

**Birkbeck**  
**(University of London)**

**BSc/FD EXAMINATION**

**Department of Computer Science and Information Systems**

**Introduction to Programming (BUCI007H4)**

**CREDIT VALUE: 15 credits**

**Summer 2018**

**Date of examination: 22nd MAY 2018**

**Duration of paper: 14:30 – 16:30**

**The programming language is PYTHON.**

There are **ten** questions in this paper.

Answer all **ten** questions.

Each question carries **10** marks in total.

Calculators and other electronic devices are not permitted.

The examination is closed book.

No supplementary material is provided.

This paper is not prior disclosed.

The quote marks for strings are in this style "string".

1. (a) Which of the following are accepted by the compiler as names of variables and which are not accepted? (5 marks)

- i) ZZZZ
- ii) oscar\_6
- iii) \_dollar\$
- iv) 0Ooo
- v) O00oo

- (b) Consider the following code. (5 marks)

```
a = 2
b = 7
c = 3
d = 4
e = 4
a = a + a * a
b = 3 * b // 4
c = round(400 / (300 / (200 / (11 % c))), 2)
d = 10 ** d * 3 ** 2
e = round(34 / e + 6)
```

What are the values of a, b, c, d and e after the above code is executed?

2. (a) Explain the difference between compile-time error, run-time exception and run-time error. Give an example for each category. (6 marks)
- (b) A shop sells two kinds of cross-trainers Type A and B. The purchase price, marked price and discount for each type are listed below. (4 marks)

Type	Purchase price	Marked price	Discount
A	£120	£190	10%
B	£200	£310	20%

Create necessary variables and write Python code to compare the profit made by selling one of the cross-trainers of each type. The Python output should be one of the following:

- i) Selling one Type A cross-trainer makes more profit.
- ii) Selling one Type B cross-trainer makes more profit.
- iii) Both types make the same profit.

3. Find five errors in the following code. Include the line numbers to explain the errors. (10 marks)

```
1 originalPrice = input("Please enter the original price:")
2 if onSale:
3     discount = 0.9
4 sellPrice = trunc(originalPrice * discount, 0)
5 print("Original price (plus handling fee) is", originalPrice+1)
6 print("Sell price is" + sellPrice)
```

4. (a) Write Python code to read in from the keyboard the number of cans in a pack of cans and assign the value to the variable numCans. Set numCans to be 0 if the number entered is negative and include an error message. (2 mark)
- (b) What is the value of numCans in your code if the user enters the following in the shell? (3 marks)
- i) Please enter the number of cans in a pack: 5
  - ii) Please enter the number of cans in a pack: 7.0
  - iii) Please enter the number of cans in a pack: -4
- (c) What is printed when the following code is run? (5 marks)
- i) `print(float("3e1")+int(-7.6))`
  - ii) `print("smart"[2] * 3 + 5 * 'kids'[-2])`
  - iii) `print(len("\come on!\\"))`
  - iv) `print(float(str(-4*2)))`
  - v) `bbk = ["Birkbeck", "University", "Of", "London"]`  
`print(bbk[1][-2])`

5. (a) Consider the following statements. (6 marks)

```
price = 1.229
print("Price per litre: %5.2f" % price)
```

Identify the format specifier, the format string and the string format operator.

- (b) Consider the following statements.

```
percentage = 69.9763
print("A:", "%d" % percentage)
print("B:", "%.f" % percentage)
print("C:", "%s" % percentage)
print("D:", "%06.2f" % percentage)
```

Describe the print out when the above statements are executed. (4 marks)

6. (a) Evaluate the following expressions.

- i)  $4 > 7$
- ii)  $4 == 4$
- iii)  $2 < 5 > 6$
- iv)  $4 != 5$

(2 marks)

- (b) Define a Boolean expression that has the value **True** if at least one of the three variables  $x$ ,  $y$ ,  $z$  has a value 0. Otherwise the expression has the value **False**. An if statement is not accepted. (4 marks)
- (c) Write out the truth table for the Boolean expression  $A$  and  $\text{not}(B)$ . (4 marks)

7. (a) What is printed when the following code is executed, firstly with x equal to 4 and secondly with x equal to 5? Justify your answer.

```
if x == 4 :
    print("a")
else :
    if x == 5 :
        print("b")
        print("c")
```

(2 marks)

- (b) What is printed when the following code is executed, firstly with x equal to 4 and secondly with x equal to 5? Justify your answer.

```
if x == 4 :
    print("a")
else :
    if x == 5 :
        print("b")
print("c")
```

(2 marks)

- (c) Write out the code for an if statement that prints **True** if the variable x has a value of type float in the range 0 to 4 inclusive and that prints **False** if x has a value of type float strictly outside the range 0 to 4. (6 marks)

8. (a) What is printed when the following code is executed? Justify your answer.

```
i = 0
sum = 0
while (i <= 2) :
    sum = sum+i
    i = i+1
print(sum)
```

(4 marks)

- (b) Rewrite the code in part (a) of this question in order to replace the while loop by a for loop. (4 marks)
- (c) Describe one application in which a while loop is preferable to a for loop. (2 marks)

9. (a) Identify the function header and the function body in the following code.

```
def cubeVolume(sideLength) :  
    if (sideLength <= 0) :  
        return 0  
    volume = sideLength**3  
    return volume
```

(2 marks)

- (b) The function `cubeVolume` is as defined in part (a) of this question. What is printed when the following code is executed?

```
sideLength = 2  
print(cubeVolume(-1))  
sideLength = 3  
print(cubeVolume(sideLength))
```

(4 marks)

- (c) Rewrite the function `cubeVolume` in part (a) of this question to obtain a new function `cubeVolume2`. The new function has no argument or parameter. Instead, it requests a value of `sideLength` from the keyboard using the prompt

Please enter the side length:

and then returns the volume of the cube. Write out the code for `cubeVolume2` in full.

(4 marks)

10. (a) Consider the list `ls = [3, 1, 7, 2]`. Write out the full set of values of `i` such that the statement `print(ls[i])` is executed without error.

(4 marks)

- (b) The list `ls` is as defined in part (a) of this question. List the value or values of `i` such that `i` and `ls[i]` have the same value, i.e. the expression `i == ls[i]` has the value `True`.

(2 marks)

- (c) What is printed when the following code is executed?

```
ls = [3, 1, 7, 2]  
ls.insert(2,3)  
print(ls)
```

(4 marks)