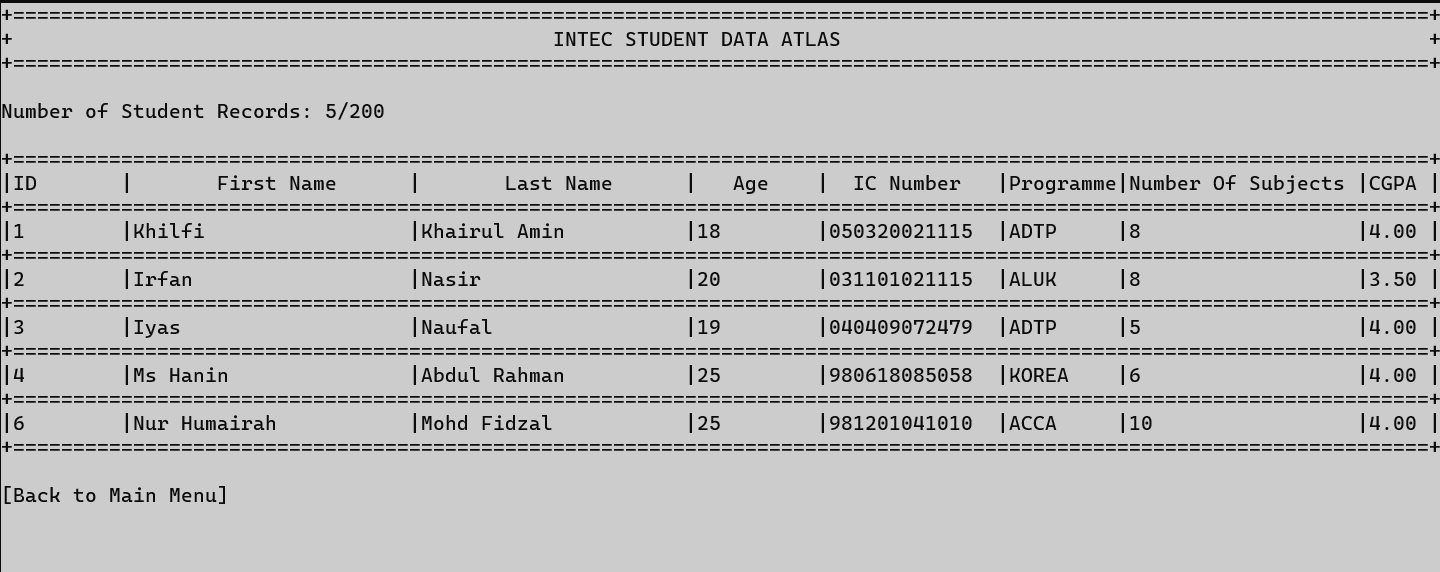


Diagram 4.2 shows main menu

* Press enter to return to the main menu

● Option 2 (Search)

- This option allows the user to search for a student’s data

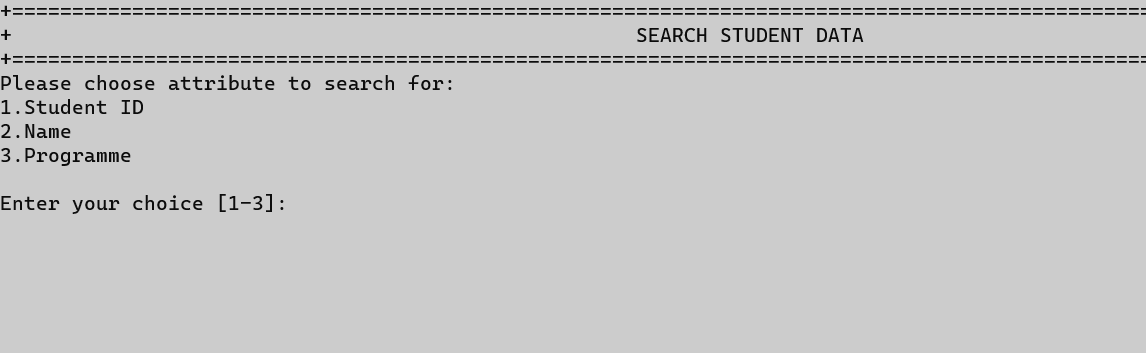
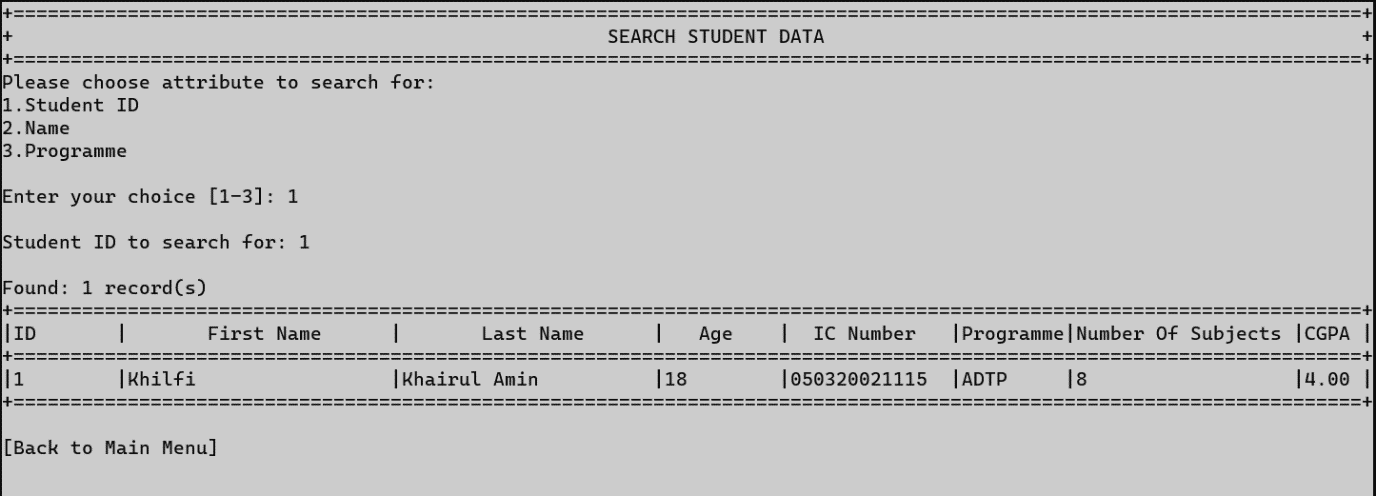


Diagram 4.3.1 shows options for searching student

* There are three options to search list of students’ data.

Diagram 4.3.2 shows result for searching based on student’s ID

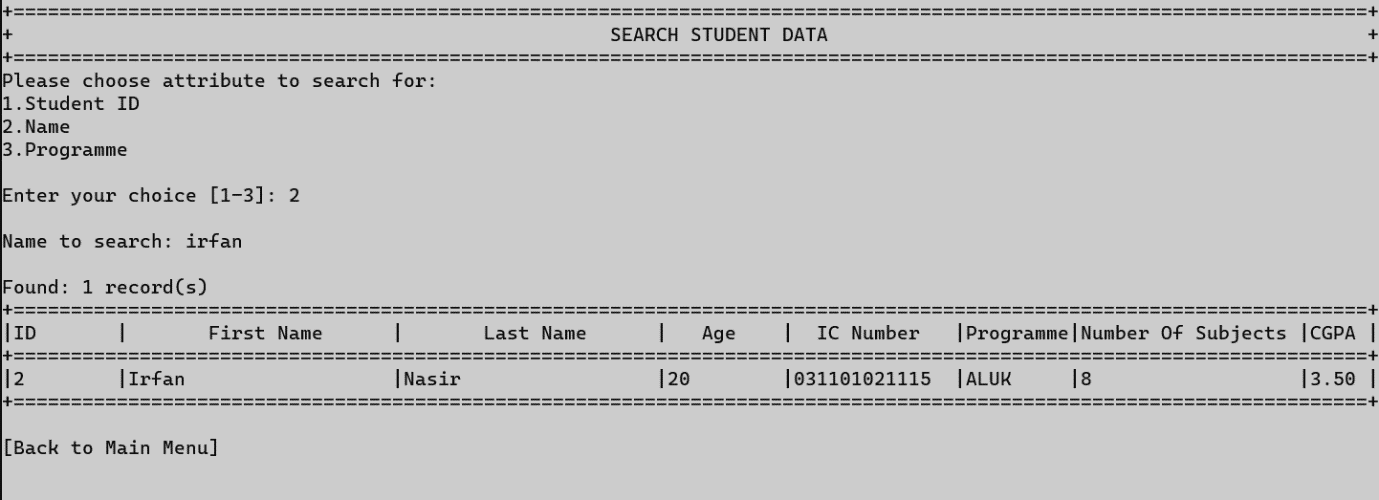


Diagram 4.3.3 shows result for searching based on student’s name

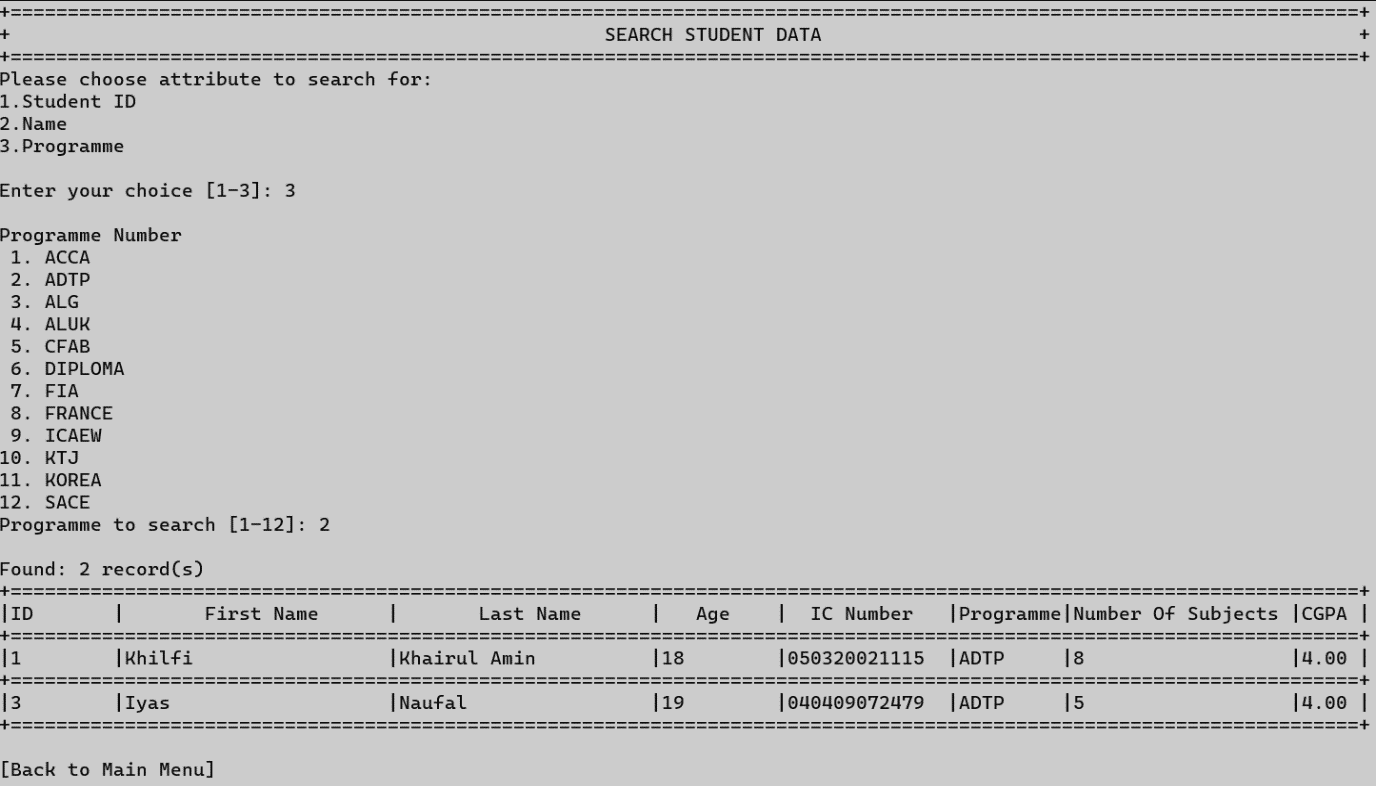


Diagram 4.3.4 shows result for searching based on student’s name

● Option 3 (Add)

* This option allows user to add new student’s data into the system. There are 7 information that are required to be filled which are first name, last name, age, IC number, programme, number of subjects and CGPA

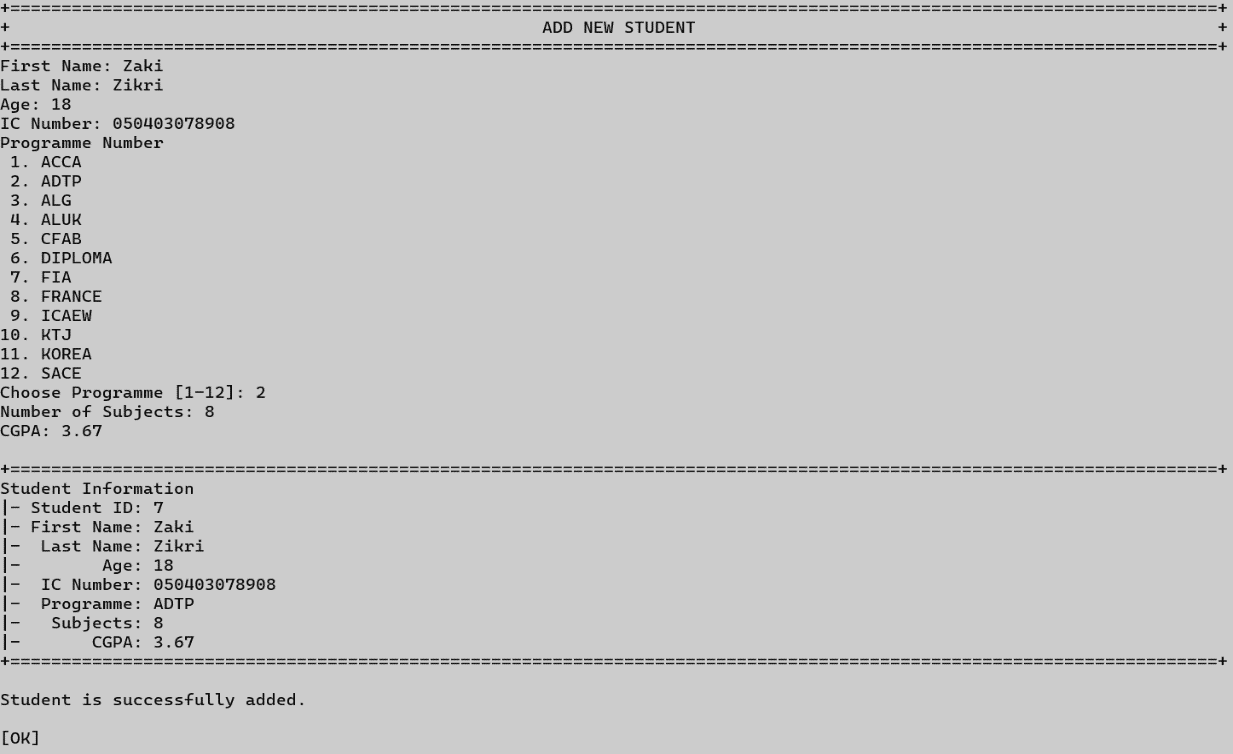
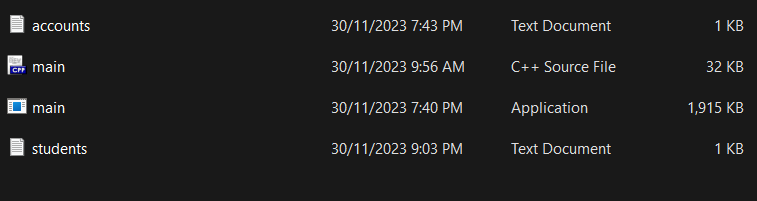


Diagram 4.4 shows result for searching based on student’s name

● User need to make sure a file named students is created in the same file location where the main program is saved



● Option 4 (Edit)

* There are 7 types of information that can be edited which are first name, last name, age, IC number, programme, number of subjects and CGPA
* A student ID must be entered first before any changes can be made

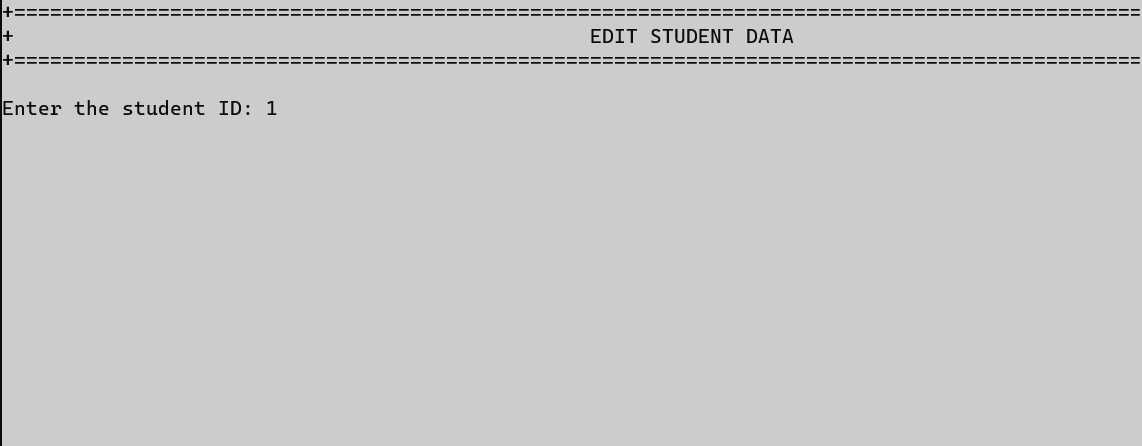


Diagram 4.5.1 shows example of student ID that will be entered

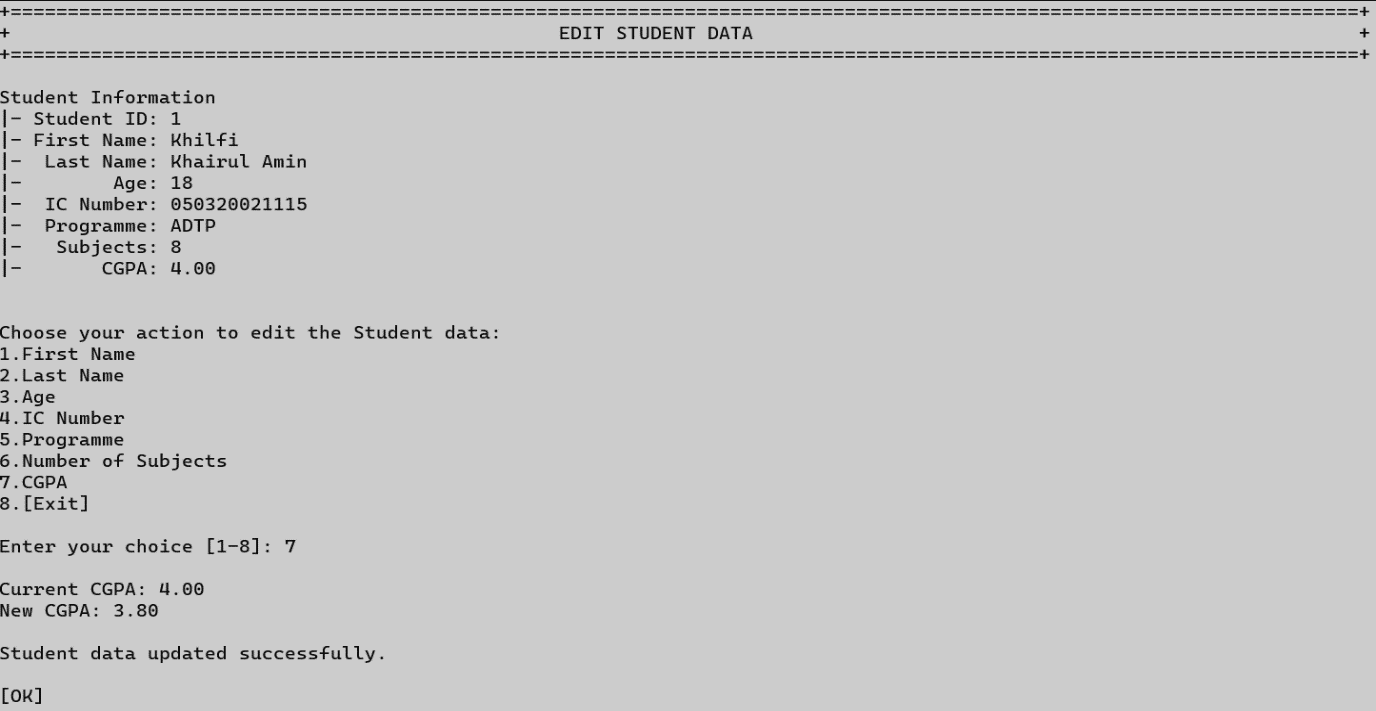


Diagram 4.5.2 shows example of student’s data that has been updated successfully

● Option 5 (Delete)

* This option allows user to delete student data from the system. A student ID must be entered first before deletion can be made. A confirmation message will be shown to get confirmation from the user if the user changed his/her mind.

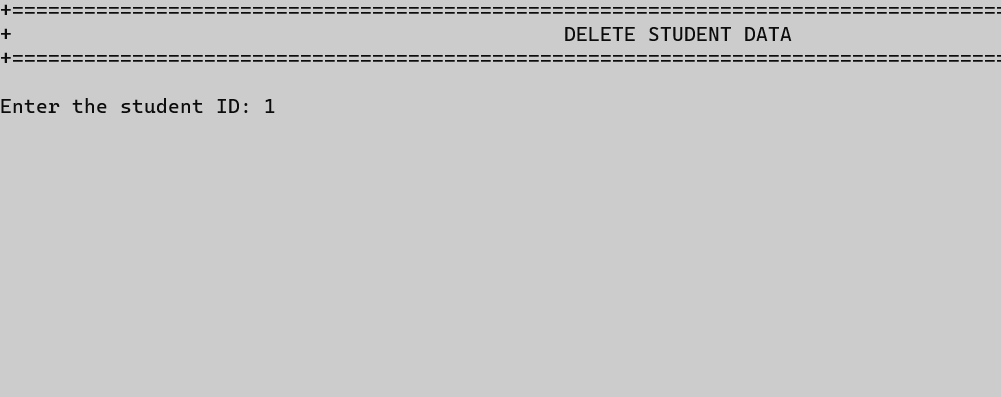
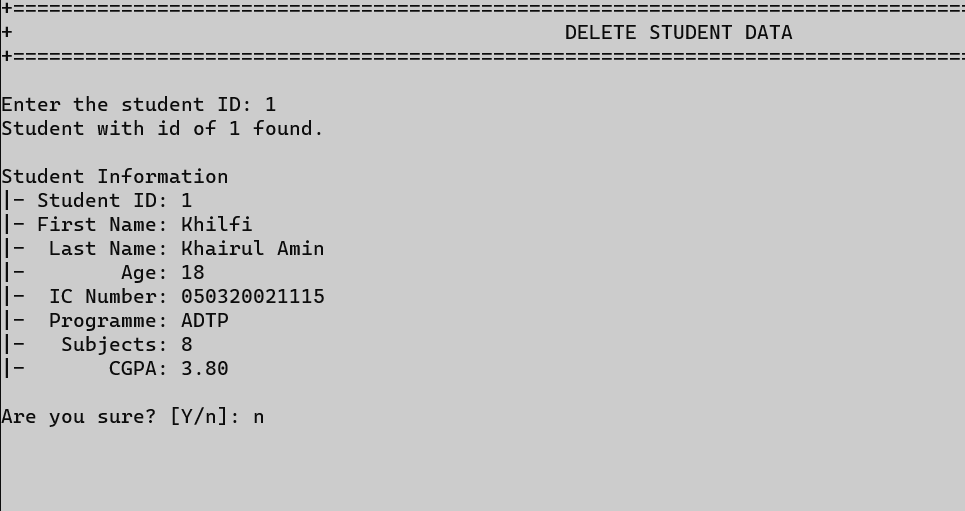


Diagram 4.6.1 shows example of student ID that will be entered

Diagram 4.6.2 shows example if the user changed his/her mind

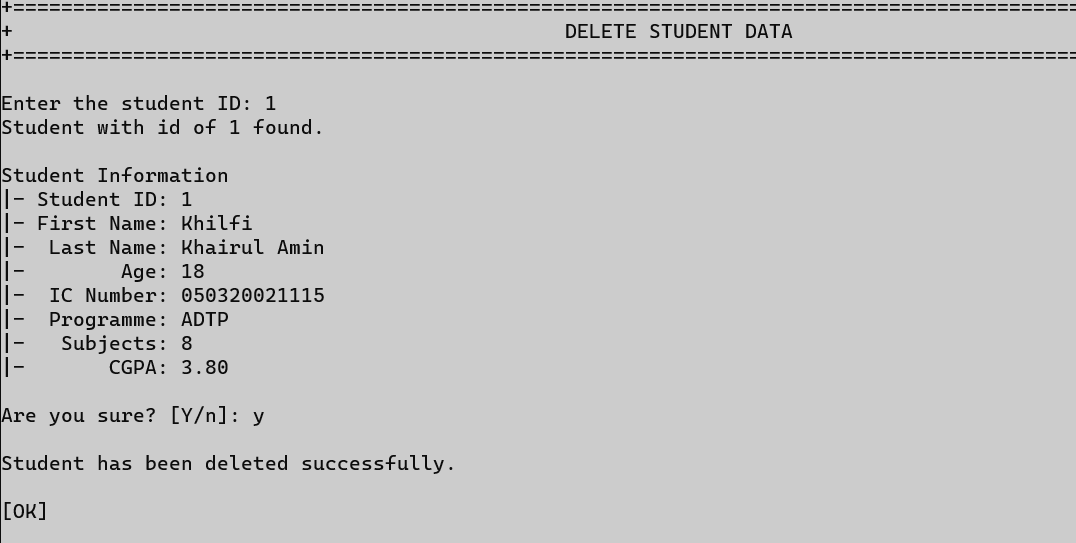


Diagram 4.6.3 shows example of a student’s data has been deleted successfully

● Option 6 (Exit)

* This option allows the user to exit or log out the system

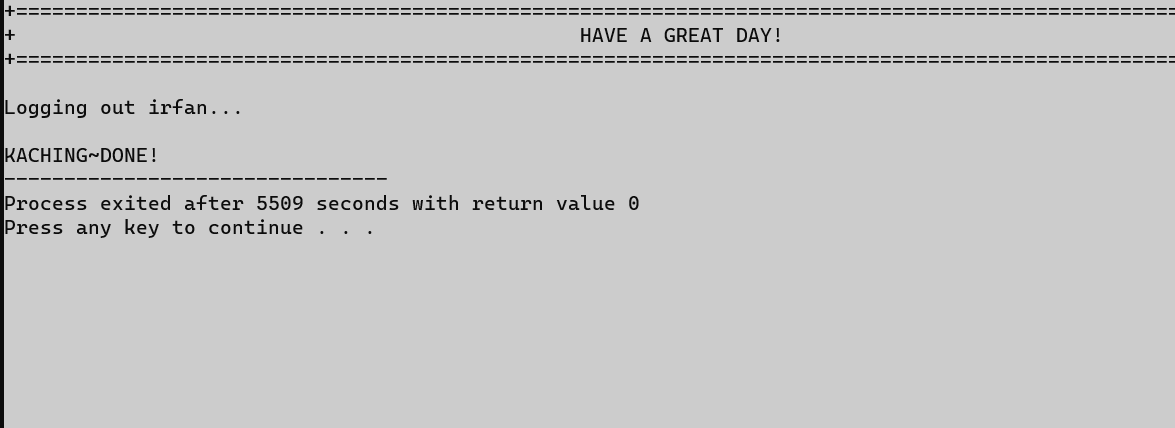


Diagram 4.7 shows example of successful exit or log out

# 5.0 CONCLUSION

In conclusion, the development of Student Database Management System is important to ensure all information of students can be stored in organized manner. It requires us to utilize what we have learned it class including loops, read and write file, function modularity, and others. This project also requires us to use elements we haven’t learned in class to ensure our code can be as efficient as it can. We are able to achieve a successful way of producing hundreds line of codes by using a simple step which is by identifying the problem and the problem definition, develop a solution by using flowchart, code the solution and debug the codes if there’s any error. Moreover, we focuses on asking user’s input in order for the system to run the next step. We asks user for their student Id for each task to standardize it and make sure the coding process is easier. We also includes input validation to ensure user entered correctly and the task will repeat by itself. We are in high hopes that out developmental result can help the school to organize their student information correctly.