W255/711

DUBLIN INSTITUTE OF TECHNOLOGY

DT255 BSc. (Honours) Degree in Information Systems / Information Technology (Full-time)

Stage 2

WINTER EXAMINATIONS 2016-2017

OBJECT-ORIENTED PROGRAMMING [CMPU2029]

DAMIAN GORDON
DR. DEIRDRE LILLIS
MR. JEREMY BRUCE-SMITH

TUESDAY 8TH JAN 2017 4.00P.M.-6.00 P.M.

2 HOURS

INSTRUCTIONS TO CANDIDATES

ANSWER **THREE** QUESTIONS OUT OF **FOUR**.

ALL QUESTIONS CARRY EQUAL MARKS.

1. (a) What is *inheritance*? Provide an example. (5 marks)

(b) What is *polymorphism*? Provide an example. (5 marks)

(c) Develop a simple address book system that keeps track of names and e-mail addresses. Create a class called `Contact` that keeps the list of all contacts, and initialises the names and addresses for new contacts:

- So we need an attribute `contacts_list`, and
- we need a method `_ _init_ _(self, name, email)`

(8 marks)

(d) Develop a new class called `Supplier` that inherits from `Contact` and also has the following:

- a new method called `order(self, order)` that takes in a string called `order` (e.g. "100 bananas") prints out the `order` string and the `name` of the current object.

(8 marks)

(e) Based on what you have done in Parts (c) and (d), show how you would declare two instances each of `Contact` and `Supplier`, and initialise them and print out each ones `name` and `email` addresses. Explain what would happen if you tried to order from your `Supplier` instances, and also explain what would happen if you tried to order from your `Contact` instances

(7 marks)

2, (a) What is *overriding*?

(3 marks)

(b) Assume you have a class called `Customer` that has an `Order` method (that just prints out a message saying "`The order is complete`"), create a new class `CustomerCheck` with a new `Order` method that takes a parameter `balance`, and the method checks if the `balance` is less than zero, and if so it prints out a message saying "`The customer is in debt`", else it prints the standard message of "`The order is complete`".

(12 marks)

(c) Show how you would create two instances of `Customer` and two instances of `CustomerCheck`, and demonstrate how you would call the `Order` method in both cases.

(8 marks)

(d) What is *super*?

(3 marks)

(e) Rewrite the `CustomerCheck` class to use a super.

(7 marks)

3. (a) What is *Multiple Inheritance*?

(3 marks)

(b) Assuming you have two classes, one called `Super1` and one called `Super2`, show the Python syntax to create a class `MySubClass` that inherits from both of those classes.

(6 marks)

(c) What is a *Mixin* class?

(3 marks)

(d) You want to develop a system that will play a range of audiofiles. So you need a method called `play()` to play an audio file. The instruction to play the file is simple as:

```
>>>  audio_file.play( )
```

However, different audio files use different compression algorithms (e.g. .mp3, .wma, .ogg), and some aren't stored as compressed at all (e.g. .wav). You need to develop a system whereby each filetype is represented as a different subclass of AudioFile, and each of those has a `play( )` method.

Write a Python program to address the above scenario:

(i) So create a Mixin class called `AudioFile` that has the following:
- an `_ _init_ _()` method that takes in a filename and checks for a known filetype.

(12 marks)

(ii) And create three classes the inherit from `Audiofile` (for filetypes WAV, MP3, and OGG) that do the following:
- sets a variable `EXT` to the file type
- Has a `play()` method that prints "`playing` **`<FILENAME>`** `as` **`<FILETYPE>`**"

(9 marks)

4. (a) Given the following procedural Python:

```
import math

def distance(p1, p2):
    return math.sqrt((p1[0] – p2[0])**2
                     + (p1[1] – p2[1])**2)
# END distance

def perimeter(polygon):
    perimeter = 0
    points = polygon + [polygon[0]]
    for i in range(len(polygon)):
      perimeter += distance(points[i], points[i+1])
    # ENDFOR
    return perimeter
# END perimeter
```

(i) Explain what the following statements will do:

```
>>> square = [(1,1),(1,2),(2,2),(2,1)]
>>> perimeter(square)
```

(8 marks)

(ii) Discuss with examples how you would convert this code into Object-Oriented Python code.

(12 marks)

(iii) Show how you would create an instance of the square and calculate its perimeter as defined in the Object-Oriented version from part (ii).

(8 marks)

(b) What does the `property` function in Python do?

(5 marks)