1 Overview

The problem I have identified to solve in this project is knowing what my final mark is for my degree and module ahead of time. This helps in personal planning, as I am able to set myself more realistic goals if I know what my current status is with a module or with my degree.

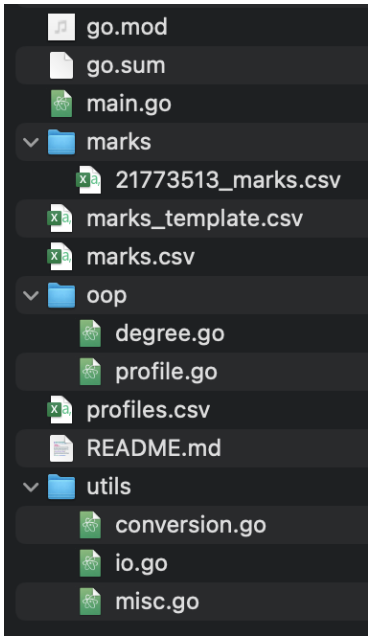
Thus, the solution that this application provides is to calculate a user's real time mark for their degree and their modules.

2 Program structure

This project consists of a number of files:

1. `main.go` Is the main program.
2. `oop/degree.go` Contains structs and functions related to a degree, such as the degree, module, and component structs, and functions to calculate appropriate marks.
   (a) A degree has a name, mark, and list of modules.
   (b) A module has a name, mark, and a list of components. It is assumed that all modules weigh the same amount to the degree.
   (c) A component (such as an assignment or test) has a mark and weight.
3. `oop/profile.go` Contains the struct related to the user's profile.
4. `utils/conversion.go` Consists of conversion helper functions used by the main and oop packages.
5. `utils/io.go` Consists of Input/Output helper functions used by the main and oop packages.
6. `utils/misc.go` Consists of miscellaneous helper functions used by the main and oop packages.
7. `marks/` Contains the program output, with users with profiles' output being in the format of `<username>_marks.csv` and anonymous users having their output saved in `marks.csv`
8. `profiles.csv` Contains a list of the usernames of the created profiles.
9. `marks.csv` Is the default input for the program's marks csv.
10. `marks_template.csv` Is an example input that the user can consult for the creation of their own .csv such that it is in a format that the program can read.

Below is a screenshot of the above-mentioned folder structure:



3 Design Decisions

This is a terminal-based program which has been tested on `MacOs`. The expected input is a csv, with no spaces allowed due to the way that the go code reads csvs. The whole program is wrapped in a loop, such that the user can view and create multiple profiles / marks.

4 Understanding the code

4.1 User flow

The user is first greeted by program and displayed the main menu (start program or see csv input instructions).

```
Welcome to my Mark Calculator!
This program allows you to know your final mark for all your modules and your degree in real time.
You may enter 'exit' or 'quit' at any input point if you wish to quit the program.
----------------------------MAIN MENU-----------------------------------
        Enter 0 to start
        Enter 1 to get the guidelines of how your csv must look like
        Enter exit to quit the program
```

Then they are asked whether they have a profile, wish to create a profile, or continue as an anonymous user.

```
Do you have a profile?
        Enter 0 if you have a profile,
        Enter 1 if you would like to make a new profile,
        Enter any other key to continue without a profile
```

4.1.1 Has Profile

If they have a profile, then they are asked what their username is. Username validation takes place through the program checking the `profiles.csv` file to see whether the username is listed.

If it is not listed, then the user is allowed to enter their username again in case they misspelled it, or they can choose to go to the main menu, or exit the program.

If the username is found, then the user user is welcomed back and is presented with the option to view their results, or update their results with a new csv. After they have chosen their option, they can query their profile options again, or choose to view another profile's marks. They can also exit the program at any time.

```
Welcome back, 21773513!

--------------------------PROFILE INPUT MENU---------------------------------
        Enter 0 to view your results
        Enter 1 to update your results (import a new csv with your updated results)
        Enter any other key to go back to the main menu
0
Your overall degree mark: 53%
        Advanced_Information_Systems_Theory_and_Practice: 76%
        Computing_in_Information_Systems: 50%
        Information_and_Knowledge_in_Organisations: 86%
        Research_Assignment:_Information_Systems_Management: 0%

--------------------------PROFILE INPUT MENU---------------------------------
        Enter 0 to view your results
        Enter 1 to update your results (import a new csv with your updated results)
        Enter any other key to go back to the main menu
1

Enter the relative path of your csv file
        If you have a profile the default is set to marks/<username>_marks
        The default is marks.csv if you just created a profile, or are an anonymous user

Your overall degree mark: 53%
        Advanced_Information_Systems_Theory_and_Practice: 76%
        Computing_in_Information_Systems: 50%
        Information_and_Knowledge_in_Organisations: 86%
        Research_Assignment:_Information_Systems_Management: 0%

Writing your results to .csv...

--------------------------PROFILE INPUT MENU---------------------------------
        Enter 0 to view your results
        Enter 1 to update your results (import a new csv with your updated results)
        Enter any other key to go back to the main menu
```

4.1.2 Create Profile

The user is asked for a username, and this username is checked to be unique. If it is not unique, then the user is allowed to enter another username, or they can choose to go to the main menu,

or exit the program.

After this, they are requested to input the path to the csv file with their marks (in the format of the example `marks_template.csv`). If the file is valid, then they their mark is calculated and outputted both to the terminal and to a csv in the `marks/` directory with their username prefixing the output file. The user may then choose to go to the main menu or exit the program. If the file is not valid, then the user is provided an appropriate error message and the program may ask them for another try or it may exit.

```
y
----------------------------MAIN MENU-----------------------------------
        Enter 0 to start
        Enter 1 to get the guidelines of how your csv must look like
        Enter exit to quit the program
0

Do you have a profile?
        Enter 0 if you have a profile,
        Enter 1 if you would like to make a new profile,
        Enter any other key to continue without a profile
1

Great, let's create a profile for you!
Choose a username:
Richard

Hi, richard! Happy to have you here!

Enter the relative path of your csv file
        If you have a profile the default is set to marks/<username>_marks
        The default is marks.csv if you just created a profile, or are an anonymous user

Your overall degree mark: 53%
        Advanced_Information_Systems_Theory_and_Practice: 76%
        Computing_in_Information_Systems: 50%
        Information_and_Knowledge_in_Organisations: 86%
        Research_Assignment:_Information_Systems_Management: 0%

Writing your results to .csv...

Would you like to calculate another profile's mark?
        Enter 'Y' if you do
        Enter any key to exit the program
exit

Thank you for using my mark calculator!
```

4.1.3 No Profile

If the user chooses to remain an anonymous user, then their user flow is the same as the Create Profile user, except they are not asked for a username, and the output .csv is always `marks/marks.csv`.

	A	B	C	D	E	F	G	H
1	BComHons_Information_Systems_Management	0						
2	Module_Name	FinalModuleMark	Mark1(%)	Weight1(%)	Mark2(%)	Weight2(%)	Mark3(%)	Weight3(%)
3	Advanced_Information_Systems_Theory_and_Practice	0	75	50%	75	25%	77	25%
4	Computing_in_Information_Systems	0	100	50%	0	50%		
5	Information_and_Knowledge_in_Organisations	0	96	50%	77	50%		
6	Research_Assignment:_Information_Systems_Management	0	0	100%				

Figure 1: marks_template.csv

4.2 Input

This program's main expected input is a csv format in the same format as the example `marks_template.csv`. The user is informed of the following guidelines for the creation of the input csv:

Use the `marks_template.csv` provided to get an idea of what your file should look like.

NOTICE:

1. There may be no spaces, commas, or full stops in your csv whatsoever.
2. All marks have to be integers.
3. The first row is for degree details.
   (a) The first column is for the degree name.
   (b) The second column is a placeholder for the degree final mark (which this program will calculate).
4. The second row is for the column headers.
   (a) The first 2 columns are for the `ModuleName` header and `ModuleFinalMark` header.
   (b) The other columns are for the component mark-weight combinations. You may add/remove mark-weight column pairs.
5. The (first) column under the `ModuleName` header is reserved for the Module names.
6. The (second) column under the `ModuleFinalMark` header is reserved for the calculated final marks. These are set to 0 by default.

As can be seen with the example `marks_template.csv` in Figure 1, the final marks are zero. These fields will be calculated by the program and outputted both to the terminal and the user's csv file.

4.3 Output

This program's main output is that of the user's current marks to the terminal, as well as to a csv stored in `marks/`.

As can be seen with the example `marks/marks.csv` in Figure 2, the program outputs the final marks of the user, with the assumption that each module has the same weighting toward the degree.

	A	B	C	D	E	F	G	H
1	Degree:BComHons_Information_Systems_Management	53						
2	Module_Name	FinalModuleMark	Mark1(%)	Weight1(%)	Mark2(%)	Weight2(%)	Mark3(%)	Weight3(%)
3	Advanced_Information_Systems_Theory_and_Practice	75	75	50%	75	25%	77	25%
4	Computing_in_Information_Systems	50	100	50%	0	50%		
5	Information_and_Knowledge_in_Organisations	86	96	50%	77	50%		
6	Research_Assignment:_Information_Systems_Management	0	0	100%				

Figure 2: marks/marks.csv

4.4 Error handling

This program is quite robust in handling user input. If the user inputs an invalid input, then the program generally tells the user what the expected input it, and gives them the option to try again, go to the menu, or exit the program.

5 Future functionality

Of course, there this program can certainly be expanded. If I had more time, it would be nice to implement features such as:

1. **GUI:** A GUI would great to make this program more interactive and accessible. It could include features such as a file uploader and allow the user to edit the csv file within the program itself.
2. **More flexible input:** Currently the program is quite strict in terms of the expected input. As discussed in the Input section, the program expects a csv with only integers and no punctuation other than the ; separating the columns. It would be more convenient if the program were to cater for a more variety of inputs.
3. **Module weighting adjustment:** Allow the user to adjust the weighting of a module toward the final degree mark. This could be useful in the case where a user would like to calculate their HEMIS score, or require a weighted average.
4. **Ranking:** Rank all the users from top performing to least performing in the degree/module

6 Summary

This program does what is specified in the Overview.