User Manual

**First of all**, the user should provide a valid input file which contains students names, ids, absence, midterm, classwork, and final in the following format:  
ID : Name : Absence : Midterm Grade : Classwork Grade : Final Grade

The valid file and the valid input should contain:

ID should be a 9 digit number only  
Absences should be an integer number with 2 digits at most  
Midterm Grade should be a number between 0 and 35  
Classwork Grade should be a number between 0 and 25  
Final Grade should be a number between 0 and 40

The program will prompt the user to enter a file name.  
If one or more of the conditions above was violated, the user will get an error and will have to reenter the file name with the correct data this time

**Remember**: All input should obey all the conditions above.  
If the user enters any non-valid input in the program, a proper message will appear, and he will be returned to the main list.

After running the program, the user will have to enter a number from 1-7 or 8 to reprint the list, else he will get an error

...

**For number 1**

If the user press 1 he will have an output file that contains names, ids, and total mark,  
Name : ID : Total mark, which is equal to **Final Grade + Midterm Grade + Classwork Grade** out of 100, after everything is completed the user will get a *“Done”* message.

...

**For number 2**

If the user press 2 he will have to enter **a** or **b** or **c** or **d**.

1. to search by absence
2. to search by Midterm
3. to search by classwork
4. to search by Final grade

Then the user should provide the case sign if it > or < or =, then he will enter a number which he wants to search about. Finally, he will see if there is a student who met this criterion by printing all the student information else “There's none” message will be printed.

**For number 3 (for adding student)**

If the user press 3, he should enter the student id he wants to add and following by the student name, absences, midterm grade, classwork grade and final grade.

**For number 4 (removing a student)**

If the user press 4, he should enter

1. Removing by id
2. Removing by name

else will receive an error message.

If he presses **a,** he has to enter an id from the input file else he will receive *“the enter id was not found”* message. The program will print the student information and ask the user for confirmation. the user should put **y** to confirm removing him or anything for no.

Or by name by pressing **b** he should enter a name from the input file else he will receive *“the entered name was not found”* message. when he put a correct name, the program will print all student with the same exact name, if there are more than one student with the same name the program will ask

1. Remove all of them.
2. Retry with the ID

the user should put **y** to confirm removing him/them or anything for no.

...

**For number 5 (modifying student information)**

If the user press 5, he should enter the student id that wants to modify then he will have to choose from a to f

1. to change the id
2. to change the name
3. to change the absence
4. to change the midterm
5. to change the classwork
6. to change the final grade

If he chooses any letter, he will enter a new id or name or absence or midterm or classwork or final.

...

**For number 6 (show the data for specific student)**

If the user press 6, he should enter the student id that wants to see his information then it will appear the id should be correct id else he will get an error message.

...

**For number 7(terminate a program)**

If the user press 7 the program will stop immediately.

For function one (about total)

First, it will open the input file then The program will create an output file name totalmark.txt then we use for loop to read all the input file then we create a list which contains name ID and total mark then write it in the output file

Function two to add student

First, we will open a file then require from user to input ID and students name and absence and midterm and classwork and final grade then it will take the student ID and absence classwork midterm and final to validate function

If it’s not none type if it’s correct it will write the student ID and students name and absence and midterm and classwork grade and final grade in input file then the program will require from the user if he wants to do total mark again by putting y else for no

Function three to remove a student

First, we will open the input file then student ID in the dictionary and students name in another dictionary then we have open list and for to read all input file then we will define that position zero for ID and position one to the student's name

In function for modifying by ID

First, we open a file then create a dictionary that name student then read all the input file

Here we use .lower() to change all the input from the user to a small letter

Also, we use pop by position and insert

In function show

First, we open the input file then create a dictionary name students then read all the input file then require from user to enter the ID for the student then see if it’s in the student's dictionary or not

If it correct it will show all the information for the specific student if it’s not the program will print no ID like this in the file and returning to the main list

Function criteria

First, we open the input file then create a dictionary name students then read all the input file then require from the user to enter the letter from a to d then the user will input the sign he wants><= then he will enter a float number then the program will see if the file has the same number that he choose If it’s a yes then the program will print all the student that follow the criteria

Validate function

first, it will see if the object not equal to none if it’s none it will check about the main and the file and also The validation equal to integer and validation equal to float and validation equal midterm and validation equal classwork and validation equal final and validation equal ID

Function file to require from the user to input file name then it will go to function validate to see if it none or not if it none type the user will enter a file name until it is correct

Function start it’s to give the user all choices

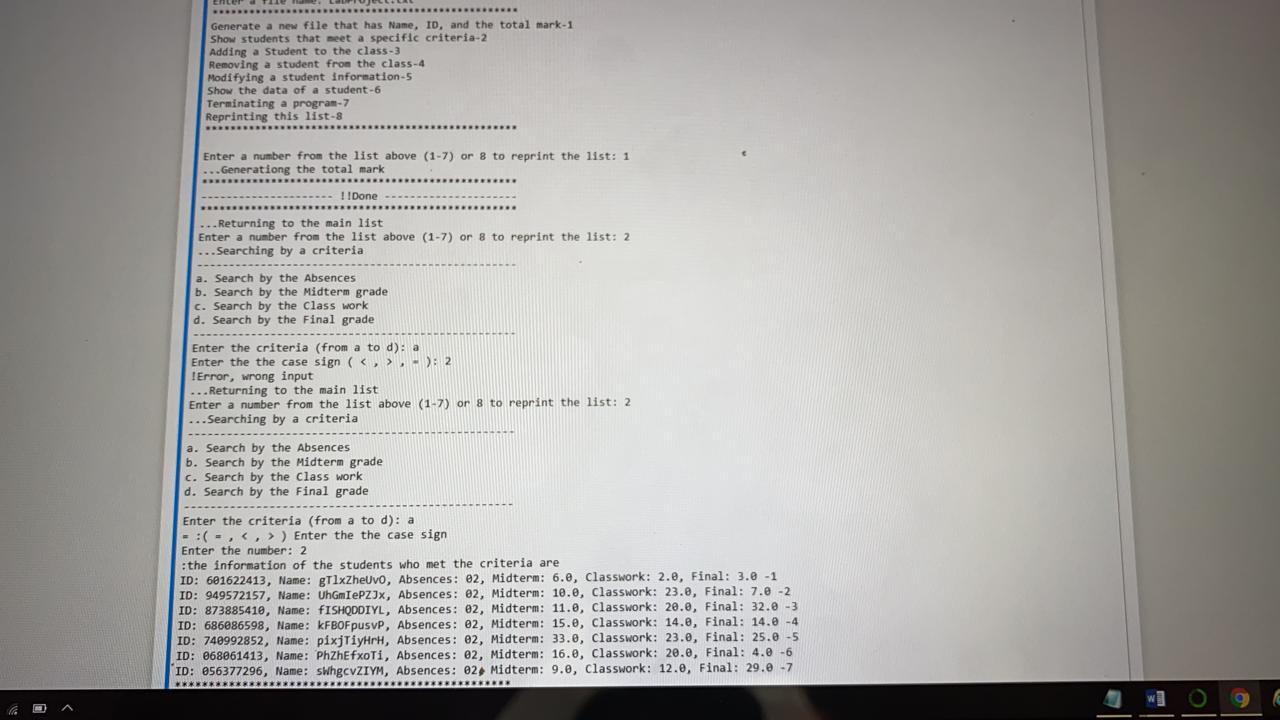
Function main

First it will defined the name of the file and then the choice if it’s an integer or not or from 1 to 7 or not if he chooses number one the function will go to the function number one and choose two for function two and continue until 7

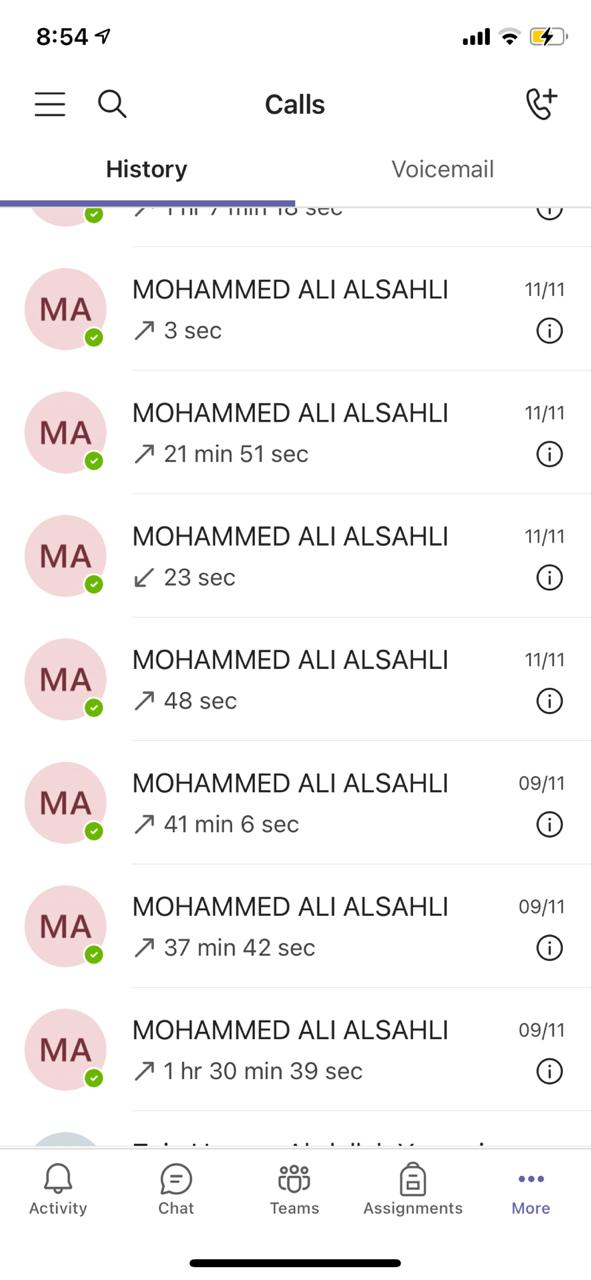
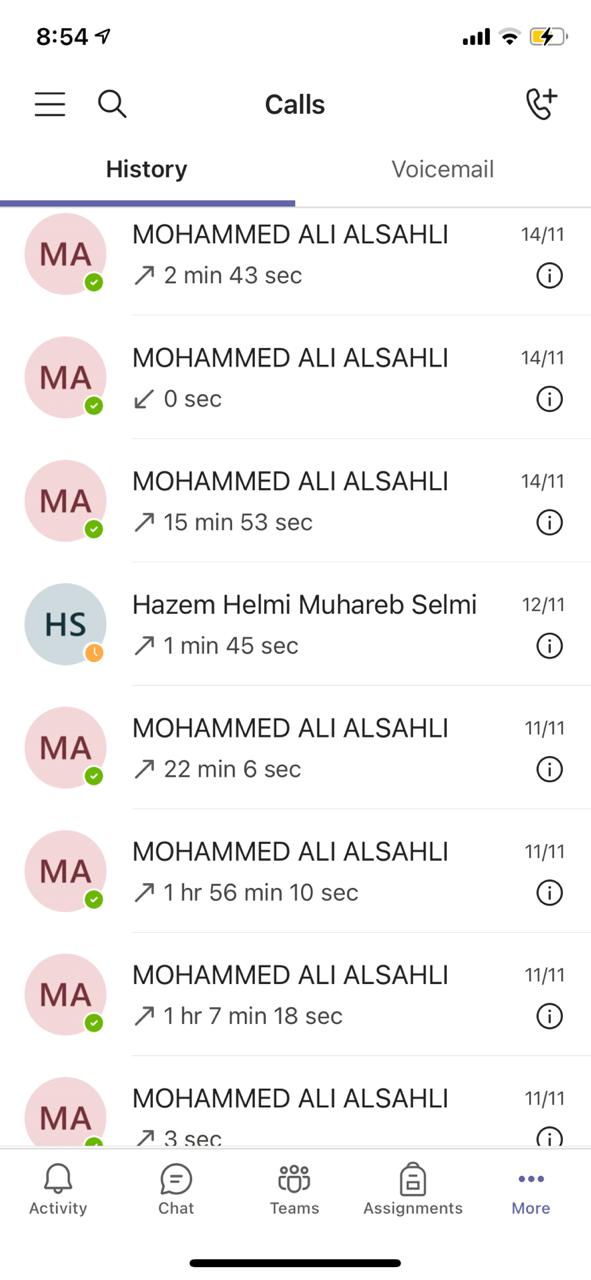
The biggest problem:

The none type

We solve it by if condition (if! = none)



**In this program Mohammed and Abdulmajeed worked together**

****

