UNIVERSITY OF LONDON

GOLDSMITHS COLLEGE

Department of Computing

B. Sc. Examination 2018

IS51021B Problem Solving for Computer Science

Duration: 2 hours 15 minutes

Date and time:

This paper is in two parts: part A and part B. Part A carries 40 marks, and each question from part B carries 30 marks. ***You should answer ALL questions from part A and TWO questions from part B.*** *If you attempt all 3 questions in part B, only the* ***first two*** *that you attempt will be marked. 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 electronic device (such as mobile 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 multiple-choice question has one (and only one) correct answer.
For each question, write your choice on your *answer* book.

(A) What does execution of the following lines of Python code generate on the screen?

```
a = 1
b = 1
print('a-b =', str(a-b))
```

i. A run-time error.
ii. `a-b = 0`
iii. `a-b = 1`
iv. Nothing on the screen; the local printer will print "`a-b = str(a-b)`"

[2]

(B) What is the final value of variable `y` after executing the following code?

```
y = int("1")
x = y - 1
y = x
```

i. 0
ii. 1
iii. -1
iv. None of the above.

[2]

(C) What is produced on the screen when the following 2 lines of code are executed?

```
s = "15"
print(len(s))
```

i. An error message.
ii. 1
iii. 2
iv. 15

[2]

(D) What will be printed on the screen when the following Python code is executed?

```
x = "2"
print(float(x))
```

i. An error message
ii. `2.0`
iii. `2`
iv. The string "`float(x)`"

[2]

(E) What is the value of variable `a` after executing the following code?

```
(a,b) = (1,0)
a = a-b
```

i. 0
ii. -1
iii. 1
iv. None of the above

[2]

(F) Which message will the statements below produce on the screen?

```
var1 = -3
var2 = 0
if var1 <= var2:
   print("The value", var1, 'is negative.')
else:
   print("The value", var1, 'is positive.')
```

i. `The value 0 is negative.`
ii. `The value -3 is negative.`
iii. `The value -3 is positive.`
iv. An error message.

[2]

(G) What type will variable `n` be after the following line is executed?

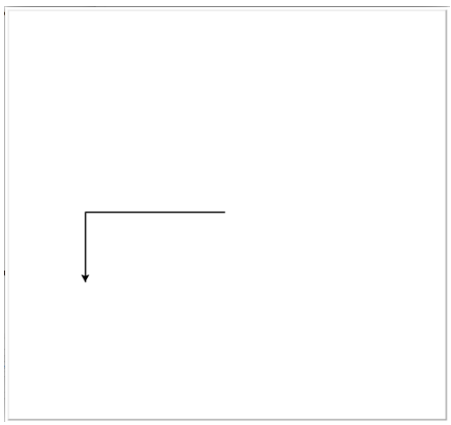
```
n = input("Please enter your choice:")
```

i. String
ii. Float
iii. Integer
iv. Boolean

[2]

(H) In which order should these Python statements be executed for the turtle to draw the figure plotted in the graphic window (canvas) shown on the right?

```
1  jo = turtle.Turtle()
2  jo.forward(100)
3  import turtle
4  wn = turtle.Screen()
5  jo.forward(200)
6  jo.left(90)
7  jo.right(180)
```



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

[2]

(I) What does the following snippet of code print on the screen?

```
import random
print(random.randrange(1,100,2))
```

i. A random even number between 1 and 100
ii. A random odd number between 1 and 100
iii. A random float between 1 and 2
iv. None of the above

[2]

(J) What do the following lines of Python code produce as output?

```
n = 1
print(n > 1 or 4%2 == 0)
```

i. An error message
ii. The string `"n > 1 or 4%2 == 0"`
iii. `False`
iv. `True`

[2]

(K) How many times does the following ***for*** loop print "Hello" on the screen?

```
for i in [1,4,2]:
    print("Hello")
```

i. 1 time
ii. 2 times
iii. 3 times
iv. None of the above

[4]

(L) What does execution of the following ***for*** loop produce on the screen?

```
n = 2
for i in range(n**n, n, -1):
    print(i)
```

i. An error message
ii. The numbers 4, 3 (printed on separate lines)
iii. The numbers 4, 3, 2 (printed on separate lines)
iv. The numbers 3, 2, 1, 0 (printed on separate lines)

[4]

(M) What does the following Python program produce as output?

```
def myChoice(a,b,c):
    if a > 0:
        return a-b
    elif a < 0:
        return a-c
    else:
        return c-b

a = 1
print(myChoice(0,1,2))
```

i. 1
ii. -1
iii. -2
iv. None of the above

[4]

(N) What is the output of the following code extract?

```
s = "Hi"
s = s + s[0:1]
print(s)
```

i. An error message
ii. `Hi`
iii. `HiH`
iv. `HiHi`

[4]

(O) What does the following code snippet produce?

```
a = [1, 2, 3]
a = a + [4, 5, 6]
print(a[0])
```

i. 0
ii. 1
iii. 4
iv. 5

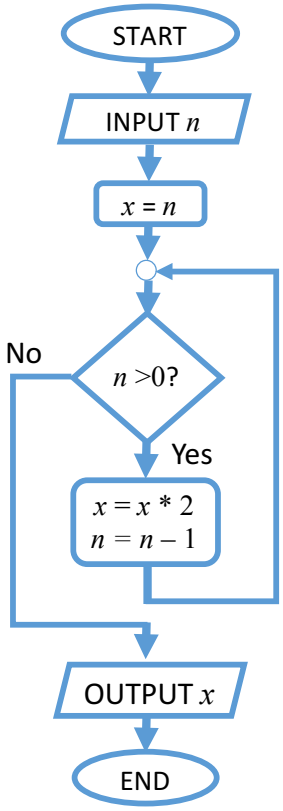
[4]

Part B

You should attempt two of these three questions

QUESTION B1

(a) Consider the algorithm described by the flowchart below, with n integer and ≥ 0:



i. If the value of n entered by the user is 0, how many times will the loop body be repeated?

[2]

ii. If the value of n is an integer > 0, how many times is the loop body repeated?

[3]

iii. What happens to variables x and n in the loop body?

[3]

iv. In view of the above points, what value is printed at the end for any given value n provided by the user?

[4]

v. Write a Python program that implements this algorithm using a ***while*** loop (**Note**: your code should *not* contain any new function definitions).

[8]

(b) What is wrong with the following program? Provide a brief explanation, indicating what should be added to or changed in the code to remove any error(s).

```
1   def multiply(x,y):
2         z = y*x
3
4   result = multiply(2,3)
5   print(z)
```

[10]

QUESTION B2

(a) Consider the following Python program, in which the lines have been numbered for convenience:

```
1  import turtle
2  john = turtle.Turtle()
3  wn = turtle.Screen()
4  john.color('red')
5  john.backward(100)
6  john.forward(100)
7  john.left(90)
```

i. Draw a diagram showing what the graphic window (canvas) looks like after the above commands are executed, labelling the colour of any lines drawn in it.

[2]

ii. Explain what happens to the internal representation of variables and objects that Python maintains when line 2 (`john = turtle.Turtle()`) is executed.

[4]

iii. Now suppose that the following lines of code are executed after the previous ones:

```
8  mary = turtle.Turtle()
9  mary.color('blue')
10 mary = john
11 mary.forward(100)
```

What will the canvas look like at the end? Draw a second diagram on your answer book, labelling the colour of any new lines drawn in it.

[3]

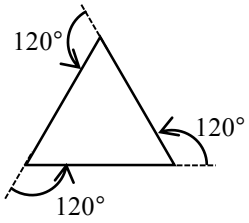
iv. Provide a brief explanation of your answer to point iii. above: what happens to the variables `john` and `mary` when the assignment "`mary = john`" (line 10) is executed? What happens to the turtle objects these two variables refer to?

[4]

(b)

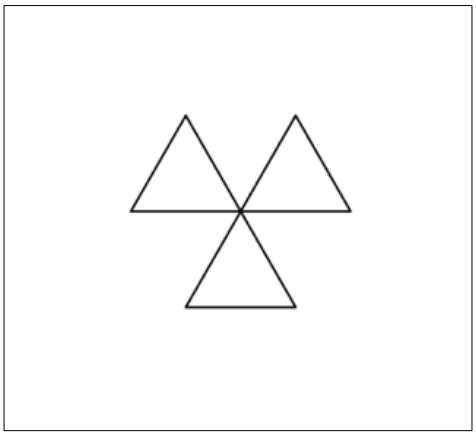
i. Write a function called `drawTriangle(t,sz)` that uses the turtle `t` to draw an equilateral triangle having `sz` pixel-long sides, using `t`'s current colour. Recall that the external angle of an equilateral triangle is 120° (as shown below).

[4]



ii. Write a second function, called `Koch(t,sz)`, that calls `drawTriangle(t,sz)` three times to draw 3 identical triangles (of size `sz`) arranged as shown in the figure below:

[5]



(c) We wish to define a new class in Python for representing a shopping list, i.e., a list of items to be purchased at the grocery. Assume the following class header is given:

```
class shoppingList:
```

i. Write the "`__init__(self)`" class constructor that you'll need, assuming that a newly created shopping list should contain no items.

[3]

ii. Enrich the class with a method called `addItem(name)` which enables adding a new item to the shopping list. For example, to create a list containing the two items `"apples"` and `"tomatoes"`, the user should be able to write:

```
myList = shoppingList()
myList.addItem("apples")
myList.addItem("tomatoes")
```

[5]

QUESTION B3

(a) The following Python function takes a list `x` of integers as input parameter:

```
def incognito(x):
    s = 0
    for i in x:
      if i == -1:
          s += 1
    return s
```

i. If the function is passed an empty list, i.e., `incognito([])`, how many times is the body of the ***for*** loop repeated?

[2]

ii. If list `x` contains *n* integers, how many times is the body of the ***for*** loop repeated?

[2]

iii. If the parameter `x` is the list `[2,1,0]`, what will the value of the loop variable `i` be during the first iteration of the ***for*** loop? And during the last one?

[2]

iv. What value is returned by the function if the parameter `x` is the list `[5,-1,-1]`? And if `x` is the list `[0,2,4,-5,6]`?

[2]

v. What value does the function return in general, for *any* given list `x` of integers?

[2]

(**b**) Write a Python function called `takeAway(x, y)` that takes two strings `x`, `y` as input parameters and returns the original string `x` after removing from it all characters that appear also in `y`.

> For example, `takeAway('Rubber', 'bar')` should return `'Rue'`: in fact, `'b'` and `'r'` appear also in `'bar'`, and when these are removed from `'Rubber'`, only `'Rue'` is left. (Note that, in this example, uppercase `'R'` should not be removed).

NOTE: your code may *not* use any *list operators* except for "`in`", "`not in`" *and* "`+`".

[8]

(**c**)

i. Explain how the *Selection Sort* algorithm works. Provide an example involving sorting a list of integers in ascending order.

[8]

ii. How many "passes", and how many item comparisons, in total, will this algorithm require to sort a list of *n* elements?

[4]

End of Exam