# Sheet 5

## Question 1:

1- Create **Computation class** with the following instance methods:

- √ **factorial()** which allows to calculate the factorial of an integer.
- √ **sumN()** allowing to calculate the sum of the first n integers **1 + 2 + 3 + .. + n.**

2- Then, create an object from Computation class to invoke all its instance methods.

**Example**:

```
Comput= Computation ()
print(Comput.factorial (5))
print(Comput.sum (5))
```

```
120
15
```

## Question 2:

1. Define a **Book** class with the following attributes: **Title, Author (Full name), Price**.

2. Define a **constructor** used to initialize the attributes of the method with values entered by the user.

3. Set the View() method to display information for the current book.

4. Write a program to testing the Book class.

**Example**

```
MyBook = Book("ntroduction to Python" , "Gowrichankar S. " , "23 $")
print( MyBook.view())
```

```
('Book Title: ', 'ntroduction to Python', 'Book Author: ', 'Gowrichankar S. ', 'Book Price: ', '23 $')
```

## Question 3:

1. Create a Python class **Person** with attributes: **name** and **age** of type string.
2. Create a **display()** method that displays the name and age of an object created via the Person class.
3. Create a **child class Student** which **inherits** from the Person class and which also has a **section** attribute.
4. Create a **method displayStudent()** that displays the name, age and section of an object created via the Student class.
5. Create a **student object** via an instantiation on the Student class and then test the displayStudent method.

**Example**:

```
P = Person("Mohammed", 37)
P.display()
print("------------------------------")
S = Student("Ahmed", 23 , "Mathematics")
S.displayStudent()
```

```
Person name :    Mohammed
Person age =    37
------------------------------
Student name :    Ahmed
Student age =    23
Student section =    Mathematics
```

## Question 4:

1. Write a **Rectangle class** in Python language, allowing you to build a rectangle with **length** and **width** attributes.
2. Create a **Perimeter()** method to calculate the perimeter of the rectangle and a **Area()** method to calculate the area of the rectangle.
   **Perimeter=2*(length+width)  area=length*width**
3. Create a method **display()** that display the length, width, perimeter and area of an object created using an instantiation on rectangle class.
4. Create a **Parallelepipede** child class **inheriting** from the **Rectangle class** and with a **height** attribute and another **Volume()** method to calculate the volume of the **Parallelepiped. Note: volume=length*width*heigth**

**Example**:

```
myRectangle = Rectangle(7 , 5)
myRectangle.display()
print("--------------------------------")
myParallelepipede = Parallelepipede(7 , 5 , 2)
print("the volume of myParallelepipede is: " , myParallelepipede.volume())
```

```
The length of rectangle is:    7
The width of rectangle is:    5
The perimeter of rectangle is:    24
The area of rectangle is:    35
--------------------------------
the volume of myParallelepipede is:    70
```

# Question 5:

1- Write a **Document** class with **document name** and **document content** attributes.

- create a *class method* **create_document()** that creates a new document object

- create an i*nstance method* **append_to_document()** that append additional text to the document content

- create an *instance method* **read_doc()** that retruns the document contents as a string

- create an *instance method* **is_txt_exist()** method that takes a text as its parameters to search whether it exist in the document or not.

- create an *instance method* **find_txt()** that takes a text as its parameter and return the location(line #, index in the line) of this text in the document if exist and returns -1 otherwise

**Example**:

```
d=Document('d1.txt', 'hello in python course.\n')
d.append_to_document('This is OOP assignmnet')
print(d.read_doc())
print('---------------')
print(d.find_text('OOP'))
```

hello in python course.
This is OOP assignmnet
---
OOP is exit in line #2 starting from index 8

**Question 6:** Trace the following program

```
class Person:

    def __init__(self, myname, myage, *myaddress):
        self.set_name(myname)
        self.set_age(myage)
        self.set_address(*myaddress)

    def get_name(self):
        return self.__name

    def get_age(self):
        return self.__age

    def get_address(self):
        return self.__address

    def set_name(self, name):
        self.__name=name


    def set_age(self, age):
        self.__age=age

    def set_address(self, *myaddress):
        self.__address=""
        for address in myaddress:
           address+=", "
        self.__address= self.__address + address


p1=Person("ahmed", 23, "Giza")
print(p1.get_address())
p2=Person("mohammed", 24, "Giza", "Ahmed Zewail st.")
print(p2.get_address())
p3=Person("mahmoud", 25, "Giza", "Ahmed Zewail st.", "Cairo Uni.")
print(p3.get_address())
```