INTRODUCTION TO ALGORITHM (ITS30705)

GROUP PROJECT

Title: Exam Marks and Grading System

Group: CHINDIANS
Members:
Isaac Lee Jo Min (0353165)
Jason Chin (0353423) (Group Leader)
Jeffrey Lau Wen Kang (0353187)
Lim Kean Yee (0352896)
Surryaraj Poobalan (0353127)

Roles and Responsibilities

For this assignment, our group created an exam mark and grading system program. We distributed the roles and responsibilities of this project respectively among the five members of our group. All five members were tasked to develop a section of the code for the program. Jason Chin developed the code to check the student IDs, create new student IDs, and link it to the first user interface. Isaac Lee Jo Min wrote the code that calculates the total grade, the marks table update, and the end system sequence. Lim Kean Yee and Jeffrey Lau Wen Kang were both assigned to create a code to calculate the GPA for each subject and the overall CGPA of the student. Surryaraj Poobalan wrote the code that updates the text file, in this case, the student database. Besides that, Jason and Isaac designed the flowchart, and Surryaraj, Jeffrey, and Kean Yee wrote the report.

Program Features

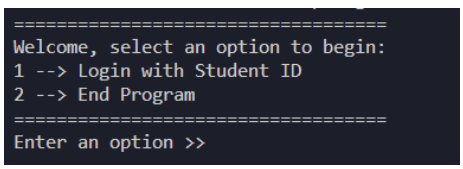
We came up with several features for our Exam Marks and Grading System. The first feature is a login system that can detect whether a student ID is already in the database. If the student ID entered does not exist, the user can input the seven-digit student ID to save it into the database with default values. Secondly, entering marks into the database. The user can input new values or modify the existing values for each subject.

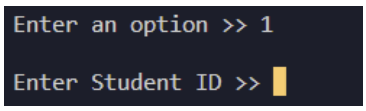
Moreover, the third feature displays the grade for each subject taken and the marks for those subjects alongside it. The fourth feature carries out a calculation that sums up the overall grade and marks to calculate the GPA and CGPA of the student. After summing up the total and the average, the GPA and CGPA values are displayed. Last but not least, the fifth feature allows the user to log out, which then they will be brought back to the login page if needed to log in with another student ID.

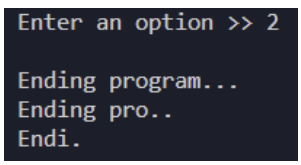
There is also a feature that allows the user to end the program after completing it, both on the log-in page and the marks editing page.

Algorithms Used in Developing the Program

After executing the code, the user is directed to the login page. Here, the user can enter from 2 of the available options "1" or "2". If "1" is entered, the program will ask the user to input a student ID not bound to any limitations of character length whereas if the user enters "2", it ends the program.







After inputting "1" and entering the student ID, the application checks to see if the ID exists in the database. If the ID is not found in the database, the program will prompt the user to enter "Y" if they wish to register it as a new ID or "N" if they do not. If "Y" is selected, the program will then prompt the user to provide the first name and surname to bind with the following student ID before going to the second user interface. If "N" is input, the user is returned to the login page. Consequently, if the entered student ID is already in the database, the program proceeds to the second user interface.

```
Enter Student ID >> 0115932
That ID doesn't exist, would you like to register this new ID? (Y or N) y
Enter First name >> Test
Enter surname >> Subject
```

```
13    0115932 // TEST // SUBJECT // ungraded // ungraded // ungraded // ungraded
```

The second user interface displays a menu that consists of eight options and the current marks following the student ID. If the user enters "1", "2", "3", or "4", the program prompts the user to enter a value from 0 to 100. The entered value will then be updated in the database following the student ID. When the option is "1", the entered value will update the existing mark for the Algorithm subject, in this case, the fourth element of the database line. The same goes for "2", "3", and "4". "2" updates the mark for Computing Mathematics in the fifth element of the database line, "3" updates the mark for English in the sixth element of the database line and "4" updates the mark for the Web Programming subject in the seventh element of the database line. If the entered value exceeds 100 or is lower than 0, “ERROR: Enter a valid mark” is displayed and the value would not be updated into the database.

```
Enter Student ID >> 0123456
Select an option:
    1 --> Enter marks for Algorithm
    2 --> Enter marks for Computing Mathematics
    3 --> Enter marks for English
    4 --> Enter marks for Web Programming
    5 --> Display marks and grades
    6 --> Display GPA and CGPA
    7 --> Log out
    8 --> End the program

Enter your option >>
```

```
Enter your option >> 1

Enter your mark here => 24
```

```
Enter your option >> 2

Enter your mark here => 101
ERROR: Enter a valid mark!
```

If the user enters "5", the program will display the grade for each subject taken as well as the marks for each of those subjects. If the mark for a subject has yet to be entered, "No Grades entered" will be displayed for the grades and the marks will be kept as ungraded. If the user enters "6", the program will display the GPA for each subject and the CGPA. If a subject's grade has not yet been recorded, an error message stating "No grades are entered" will be shown. As a result, the GPA for subjects with marks will be displayed, however, the GPA for subjects without marks, as well as the CGPA, will not be displayed.

```
Enter your option >> 5
Marks-->
        Algorithm : 24
        Computing_Mathematics : 56
        English : 70
        Web_Programming : 70

Grades-->
        Grade for Algorithm is F
        Grade for Computing Mathematics is C+
        Grade for English is B+
        Grade for Web Programming is B+
```

```
Enter your option >> 6
GPA for Algorithm is 0.00
GPA for Computing Mathematics is 2.33
GPA for English is 3.33
GPA for Web Programming is 3.33
==================
Your CGPA is 2.25
==================
```

If the user was to enter “7”, the program would simply bring the user back to the menu where they can choose to enter a new student ID, essentially logging them out. To which the program can basically be restarted. Lastly the final option would be “8”, in which the user is able to straight up end the program.

```
Enter your option >> 7
==================================
Welcome, select an option to begin:
1 --> Login with Student ID
2 --> End Program
==================================
Enter an option >>
```

```
Enter your option >> 8

Ending program...
Ending pro..
Endi.
```

Reference

1. Chempaka Seri Abdul Razak. (2022). Chapter 1 - Introduction to Problem Solving Using Algorithm [PowerPoint slides]. School of Pre-University Studies Taylor's College.

   https://mytimes.taylors.edu.my/mod/resource/view.php?id=235161

2. Chempaka Seri Abdul Razak. (2022). Chapter 3 - Operator, Function, Parameter and Return_ STD [PowerPoint slides]. School of Pre-University Studies Taylor's College.

   https://mytimes.taylors.edu.my/mod/resource/view.php?id=261529

3. Chempaka Seri Abdul Razak. (2022). Chapter 4 - Conditional Execution [PowerPoint slides]. School of Pre-University Studies Taylor's College.

   https://mytimes.taylors.edu.my/mod/resource/view.php?id=261836

4. Chempaka Seri Abdul Razak. (2022). Chapter 5 - Chapter 5 -Logic and Indefinite Loops [PowerPoint slides]. School of Pre-University Studies Taylor's College.

   https://mytimes.taylors.edu.my/mod/resource/view.php?id=266157

5. Chempaka Seri Abdul Razak. (2022). Chapter 6 - List, Set, Dictionary, Tuple [PowerPoint slides]. School of Pre-University Studies Taylor's College.

   https://mytimes.taylors.edu.my/mod/resource/view.php?id=270394

6. Chempaka Seri Abdul Razak. (2022). Chapter 7 - File Input or Output [PowerPoint slides]. School of Pre-University Studies Taylor's College.

   https://mytimes.taylors.edu.my/mod/resource/view.php?id=273532

7. Refsnes Data AS. (2019). Python File Write. W3schools.com.

   https://www.w3schools.com/python/python_file_write.asp