Student ID number:

_____

Pages: 6

Questions: 9

UNIVERSITY OF TASMANIA

KIT100 Programming Preparation

Test 2, Semester x, xxxx

SAMPLE

Time allowed: 60 minutes

Instructions:

- Write your student number in the box above. Answer ALL 9 questions by circling the correct answers provided in this test paper with a black or blue pen.
- This is a closed-book test and do not allow using Python software.
- This is a pass/fix/fail test, so there are no numerical marks. If you only make a small number of mistakes you will have the opportunity to correct them and explain your corrections to your tutor.

1. Complete the table by circling `True` or `False` in the Answer columns for each of the following list-related statements to indicate whether the **final statement** evaluates to True or False.

Logical expression	Answer (circle)	
`lst = []` `len(lst) == 0`	True	False
`lst = [1,2,3]` `lst.pop() == 1`	True	False

`lst = [1,2,3]`
`lst.append(4)`
`lst[3] == 4`
	True	False

`lst = [1,2,3]`
`lst.insert(0,1)`
`lst == [1,2,3,1]`
	True	False

`lst = ["a","b"]` `lst.count("A") == 1`	True	False

2. Consider a python program with the following statement:

```
myString = "Aardvarks Always Accept Albatrosses "
```

A programmer has to determine how many times the letter 'a' (upper or lower case) occurs in the string referred to by `myString`. Which one of the following will do this correctly?

Hint: *think about the order of operations, the outermost (right-most) method is called last!*

A. `count = myString.count("a").lower()`

B. `count = myString.lower().count("a")`

C. `count = myString.upper().count("a")`

D. `count = myString.count("A").upper()`

3. What will be the output from the following python code?

```
lst = [1,2,3,2]
lst.reverse()
lst.extend([4,5])
print(lst)
```

A. `[2,3,2,1,4,5]`
B. `[1,2,3,2,4,5]`
C. `[4,5,1,2,3,2]`
D. `[4,5,2,3,2,1]`

4. What is the purpose of the `__str__` method when it is defined in a Python class? When does it typically execute?

A. Its purpose is to initialise an object's **data attributes**. It executes immediately and only one time after the object is created.
B. Its purpose is to **produce a string representation** of an object's data attributes. It executes immediately **every time the object is used**.
C. Its purpose is to **produce a string representation** of an object's data attributes. It executes whenever you **attempt to print** the object.
D. It's only ever used to tell python how to create GUIs. It executes **whenever graphics** are used.

5. How many **syntax** errors are in the following python code?

```
myList = [1,2,3,4
myOption = input("Do you want to print the list?)

if myOption == "yes":
print(myList)
```

A. one
B. two
C. three
D. There are no syntax errors in the code

6. Given a function defined as below:

```
def multiply(num1, num2):
    result = num1 * num2
    return result
```

What will be displayed if the statement: `print(multiply(4,5))` is executed:

A. `20`
B. `4*5`
C. `multiply(4,5)`
D. Nothing will be displayed

7. Given a function defined as below:

```
def hello(greeting = "a default value!"):
    print(greeting)
```

What is the output of the following function calls:

```
hello()
hello("there")
```

A. *(a blank line)*
```
there
```
B.
```
a default value
there
```
C.
```
greeting
greeting
```
D. Nothing will be displayed

8. Given the program below:

```
class Pet:
    def __init__(self, name, species):
        self.name = name
        self.species = species
    def getName(self):
        return self.name
    def getSpecies(self):
        return self.species
    def __str__(self):
        return ("%s is a %s") % (self.name, self.species)
davesPet = Pet("Kitty", "Cat")
```

What is the output of the statement: `print(davesPet)`

A. `Kitty`

B. `Cat`

C. `Kitty is a Cat`

D. Nothing will be displayed as you can't print an object

Continued…

9. Examine the program below (line numbers are included for clarity):

```
 1 class Employee:
 2     def __init__(self, empNum, empName):
 3         theEmpNum = empNum
 4         self.theEmpName = empName
 5 
 6     def getEmpNum(self):
 7         return theEmpNum
 8 
 9     def getEmpName(self):
10         return self.theEmpName
11 
12 myEmployeeObject = Employee(16,"Jane Smith")
13 print(myEmployeeObject.getEmpNum())
```

The last line of code (line 13) causes an error to occur:

```
NameError: name 'theEmpNum' is not defined
```

What one change should you do to fix the error?

A. On line 7, put `self.` before `theEmpNum` i.e. `self.theEmpNum`

B. On line 7, use `empNum` instead of `theEmpNum`

C. On line 13, use `print(myEmployeeObject.getEmpNum(self))`

D. On line 13, use `print(getEmpNum())`