QUESTION STYLES

Faculty of Applied Science
Department of: Cybersecurity
Course: Object-Oriented Programming

Theoretical

Q. Choose the correct answer.

1. What is the output of this code? → print(list(range(10,2,-4)))

A. [10, 6, 2]
B. [10, 6]
C. [6, 10]
D. [2, 6, 10]

2. ……. is the practice of breaking up a program into small parts.

A. Functionality
B. Class
C. Modularity
D. Loop

Q. Write down the output of the block of code below.

```
for i in range(1, 5):
    if (i%2 = = 0):
        continue
    print(i**2 , end = ",")
```

Q. Write down the output of the block of code below.

```
def myFunction(a):
    b = 10
    while (b > 5):
        b = b – a
    return a*b
print("Result is" , myFunction(3))
```

Q. Write down the output of the block of code below.

```
course = "Cryptography"
for char in course:
    if (char == 'y') or (char == 'p'):
        continue
    print(char , end = '**')
```

Practical

Q. Write Python code to do the following.

A. First, create the following class.
B. Then create one object for an employee.
C. Call *annualSalary( )* for that employee object. Then print name and annual salary of it.

Employee Class
Attributes
empID
empName
monthlySalary
Methods
getIDandName( )
setMonthlySalary (***newSalary***)

- ***getIDandName( )*** returns student's ID and name.
- ***setMonthlySalary(newSalary)*** sets monthly salary to a new amount.

Q. Write Python code to do the following.

A. Create the **Book** class for books in the 'Science' category.

B. Then create one object for a book.

Book Class	
Class Attribute	
category = 'Science'	
Instance Attributes	
Code	→ Private attribute
AuthorName	→ Private attribute
PageNumbers	→ Public attribute
Method	
bookVolume( )	

- ***bookVolume( )*** returns 'thick book' if the book's number of pages is more than 100, otherwise it returns 'thin book'.

QUESTION STYLES

Faculty of Applied Science
Department of: Cybersecurity
Course: Object-Oriented Programming

Q. Write Python code to do the following.

A. Create three classes: **Teacher, Full-time** and **Part-time**. Consider class inheritance, so that both **Full-time** and **Part-time** are subclasses of **Teachers**.

B. All teachers have **ID**, **name**, and **department** attributes.

C. Consider class polymorphism by defining a method named **calculate_salary( )** in all classes to calculate and return salary.

a. Salary of full-time teachers is calculated as base_salary + extra_payment, while extra_payment is calculated as extra_hours*pay_per_hour.

b. Salary of part-time teachers is calculated as hours_worked*pay_per_hour.

D. Create one object of each Full-time and Part-time teacher.

E. Print the salary of both teachers, if:

Full-Time Teacher → Base Salary = 2000 Extra Hours = 10 Pay Per Hour = 50

Part-Time Teacher → Hourse Worked = 8 Pay per Hour = 100