ASSIGNMENT 3 – PART 1

CSIT110– Programming Fundamental Using Python

Session 1: January to March 2024

INSTRUCTIONS TO CANDIDATES

1. The assignment consists of two parts. This is part 1 of the assignment.
2. Part 2 is Moodle quiz. Should be done in class.
3. The name of the program must be **YourName_ClassListNo_A3.**py (Only one Python file)
4. **Total mark of Assignment 1 is 6 marks; 4 marks for Part II.**

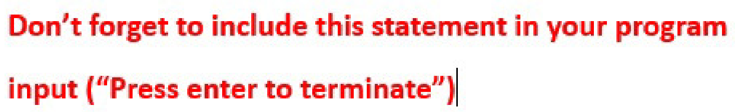
Your program, should begin with

// Full Name:
// Tutorial Group
// Declaration: …… tell me if it is your own work …. and whether you have
// passed your program to your friends.

Objective

The following features should be explored in your design:

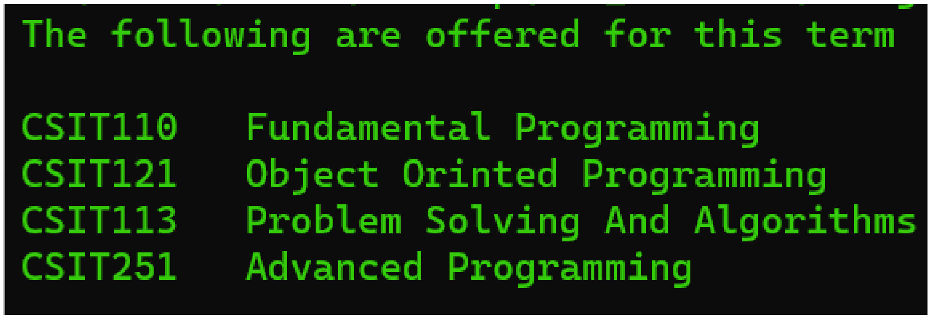
- Classes and objects
- List
- Dictionary / File processing (optional)



Don't forget to include this statement in your program

input ("Press enter to terminate")

Every term, SIM offered some subjects for students to enrol, for example



```
The following are offered for this term

CSIT110   Fundamental Programming
CSIT121   Object Orinted Programming
CSIT113   Problem Solving And Algorithms
CSIT251   Advanced Programming
```

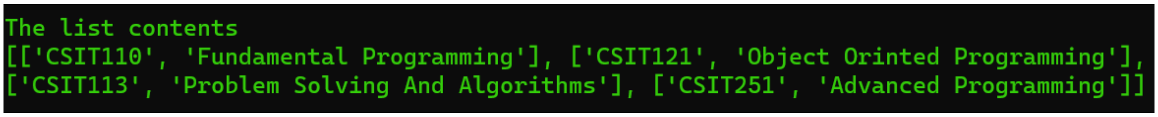
You can see each subject has a code and a title. Where to get this information and how to store them before processing? A few questions may arise:

- Read from a data file ... but not yet covered in syllabus
- User input the information ... very tedious, time consuming
- Hard coding .... if we can make it flexible to add in new subject, not a bad idea for testing purpose

- To store them, ideally, Python Dictionary is a good choice ... but we have not yet covered that topic
- Using Python List .... that is a good choice, flexible to add in some related information in the list (you can see later this advantage)

We adopt the idea of hard coding. We define three lists!!!

- A list to store the subject codes
- A list to store the corresponding subject codes
- A list (nested) to store the subject information



```
The list contents
[['CSIT110', 'Fundamental Programming'], ['CSIT121', 'Object Orinted Programming'],
['CSIT113', 'Problem Solving And Algorithms'], ['CSIT251', 'Advanced Programming']]
```

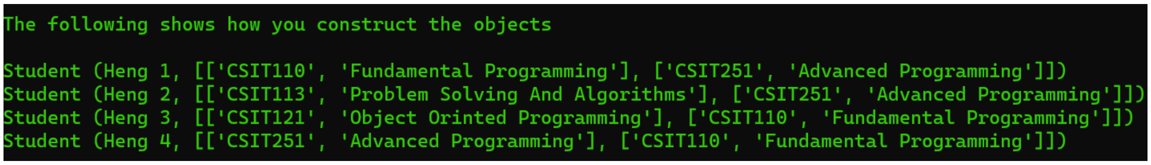
In the above screen shot, you see a list of lists; each nested list (four of them) describes the subject information.

Now we can talk about the objective of this assignment.

You design a class to describe a student. Each student has a name and a list consisting of two distinct subjects (two nested lists). To begin with, the Student class has at least the following three important functions:

- __init__ function: to initialize the name and a list of subjects enrolled by students
- __str__ function: to get a String that represents a student object
- __repr__ function: indicating how an object is created
- A display list function which displays a student object before the updates.
- Some more functions related to our objective

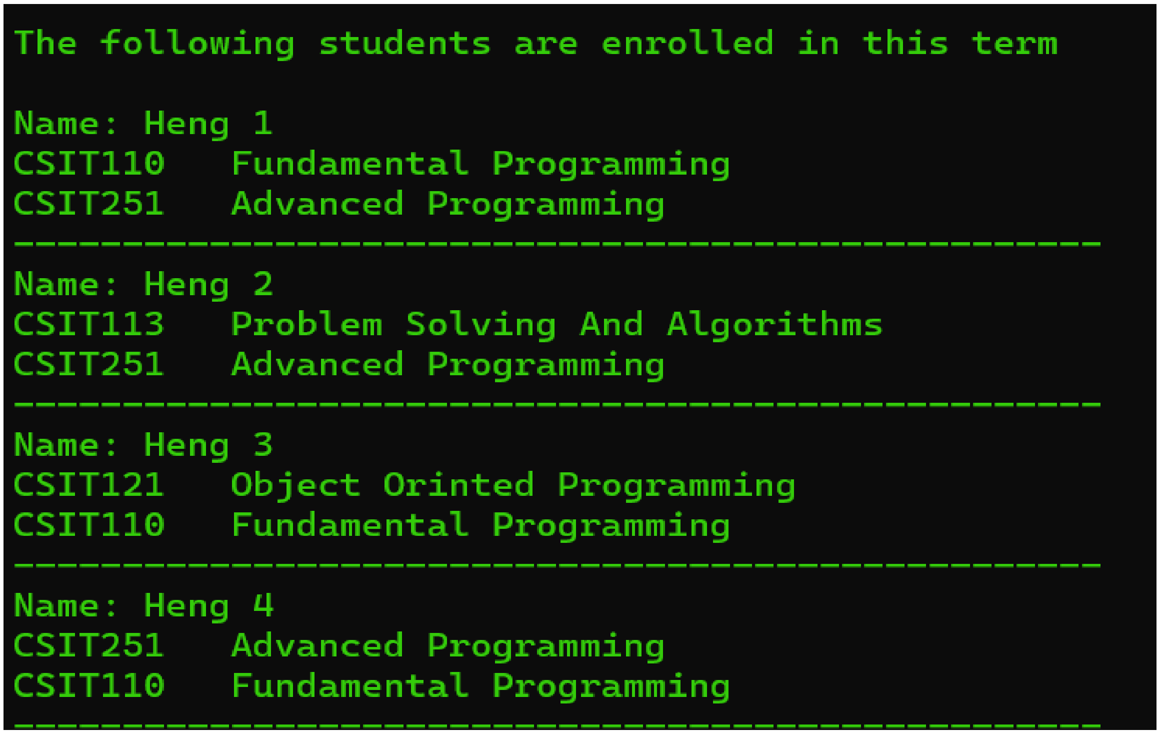
Now students can enrol to the subjects. We construct a list of Student objects. You can assume that every student takes two random *distinct* subjects. You can see from the following display how the 4 objects are created: (the use of repr function)



```
The following shows how you construct the objects

Student (Heng 1, [['CSIT110', 'Fundamental Programming'], ['CSIT251', 'Advanced Programming']])
Student (Heng 2, [['CSIT113', 'Problem Solving And Algorithms'], ['CSIT251', 'Advanced Programming']])
Student (Heng 3, [['CSIT121', 'Object Orinted Programming'], ['CSIT110', 'Fundamental Programming']])
Student (Heng 4, [['CSIT251', 'Advanced Programming'], ['CSIT110', 'Fundamental Programming']])
```

We now display the list of student objects: (the use of display list mentioned above)

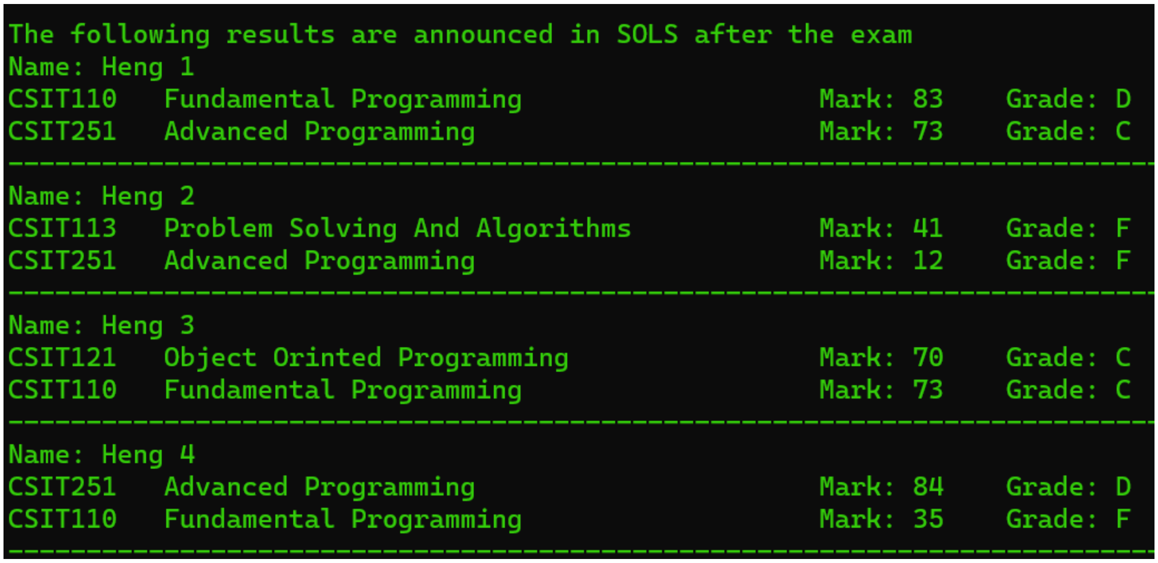


```
The following students are enrolled in this term

Name: Heng 1
CSIT110   Fundamental Programming
CSIT251   Advanced Programming
--------------------------------------------------
Name: Heng 2
CSIT113   Problem Solving And Algorithms
CSIT251   Advanced Programming
--------------------------------------------------
Name: Heng 3
CSIT121   Object Orinted Programming
CSIT110   Fundamental Programming
--------------------------------------------------
Name: Heng 4
CSIT251   Advanced Programming
CSIT110   Fundamental Programming
--------------------------------------------------
```

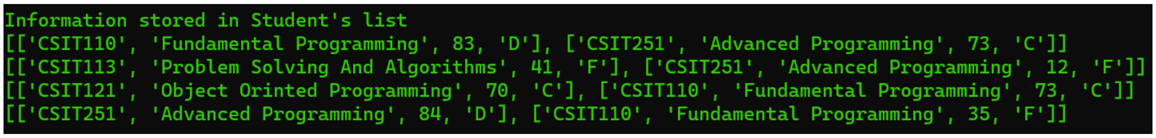
After the exam, your lecturer processes the final marks and uploads the final marks to SOLS system. The system computes the final grade. The following shows the results in your SOLS system 😊 ... (I guess, I never had a chance to see your SOLS platform)

Note here you use the __str__ function to display the Student objects



```
The following results are announced in SOLS after the exam
Name: Heng 1
CSIT110   Fundamental Programming                   Mark: 83    Grade: D
CSIT251   Advanced Programming                      Mark: 73    Grade: C
--------------------------------------------------------------------------
Name: Heng 2
CSIT113   Problem Solving And Algorithms            Mark: 41    Grade: F
CSIT251   Advanced Programming                      Mark: 12    Grade: F
--------------------------------------------------------------------------
Name: Heng 3
CSIT121   Object Orinted Programming                Mark: 70    Grade: C
CSIT110   Fundamental Programming                   Mark: 73    Grade: C
--------------------------------------------------------------------------
Name: Heng 4
CSIT251   Advanced Programming                      Mark: 84    Grade: D
CSIT110   Fundamental Programming                   Mark: 35    Grade: F
--------------------------------------------------------------------------
```

Very important in your design, the exam marks and the final grades should be appended to the lists:



```
Information stored in Student's list
[['CSIT110', 'Fundamental Programming', 83, 'D'], ['CSIT251', 'Advanced Programming', 73, 'C']]
[['CSIT113', 'Problem Solving And Algorithms', 41, 'F'], ['CSIT251', 'Advanced Programming', 12, 'F']]
[['CSIT121', 'Object Orinted Programming', 70, 'C'], ['CSIT110', 'Fundamental Programming', 73, 'C']]
[['CSIT251', 'Advanced Programming', 84, 'D'], ['CSIT110', 'Fundamental Programming', 35, 'F']]
```

When executing your program, all the screen shots listed in this assignment should be displayed one after another.

IMPORTANT

The name of your program must be **YourName_ClassListNo_A3.py** and make sure that this file can be executed. Upload **ONLY** the Python file to Moodle**. ALL ZIP FILE SUBMISSION WILL BE REJECTED**

No re-submission will be allowed after grading.

In the above file, remember to put down your name and the following declaration (some similar contents):

# Tell me if it is your own work, and whether you have passed your

# program to your friends etc etc etc

# and willing to accept whatever penalty given to you.

- **Wrong file name -0.2 mark**
- **No declaration, no name etc -0.3 mark**
- **No demo -0.5**