

THE UNIVERSITY OF AUCKLAND

Semester One, 2016
Campus: City

TEST

COMPUTER SCIENCE

Principles of Programming

(Time Allowed: 75 Minutes)

- NOTE:**
- You must answer **all** questions in this test
 - Calculators are **NOT** permitted.
 - Answer all questions in the space provided in this booklet
 - There is space at the back for answers that overflow the allotted space.

Surname	
Forenames	
Student ID	
Username	
Lab Time	

Question	Mark	Out Of
1		20
2		20
3		20
4		20
5		20
Total		100

Question 1 (20 marks)

a) Complete the output of the following code?

```
var1 = 59
var2 = 71
var3 = var1
var1 = var2
var2 = var3
print("Var1:", var1, "\nVar2:", var2, sep=" ")
```

Var1:
Var2:

(3 marks)

b) What is the output of the following code?

```
result = 1 + 2 ** 3 / 3 // 2 % 5
print(result)
```

(3 marks)

c) What is the output of the following code?

```
print("", "Midterm", "\n", "Test", "", sep = "*" * 3)
```

(3 marks)

d) What is the output of the following code?

```
sentence = "That kazoo sounds beastly!"
substring = sentence[0] + sentence[8:11] + sentence[-7:-4] \
            + sentence[-2:]
print(substring)
```

(3 marks)

ID:

- e) Complete the `main()` function that asks the user to enter a word. The function then swaps the first and last letters of the word, displaying the result. In the example below the user has entered the word **Microsoft**:

```
Please enter a word: Microsoft
The new word is ticrosofM
```

You can assume that the user will always enter a word at least 2 characters long.

```
def main():
```

(8 marks)

```
main()
```

Question 2 (20 marks)

a) What is the output of the following code?

```
import random
var1 = round(57.789, 1)
var2 = abs(3 - 60)
var3 = random.randrange(1, 59, 2)
print(max(var1, var2, var3))
```

(3 marks)

b) Given the following code:

```
object1 = 'No' * int(3.14)
object2 = round(3.1415926, 5)
object3 = round(3, 5)
```

what is the type of the three Python objects: object1, object2, and object3?

```
object1 is of type:
object2 is of type:
object3 is of type:
```

(3 marks)

c) What is the output of the following code?

```
word = 'exponent'
number1 = len(word[:2])
number2 = len(word[-3:])
print(number1 ** number2)
```

(3 marks)

d) What is the output of the following code?

```
greeting = "Easy Question"
calculation = greeting.find("noi") + greeting.rfind("e") * \
              greeting.find("asy")
print(float(calculation))
```

(3 marks)

- e) Complete the `main()` function so that it asks the user to enter a string. The function will then create a new string, where one character in the original string, selected at random, is replaced by a random letter from the alphabet. The function will then print the new string. The example below shows a possible output produced by the completed function if the user enters the string "**Videogames**".

```
Please enter a string: Videogames
```

```
Your new string is: Videogases
```

You can assume that the user will always enter a word at least 1 character long.

```
import random
```

```
def main():
```

```
    alphabet = "abcdefghijklmnopqrstuvwxy"
```

(8 marks)

```
main()
```


ID:

- c) Using the code trace technique taught in lectures, perform a code trace on the following program and show the output.

```
def first(num1):  
    num2 = 3  
    print("1.", num1)  
    return second(num1 + num2) - 3
```

```
def second(num1):  
    print("2.", num1)  
    return num1 * 2
```

```
def main():  
    num1 = 5  
    num2 = first(num1)  
    print("3.", num2)  
    num2 = second(num2 - num1)  
    print("4.", num2)
```

```
main()
```

	Output
	
	
	
	

(8 marks)

Question 4 (20 marks)

a) Complete the output produced by the following code:

```
def function3(num):
    div = 1
    result = ""
    while div <= num:
        if num % div == 0:
            result = result + " " + str(div)
            div = div + 1
    return result

def main():
    number = 27
    print("1:" + function3(number))

    number = 36
    print("2:" + function3(number))

main()
```

1:

2:

(6 marks)

b) Write the following boolean expressions. Assume that the variable, `value`, has been initialized.

`value` is less than 80 but greater than or equal to 75

(2 marks)

`value` is not equal to either 100 or 0

(2 marks)

`value` between 50 and 60 (inclusive) or equal to 75

(2 marks)

ID:

- c) Complete the following function `get_grade()` which converts an integer mark into its corresponding grade. The grade boundaries to be used in this program are as follows.

A – greater than or equal to 90;
B – less than 90 but greater than or equal to 80;
C – less than 80 but greater than or equal to 70;
D – less than 70 but greater than or equal to 60;
F – less than 60.

For instance, the following program with the completed function gives the output:

89 – B grade

65 – D grade

```
def get_grade(mark):
```

(8 marks)

```
def main():  
    mark = 89  
    print(mark, "-", get_grade(mark), "grade")  
  
    mark = 65  
    print(mark, "-", get_grade(mark), "grade")  
  
main()
```

Question 5 (20 marks)

a) Given the following code:

```
object1 = ['Fun', 'is', 'just']
object2 = object1 + ['a point of view']
object3 = object1[2]
object4 = len(object2[2])
```

what is the type of each of the four Python objects: object1, object2, object3 and object4?

```
object1 is of type:
object2 is of type:
object3 is of type:
object4 is of type:
```

(4 marks)

b) Complete the output produced by the following code:

```
list1 = [2, 1, 3, 5, 4]
print("1.", list1)

list1[4] = list1[1] + list1[2]
list1[3] = list1[2] + list1[3]
list1[1] = list1[1] * list1[1] + 2
print("2.", list1)

list1[-2] = list1[-1]
list1[-1] = list1[-2]
print("3.", list1)
```

1. [2, 1, 3, 5, 4]

2.

3.

(5 marks)

ID:

- c) In the boxes below, show each element of `list3` after the following code has been executed. Use as many of the boxes as you need.

```
list1 = [2, 1, 3, 5, 4]
list2 = [8, 4, 5, 2]
list3 = []
for element in list1:
    if element in list2:
        list3 = list3 + ['A']
    else:
        list3 = list3 + ['B']
```

0	1	2	3	4	5	6	7

(5 marks)

- d) Complete the `get_list_of_minimums()` function which is passed two lists of integers as parameters. The function returns a list containing the minimum of each two corresponding elements from the two parameter lists. You can assume that both parameter lists contain exactly the same number of elements. For example, the following code with the completed function:

```
list3 = get_list_of_minimums([3, 6, 9, 2], [1, 8, 5, 3])
print(list3)
```

prints:

```
[1, 6, 5, 2]
```

```
def get_list_of_minimums(list1, list2):
```

(6 marks)

– Overflow Sheet 1 –

Write the question number and letter next to your answer. You must ALSO indicate in the allotted space that you have used the overflow sheet.

– Overflow Sheet 2 –

Write the question number and letter next to your answer. You must ALSO indicate in the allotted space that you have used the overflow sheet.

– Overflow Sheet 3 –

Write the question number and letter next to your answer. You must ALSO indicate in the allotted space that you have used the overflow sheet.