

Seat Number

Family Name _____

First Name _____

Semester Two Final Examinations, 2019

This paper is for St Lucia Campus students.

Reading Time: 10 minutes

This is a Central Examination

This is a Closed Book Examination - no materials permitted

During reading time - write only on the rough paper provided

This examination paper will be released to the Library

Materials Permitted In The Exam Venue:

(No electronic aids are permitted e.g. laptops, phones)

Calculators - No calculators permitted

Materials To Be Supplied To Students:

1 x Multiple Choice Answer Sheet

Instructions To Students:

Additional exam materials (eg. answer booklets, rough paper) will be provided upon request.

Answer all questions on the MCQ sheet provided.

Total questions: 40

Total marks: 40

For Examiner Use Only

Question	Mark
----------	------

[illegible]

Total

For all questions, please choose the most appropriate answer if it appears that more than one option is a potentially correct answer. All coding questions relate to the Python 3 programming language. If an evaluation produces an error of any kind, choose Error as your answer. Different questions may have different numbers of choices. Each question is worth one mark.

1. What does the expression `(12.4 - 5.4) % 3` evaluate to?
 - a) 1
 - b) 1.0
 - c) 2.3333
 - d) Error
2. What does the expression `3 + 7 // 4` evaluate to?
 - a) 2.5
 - b) 4
 - c) 4.75
 - d) Error
3. What does the expression `11.0 % 3 ** 2` evaluate to?
 - a) 4
 - b) 2
 - c) 4.0
 - d) 2.0
 - e) Error
4. What does the expression `(2, 3) + (4)` evaluate to?
 - a) `(2, 3, 4)`
 - b) `(2, 3)`
 - c) `(6, 7)`
 - d) Error
5. What does the expression `['g', 'y'] < ['b', 'd']` evaluate to?
 - a) True
 - b) False
 - c) ''
 - d) Error

6. What is the result of `not 4 > 2 and 2 < 3`?
- a) 2
 - b) True
 - c) False
 - d) Error
7. What will be printed out when the following loop is executed? (Note that `\n` corresponds to a new line).
- ```
for a,b in [(3,4),(6,7)]:
 print(a,b)
```
- a) (3,4) 0\n(6,7) 1
  - b) 3 0 4 1 6 2 7 3
  - c) 3 4\n6 7
  - d) None of the above
8. What is the value of `x` after the following statements are evaluated?
- ```
x = ['a', 'b', 'c']  
y = x  
y[1] = 0  
x[0] = 1
```
- a) 0
 - b) ['a']
 - c) [1, 0, 'c']
 - d) [1, 'b', 'c']
 - e) Error
9. After the assignment `s1 = "Silver lining"`, which of the following statements assigns 'r l' to `s2`?
- a) `s2 = s1[5:7]`
 - b) `s2 = s1[5:8]`
 - c) `s2 = s1[-8:-6]`
 - d) `s2 = s1[-6:-8]`
 - e) More than one of the above is correct.

10. After the assignment `s1 = "Silver lining"`, what will the value of `s2` be after the following command is entered?

```
s2=s1[: : -2]
```

- a) 'gii elS'
- b) 'ngSilver lini'
- c) 'gSilver linin'
- d) None of the above

11. What is the value of `y` after the following statements are evaluated?

```
x=[2,[3,5,8],7]
```

```
y=x[1,2]
```

- a) 5
- b) 8
- c) 7
- d) Error

12. Given the following code:

```
x = input("Please enter the first number: ")
y = input("Please enter the second number: ")
print ("x - y =", x - y)
```

and assuming that the user inputs 7 and 3 respectively. What would be the output?

- a) `x - y = x - y`
- b) `x - y = 7 - 3`
- c) `x - y = 4`
- d) Error

13. After executing the following code:

```
a={1:"A",2:"B",3:"C"}  
b={3:"D",5:"E"}  
a.update(b.get(4,'Error'))
```

what would be the contents of a?

- a) {1: 'A', 2: 'B', 3: 'C', 5: 'E'}
- b) {1: 'A', 2: 'B', 3: 'D', 5: 'Error'}
- c) {1: 'A', 2: 'B', 3: 'C', 4: 'Error'}
- d) Error
- e) None of the above

14. What is the value of d2 after the following statements are evaluated?

```
d = {1:'a', 2:'b', 3:'c'}  
d2=d.update({5:['def']})
```

- a) {1:'a', 2:'b', 3:'c'}
- b) {1: 'a', 2: 'b', 3: 'c', 5: ['def']}
- c) {1: 'a', 2: 'b', 3: 'c', 5: 'def'}
- d) None
- e) Error

15. Select the correct statement from the following:

- a) Strings, tuples, and dictionaries are immutable
- b) Lists and integers are mutable
- c) Booleans are mutable
- d) None of the above are true

16. After the assignment `z='two \t \t pairs'` what does the expression `'\t'.join(z.split('\t'))` evaluate to?

- a) 'two \t pairs'
- b) 'two \t \t pairs'
- c) 'two\tpairs'
- d) None of the above

17. What is the value of a after the following code is executed?

```
def f(x):  
    a = 70  
    x += a  
    return a
```

```
a = 50  
f(a)
```

- a) 50
- b) 70
- c) 120
- d) None of the above

18. What is the value of x after the following code is executed?

```
def f(l, a, b) :  
    l.append(a)  
    l = l + [b]  
    return l
```

```
x = [5, 9]  
x = f(x, 2, 1) + x
```

- a) [5, 9, 2, 1]
- b) [5, 9, 2, 1, 5, 9]
- c) [5, 9, 2, 1, 5, 9, 2]
- d) [5, 9, 2, 1, 5, 9, 2, 1]
- e) Error

19. Which of the following descriptions best describe the purpose of the following function?

```
def f() :  
    k = 0  
    g = int(input('Please input an integer: '))  
    while g != 0 :  
        if g % 2 != 0 :  
            k += g  
        g = int(input('Please input an integer: '))  
    return k
```

- a) It is an infinite loop as the while loop condition can never be false.
- b) It returns the sum of every second integer entered.
- c) It returns the sum of all odd integers entered.
- d) It returns the sum of all even integers entered.

20. Which of the following descriptions best describe the purpose of the code below?

```
n=int(input("Enter a number:"))
total = 1
while(n > 0):
    proc_num = n % 10
    total=total * proc_num
    n=n // 10
print(total)
```

- a) It multiplies together every 10th number entered and prints the result.
- b) It multiplies together all the numbers between 1 and 10 and prints the result
- c) It multiplies all digits in a number and prints the result
- d) It does none of the above

21. What will be the value of *y* after the following statements are executed?

```
d1={1:'a', 2:'b'}
d2=d1
d2[3]='c'
y=d1.get(3,'e')
```

- a) 'e'
- b) 'c'
- c) 'b'
- d) None
- e) Error

The next 3 questions refer to the following definition that is missing three lines of code. This function loads information contained in a data file into a dictionary. The data file contains 'major keys' (in square brackets) each on a line by itself, followed by one or more lines of the form `minor key = value`. For example, the contents of the data file, values.txt, is given below.

```
[Spain]
```

```
Time zone = 3:00
Capital = Madrid
```

```
[France]
```

```
Time zone = 3:00
Capital = Paris
```

The definition of the function, `load_data`, and the desired result of applying it to the above file is given below.

```
def load_data(data_file):
    """Return dictionary of the information contained in the
    file data_file.
    load_data(string) -> dict(string:dict(string:string))
    Precondition: the first non-blank line is a 'major key'
    """
    inf = open(data_file, 'r')
    lines = inf.readlines()
    inf.close()
    data_dict = {}
    for line in lines :
        if line.strip() == '': # ignore blank lines
            ## line 1 ##
        elif line.startswith '[': # major key
            ## line 2 ##
            major_value = {}
            data_dict[major] = major_value
        else:
            # minor key/value
            minor = line.split('=', 1)
            ## line 3 ##
    return data_dict

>>> load_data('values.txt')

{'Spain': {'Time zone': '3:00', 'Capital': 'Madrid'},
'France': {'Time zone': '3:00', 'Capital': 'Paris'}}
```

22. What is the required code for ## line 1 ##?

- a) `break`
- b) `pass`
- c) `data_dict = None`
- d) `major_value = None`
- e) None of the above

23. What is the required code for ## line 2 ##?

- a) `major = line.split(']',1)[0][:-1]`
- b) `major = line.split(']',1)[:-1]`
- c) `major = line.split(']',1)[0][1:]`
- d) `major = line.split(']',1)[1:]`
- e) None of the above

24. What is the required code for ## line 3 ##?

- a) `major_value[minor[0].strip()] = minor[1].strip()`
- b) `minor[1].strip() = major_value[minor[0].strip()]`
- c) `major_value[minor[1].strip()] = minor[0].strip()`
- d) `minor[0].strip() = major_value[minor[1].strip()]`
- e) None of the above

The following recursive function definition is used in the next two questions.

```
def fn1(x) :  
    if len(x) == x[0] :  
        return x  
    return fn1(x[3:] + [x[0]])
```

25. What will the function call `fn1([2, 4, 1, 1, 5])` return?

- a) `[5, 2, 1, 4, 1]`
- b) `[2,5]`
- c) `[1]`
- d) `RecursionError` will be raised due to maximum recursion depth being exceeded.

26. What will the function call `fn1([4, 1, 3, 2, 6])` return?

- a) `[7, 4, 1, 3, 2]`
- b) `[7, 1, 2]`
- c) `[3, 7]`
- d) `RecursionError` will be raised due to maximum recursion depth being exceeded.

The following partial definition of a SwimRecord class is used in the following three questions.

```
class SwimRecord(object) :
    def __init__(self, name, club) :
        """Parameters:
        name(str): swimmer's name
        club(str): swimmer's club
        swim_record(dict): record of swim times
        """
        self._name = name
        self._club = club
        self._swim_record = {}

    def update_swim_record(self, new_results: dict) :
        """Add the dictionary, 'new_results', into the
        current dictionary of swim results."""
        ## code block 1 ##

    def get_swim_results(self, swim_meet: str) :
        """Get swim results."""
        return self._swim_record.get(swim_meet, 'Err')

    def get_personal_best(self) :
        """Get the shortest time from all swim meets in the
        swim_record. Assume all swim meet events are over
        the same distance."""
        ## code block 2 ##
```

27. What is the required code for **## code block 1 ##**?

- a) self._swim_record += new_results
- b) self._swim_record.update(new_results)
- c) self._swim_record.append(new_results)
- d) None of the code blocks above are correct.

28. What is the required code for **## code block 2 ##**?

- a) return min(self._swim_record.items())
- b) return min(self._swim_record.values())
- c) return min(self._swim_record.keys())
- d) return self._swim_record.pop()
- e) More than one of the above

29. Assume that an object from the `SwimRecord` class has been created and has the name `john_sullivan`. Assume also that it contains an accurate record of the past year's swim results, including those for the 'Noosa' swim meet. Which of the following will return the swim results for the 'Noosa' swim meet for John Sullivan?

- a) `john_sullivan.get_swim_results('Noosa')`
- b) `john_sullivan._swim_record('Noosa')`
- c) `john_sullivan._swim_record['Noosa']`
- d) More than one of the above
- e) None of the above.

The next 2 questions refer to the following definition.

```
def m(x):  
    a,b = x  
    if a > b:  
        return (b, a+b)  
    elif a < b:  
        return (a, b*b+a*a)  
    else:  
        return (a,b)
```

30. What is the value of `n` after the following is evaluated?

```
n = m(3,4)
```

- a) (1, -3)
- b) (4, 6)
- c) (3, 2)
- d) (3, 1)
- e) Error

31. What is the value of `n` after the following is evaluated?

```
n = m((3,2))
```

- a) (2, 5)
- b) (3, 41)
- c) (3, 18)
- d) (2, 71)
- e) Error

The next five questions refer to the following class definitions.

```
class A(object) :
    def __init__(self, x) :
        self._x = x

    def m1(self, x) :
        return self.m2(x) * 2

    def m2(self, x) :
        return x + 1

class B(A) :
    def m2(self, y) :
        return self._x + y

class C(B) :
    def __init__(self, x, y) :
        super().__init__(x)
        self._y = y

    def m1(self, x) :
        return self._x - self._y

class D(B) :
    def __init__(self, x, y) :
        super().__init__(x)
        self._x += y
        self._y = y

    def m1(self, y) :
        return self._y - y

    def m2(self, x) :
        return super().m2(x) + x

a = A(1)
b = B(2)
c = C(2, 3)
d = D(2, 1)
```

32. What does `a.m1(2)` return?

- a) 2
- b) 4
- c) 5
- d) 6
- e) None of the above

33. What does `b.m1(3)` return?

- a) 4
- b) 6
- c) 8
- d) 10
- e) None of the above

34. What does `c.m2(3)` return?

- a) 3
- b) 4
- c) 5
- d) 6
- e) None of the above

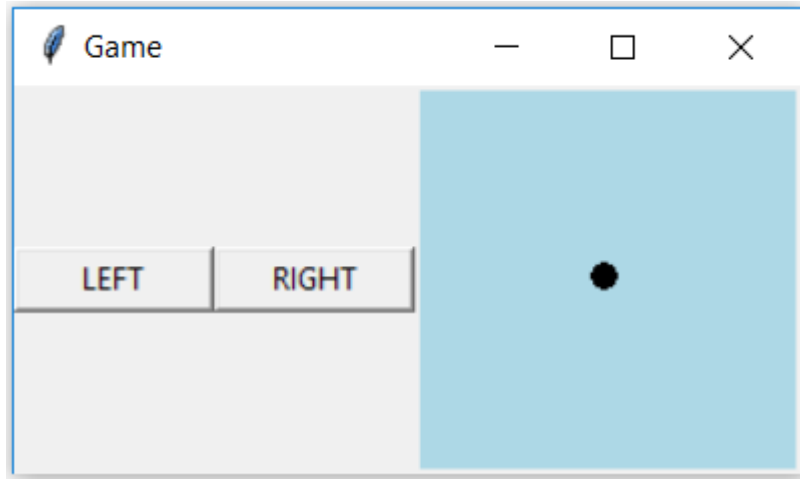
35. What does `d.m1(2)` return?

- a) -2
- b) -1
- c) 1
- d) 2
- e) None of the above

36. What does `d.m2(2)` return?

- a) 5
- b) 6
- c) 7
- d) 8
- e) None of the above

The next two questions relate to the following partial implementation of a GUI. The application has two buttons in a frame and custom canvas on which the circle can be moved left or right by clicking on the buttons. The completed GUI is shown in the image below. The code is provided on the next page.



```
import tkinter as tk

class Screen(tk.Canvas):
    def __init__(self, parent):
        super().__init__(parent, bg="light blue",
                          width=150, height=150)
        self._x, self._y = (width / 2, height / 2)
        self._redraw()

    def _redraw(self):
        """Redraw the screen after a move."""
        self.delete(tk.ALL)
        ## code block 1 ##

    def _move(self, dx, dy):
        """Move the circle by a given amount."""
        self._x += dx
        self._y += dy
        self._redraw()

    def move_left(self):
        self._move(-5, 0)

    def move_right(self):
        self._move(5, 0)

class Controls(tk.Frame):
    BUTTON_WIDTH = 10

    def __init__(self, parent, left, right):
        """Parameters:
            parent (Tk): Window for widget.
            left (method): Callback for "left button".
            right (method): Callback for "right button".
        """
        super().__init__(parent)
        ## code block 2 ##

class GameApp(object):
    def __init__(self, master):
        master.title("Game")
        screen = Screen(master)
        controls = Controls(master, screen.move_left,
                             screen.move_right)
        controls.pack(side=tk.LEFT)
        screen.pack(side=tk.LEFT, expand=True,
                     fill=tk.BOTH)
```

37. What is the required code for **## code block 1 ##**?

- a) `coords = (self._x - 5, self._y - 5,
 self._x + 5, self._y + 5)
create_oval(coords, fill="black", width=0)`
- b) `coords = (self._x - 5, self._y - 5,
 self._x + 5, self._y + 5)
self.create_oval(coords, fill="black", width=0)`
- c) `coords = (self._x - 5, self._y - 5,
 self._x + 5, self._y + 5)
Canvas.create_oval(coords, fill="black", width=0)`
- d) `coords = (self._x - 5, self._y - 5,
 self._x + 5, self._y + 5)
super().create_oval(coords, fill="black", width=0)`
- e) More than one of the above is correct.

38. What is the required code for **## code block 2 ##**?

- a) `left_btn = tk.Button(self, text="LEFT",
 width=10, command=left())
left_btn.pack(side=tk.LEFT)
right_btn = tk.Button(self, text="RIGHT",
 width=10, command=right())
right_btn.pack(side=tk.LEFT)`
- b) `left_btn = tk.Button(self, text="LEFT",
 width=10, command=Screen.left())
left_btn.pack(side=tk.LEFT)
right_btn = tk.Button(self, text="RIGHT",
 width=10, command=Screen.right())
right_btn.pack(side=tk.LEFT)`
- c) `left_btn = tk.Button(self, text="LEFT",
 width=10, command=self.left())
left_btn.pack(side=tk.LEFT)
right_btn = tk.Button(self, text="RIGHT",
 width=10, command=self.right())
right_btn.pack(side=tk.LEFT)`
- d) `left_btn = tk.Button(self, text="LEFT",
 width=10, command=left)
left_btn.pack(side=tk.LEFT)
right_btn = tk.Button(self, text="RIGHT",
 width=10, command=right)
right_btn.pack(side=tk.LEFT)`
- e) More than one of the above is correct.

39. What is the value of `z` after the following has been evaluated?

```
g = lambda x,y: (x+y)
xs = 'lots'
z = [g(x,y) for x in 'maze' for y in xs if y > 't']
```

- a) `[]`
- b) `['mt', 'at', 'zt', 'et']`
- c) `['m', 'ml', 'mo', 'mt', 'ms']`
- d) `['ml', 'ao', 'zt', 'es']`
- e) None of the above

40. After running the following code:

```
import random
xs=[1,2,3,4]
new_list=[(x,random.random()) for x in xs]
new_list.sort()
z=[(x,y) for y,x in new_list]
```

Which of the following represents the most plausible contents of `z`?

- a) Error
- b) `[(1, 0.3656826997131658), (2, 0.4789711218283632), (3, 0.20367358828920812), (4, 0.4651024789182844)]`
- c) `[(0.23845323656036166, 2), (0.5411763744080424, 4), (0.7368067435015173, 3), (0.9585633916983842, 1)]`
- d) `[(0.7070150251404196, 1), (0.9635956493452444, 2), (0.5960013756836279, 3), (0.9623962721301965, 4)]`
- e) None

END OF EXAMINATION