UNIVERSITY OF LONDON

GOLDSMITHS COLLEGE

Department of Computing

B. Sc. Examination 2017

IS50001C Foundations of Programming

Duration: 2 hours 15 minutes

Date and time:

This paper is in two parts: part A and part B. You should answer ALL questions from part A and TWO questions from part B. Part A carries 40 marks, and each question from part B carries 30 marks. The marks for each part of a question are indicated at the end of the part in [.] brackets.

There are 100 marks available on this paper.

You are not allowed to use any mobile device (such as telephones, laptops, calculators, tablets) during the exam.

THIS PAPER MUST NOT BE REMOVED
FROM THE EXAMINATION ROOM

Part A

You should attempt all of these questions

Each question has one correct answer

(a) What is the result of executing the following instructions:

```
a = 0
b = 2
print('(a/b) is', float(a/b))
```

i. A syntax error
ii. A run-time error
iii. The local printer will print the message "`(a/b) is 0.0`"
iv. The string `'(a/b) is 0.0'` will be displayed on the screen
v. None of the above

[4]

(b) What is the final value of the variable `a` after executing the following assignments:

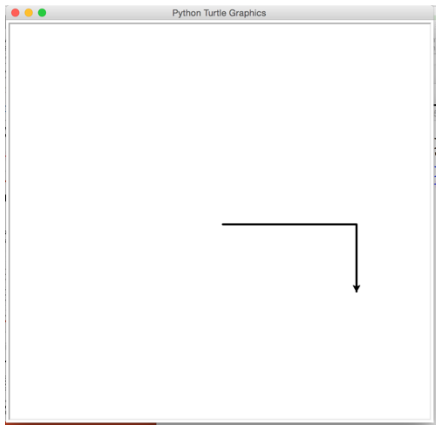
```
a = 0
b = 2
b = a-b
a = b
b = 0
```

i. 0
ii. 2
iii. -2
iv. None of the above

[4]

(c) In which order should the following statements be executed for the turtle to draw the figure plotted in the graphic window shown on the right:

```
1 jo = turtle.Turtle()
2 jo.forward(100)
3 import turtle
4 wn.exitonclick()
5 jo.forward(200)
6 wn = turtle.Screen()
7 jo.right(90)
```



i. 3, 6, 1, 5, 7, 2, 4
ii. 6, 3, 1, 2, 7, 5, 4
iii. 3, 6, 1, 7, 5, 2, 4
iv. 3, 6, 4, 1, 5, 7, 2
v. None of the above

[4]

(d) The correct Python command to generate a random integer between 1 and 10 (inclusive) is:

i. `random.randrange(0,11)`
ii. `random.randrange(0,10)`
iii. `random.randrange(1,11)`
iv. `random.randrange(1,10)`
v. None of the above

[4]

(e) What is produced in output by the following code snippet:

```
x = 0
print(x - 2 <-1 and 2 + 3 == 5)
```

i. `False`
ii. `True`
iii. `'x - 2 <-1 and 2 + 3 == 5'`
iv. An error message
v. None of the above

[4]

(f) Executing the following statements will generate a (run-time or syntax) error:

```
x = -10
if x < 0:
    print("The number", x, "is invalid.")
else:
    if x > 0:
        print("The number", x, "is positive")
    else:
        print("The number", x, "is =0")
```

i. True.
ii. False.

[4]

(g) Which sequence will the following code extract produce in output:

```
n = -2
for i in range(4,0,n):
    print(i)
```

i. 4, 2, 0
ii. 2, 0, -2
iii. 0, 2, 4
iv. 2, 0
v. 4, 2
vi. 2
vii. None of the above.

[4]

(h) What value will be printed on the screen as a result of executing the following code:

```
def subtractFrom(x,y):
    y = y-x

result = subtractFrom(0,1)
print(result)
```

i. 0
ii. 1
iii. -1
iv. `None` (no value)
v. An error message

[4]

(i) What is printed on the screen by the following code fragment:

```
a = ['a', 1] + [True] + [[1,2,3], False]
print(a[2])
```

i. `[1,2,3]`
ii. `'a'`
iii. `True`
iv. `'a[2]'`
v. None of the above

[4]

(j) What is the output of the following program:

```
def mystery(x):
    z = ''
    y = len(x)
    while y > 0:
        z = z + x[y-1]
        y = y-1
    return z

print(mystery('mystery'))
```

i. `None` (no output is produced)
ii. `''` (the empty string)
iii. `'yretsym'`
iv. `"mystery('mystery')"`
v. `'mystery'`

[4]

Part B

You should attempt two of these three questions

Question B1

(a) i. What is the output of the following code fragment? Complement your answer with a short explanation.

```
for x in [1,3,5]:
     print(x)
     x = 2
```

[6]

ii. What is the output of the following code fragment? Explain.

```
for y in range(5,-1,-2):
     y = 1
     print(y)
```

[6]

(b) What is the output of the following program? Provide a brief explanation in support of your answer.

```
def setup(x,y):
    x = 0
    y = 0

def main():
    setup(a,b)
    print(a,b)

a = 2
b = 3
main()
```

[6]

(c) Write a function called `multiplyOdds` that takes an integer *n* as a parameter and returns the product of the first *n* odd counting numbers. (You can assume the number *n* is > 0)

For example, if $n = 4$, the value returned should be the result of $1 \times 3 \times 5 \times 7$, i.e., 105.

[12]

Question B2

(a) i. What does the execution of the following program produce on the screen? Complement your answer with a brief explanation.

```
res = 0
for x in range(5):
       for y in range(5):
             res = res + 1
print(res)
```

[6]

ii. What is the output of the following code snippet? Explain.

```
for i in range(5):
    res = 0
    for j in range(5):
         res = res + 1
print(res)
```

[6]

(b) Assuming `a` is a list of integers, explain why the following code fragment generates an error:

```
for i in range(-1, len(a)):
    a[i+1] = 1 + a[i]
```

[6]

(c) Write a function called `listExpand` that accepts a list as a parameter and returns a new list obtained by replacing each element the original one with *i* copies of that element, where *i* is the position of that element in the list (i.e., the first element in the list occupies the first position, the second element the second position, and so on).

For example: given the list [2, 'a', 7.1] as input, the function should return [2, 'a', 'a', 7.1, 7.1, 7.1]

[12]

Question B3

(a) i. Consider the following code:

```
import turtle
jose = turtle.Turtle()
nick = turtle.Turtle()
jose.color('pink')
jose = nick
nick.color('blue')
jose.forward(100)
```

What color is the line drawn by the turtle '`nick`' on the graphic window? Justify your answer with an explanation.

[6]

ii. Explain what is wrong with the following program:

```
def exponent(x,y):
    z = x**y
    return z

exponent(2,3)
print(z)
```

[6]

(b) Explain what the function "`drawIt`" defined below does, and what the main program following the definition uses it for (your explanation should include a drawing showing what the program produces in the graphic window):

```
import turtle
def drawIt(t, l, a, c):
    for i in range(len(c)):
          t.color(c[i])
          t.forward(l)
          t.left(a)

w = turtle.Screen()
j = turtle.Turtle()
drawIt(j, 100, 120, ['red', 'blue', 'green'])
drawIt(j, -100, -120, ['red', 'blue', 'green'])
```

[8]

(c) Without using the method `replace()`, write a function called `noDigits` that takes a string as a parameter and returns a new string obtained by stripping the original one of all the digits.

For example, given the string "H9e2llo Wo7r3l4d", the function should return "Hello World".

[10]

End of exam