University of Asia Pacific

Department of Computer Science & Engineering

Mid-Semester Examination Fall -2020

Program: B. Sc Engineering (2020 Year/Fall Semester)

Course Title: Object Oriented Programming II: Visual and Web Programming

Course No: CSE 309 Credit: 3.00

Time: 1.00 Hours. Full Mark: 60

There are **Four** Questions. Question no 1, 2 are compulsory. Answer any of 3 or 4. All questions are of equal value. Figures in the right margin indicate marks.

Incase of writing python codes, be careful about the indentations.

1. a) Suppose you are in the magical world of J. K. Rowling's Harry Potter series. Here, a person can be distinguished as a Muggle or a Wizard. Wizard is a person who has magical abilities and has magical blood in them. On the other hand, Muggle is someone who lacks any sort of magical ability and was not born into a magical family. Muggles can also be described as people who do not have any magical blood inside them. There is another kind who are known as half-bloods. That means one of their parents is a Wizard and the other one is a Muggle. So they have both magical and muggle blood in them. We all know about Professor Severus Snape who was also known as the half-blood prince. **[8]**

Now, from the above passage find out and show the following inheritance relations using diagrams (only class names)–

i. Hierarchical Inheritance
ii. Multilevel Inheritance
iii. Multiple Inheritance
iv. Hybrid Inheritance

No class diagram or code is needed.

b) In the animal kingdom, we can divide the animals into two groups - carnivorous and herbivorous. A carnivore, meaning "meat eater", is an animal whose food and energy requirements are derived solely from animal tissue or meat, whether through hunting or scavenging. A herbivore is an animal anatomically and physiologically adapted to eating plant material, for example, foliage or marine algae, leaves etc. for the main component of its diet. **[12]**

Now, create an abstract class ***Animal***, which will have an abstract method named ***set_food_habit()***

Another two classes, ***Carnivorous*** and ***Herbivorous*** will inherit the ***Animal*** class and implement the abstract method.

You may maintain a list type attribute called "***food_list***" where you will add the food items while implementing the abstract method.

2. We all are missing the **Ekushey Boi Mela** in this month of February. So in remembrance of that, you want to create a personal profile and a page about your favourite book with Django. The **url** of the personal profile page should be ***home/<your full name>*** and the **url** of your favourite book page should be ***home/<your favourite book's name>.*** Personal profile page should contain your personal information and the favourite book page should contain four or five lines about your favourite book. **[20]**

Now do necessary changes in the following files and explain the code flow:

i. personal_profile.html
ii. favourite_book.html
iii. urls.py
iv. views.py

NB #1: No need to add default imports
NB #2: Incase of the html files, show only the desired output of the file.

3. a) Suppose your friend, Alice is trying to find all the odd numbers that are between ***X*** and ***Y***. You want to help your friend by writing her a ***for*** loop. **[10]**
Here, ***X = your ID*** , ***Y = your ID + 1000***

Write a ***for*** loop to help Alice to print the odd numbers that are between ***X*** and ***Y***.

b) Write a function that will take an integer number as a parameter and return true if the number is divisible by both ***3*** and ***5***. Otherwise the function will return false. Give an appropriate name to the function. **[10]**

or,

4. Write a function that will take two lists as parameters. Then it will compute two summations. **[20]**

sum1:= summation of elements that are in the first list and in the indexes divisible by three. (The index starts from 0)

sum2:= summation of elements that are in the second list and in the indexes divisible by five. (The index starts from 0)

Then it will compare between these two summations and return the larger sum.

Sample Input: [1,5,3,2,8,9,5,6,3,10], [2,6,4,8,10,11,7,3,9,5,6,1]
Sample output: 19