THE UNIVERSITY OF QUEENSLAND
AUSTRALIA

This exam paper must not be removed from the venue

Venue _____

Seat Number _____

Student Number |__|__|__|__|__|__|__|__|

Family Name _____

First Name _____

School of Information Technology and Electrical Engineering

EXAMINATION

Semester Two Final Examinations, 2017

CSSE1001 Introduction to Software Engineering

This paper is for St Lucia Campus students.

Examination Duration: 90 minutes

Reading Time: 10 minutes

Exam Conditions:

This is a Central Examination

This is a Closed Book Examination - specified 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 - Casio FX82 series or UQ approved (labelled)

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 supplied Multiple Choice Answer Sheet. Each question is worth one mark. Total marks: 40

For Examiner Use Only

Question	Mark

Total _____

In all questions, please choose the most appropriate answer if it appears that more than one matches. All questions relate to the Python programming language. If an evaluation produces an error of any kind then choose Error as your answer. Different questions can have different numbers of choices. Each question is worth one mark.

Question 1. What does the expression 4.0 + 9//2 evaluate to?

(a) 8.5

(b) 5.0

(c) 8.0

(d) 85.0

(e) None of the above

Question 2. What does the expression 2 * ['a', 'b'] evaluate to?

(a) [2 *'a', 2 *'b']

(b) ['a a', 'b b']

(c) ['a', 'b', 'a', 'b']

(d) ['a', 'b'], ['a', 'b']

(e) None of the above

Question 3. What does the expression list('2') + ['3', '4'] evaluate to?

(a) ['2', '3', '4']

(b) ['5', '6']

(c) [['2'], '3', '4']

(d) [['2', '3'],['2', '4']]

(e) Error

Question 4. What does the expression '3' + ['1', '5'] evaluate to?

(a) [4, 8]

(b) ['4', '8']

(c) ['31', '35']

(d) ['3', '1', '5']

(e) Error

Question 5. What does the expression (5,4) + ((3)) evaluate to?

(a) (5, 4, 3)

(b) (5, 4, (3))

(c) (8, 7)

(d) (9, 3)

(e) Error

Question 6. After the assignment x = 'Dogs and cats', which of the following assigns 'n' to the variable y?

(a) y = x[6]

(b) y = x[7]

(c) y = x[-8]

(d) y = x[7:8]

(e) More than one of the above is correct.

Question 7. After the assignment x = 'Dogs and cats', which of the following assigns 'and' to the variable y?

(a) y = x[6 : 8]

(b) y = x[6: 9]

(c) y = x[-8:-5]

(d) y = x[-8:-6]

(e) None of the above.

Question 8. After the assignment x = 'Critical mass', which of the following assigns 'cit' to the variable y?

(a) y = x[5 : 2: -1]

(b) y = x[-8 : -11]

(c) y = x[-8 : -11: -1]

(d) y = x[5 : 2]

(e) More than one of the above is correct.

Question 9. After the assignment `x = 'Critical mass'`, which of the following assigns 'tam' to the variable `y`?

(a) `y = x[3 : 10 : 3]`

(b) `y = x[3 : 3 : 10]`

(c) `y = x[4: 10 : 3]`

(d) `y = x[4: 3 : 9]`

(e) More than one of the above is correct.

The next 2 questions refer to the following definition.

```
def g(p):
    z = p.pop(0)
    p.extend(z)
    return p
```

Question 10. What is the value of y after the following is evaluated?

```
y = ['h','i','j']
g(y[:]).extend(g(y))
```

(a) `['h','i','j']`

(b) `['i','j','h']`

(c) `['i','j','h','j','h','i']`

(d) `['i','j','h','i','j','h']`

(e) Error

Question 11. What is the value of y after the following is evaluated?

```
y = ['h','i','j']
g(y).extend(g(y[:]))
```

(a) `['h','i','j']`

(b) `['i','j','h']`

(c) `['i','j','h','j','h','i']`

(d) `['i','j','h','i','j','h']`

(e) Error

Question 12. What is the value after the following is evaluated?

```
y = [4, 5, 6, 7]
y.insert(3, y.pop(2))
```

(a) [4, 5, 7, 6]

(b) [4, 6, 7, 5]

(c) [4, 5, 6, 7]

(d) [4, 5, 6, 4]

(e) None of the above

Question 13. What is the value of v after the following assignments are made?

```
z='roses are red'
v=':'.join(z.split())
```

(a) ['roses':'are ':'red ']

(b) 'roses:are:red'

(c) 'roses':'are':'red'

(d) ['roses:are:red']

(e) Error

Question 14. What is the value of a after the following is evaluated?

```
def f(x, y):
    y = y + [x]
    return y
a = [2,1]
a = f(1,a) + 2 * a
```

(a) [1, 3, 1, 2]

(b) [2, 1, 1, 2, 1, 2, 1]

(c) [2, 1, 1, 4, 2]

(d) [1, 2, 3, 1, 2, 3, 2]

(e) None of the above

Question 15. What is the value of x after the following is evaluated?

d = {'John' : {'ID': 234, 'Orders': ['soap', 'bath mat']}, 'Lavinia': {'ID': 732, 'Orders': ['mouthwash', 'toothpaste']}}

x=d.get('John').get('Orders',[])

(a) []

(b) ['soap', 'bath mat']

(c) {}

(d) Error

Question 16. What is the value of x after the following is evaluated?

d = {'John' : {'ID': 234, 'Orders': ['soap', 'bath mat']}, 'Lavinia': {'ID': 732, 'Orders': ['mouthwash', 'toothpaste']}}

x=d.get('Mary',{}).get('Orders',[])

(a) ['soap', 'bath mat']

(b) []

(c) ['mouthwash', 'toothpaste']

(d) Error

Question 17. What is the value of x after the following is evaluated?

d = {'John' : {'ID': 234, 'Orders': ['soap', 'bath mat']}, 'Lavinia': {'ID': 732, 'Orders': ['mouthwash', 'toothpaste']}}

x=d.get('John',{}).get('Orders',[]).append('toothbrush')

(a) ['soap', 'bath mat', ['toothbrush']]

(b) ['soap', 'bath mat', 'toothbrush']

(c) None

(d) Error

The next three questions refer to the following partial definition of a GoCard class.

```
class GoCard(object):
    def __init__(self, owner):
        self._owner = owner
        # balance on the card in dollars
        self._balance = 5.00

def update_balance(self, amount):
    """Update the balance with amount.
    If amount > 0, the GoCard is topped up by amount
    If amount < 0, the balance is reduced by the amount of the trip"""
    ## line 1 ##

def get_balance(self):
    """Return the balance."""
    ## line 2 ##
```

Assume that the following has been evaluated.

```
martin = GoCard("Martin Avis")
```

Question 18. What is the required code for ## line 1 ##?

(a) balance += amount

(b) __balance__ +=amount

(c) self.balance += amount

(d) self._balance += amount

(e) None of the above.

Question 19. What is the required code for ## line 2 ##?

(a) print GoCard._balance

(b) return GoCard._balance

(c) print self._balance

(d) return self._balance

(e) None of the above.

Question 20. Which of the following correctly updates the GoCard balance for fred after a trip costing 3.50?

(a) update_balance(fred, -3.50)

(b) fred.update_balance(self, -3.50)

(c) fred.update_balance(3.50)

(d) fred.update_balance(self, -3.50)

(e) None of the above.

The next 3 questions refer to the following definition.

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

Question 21. What is the value of n after the following is evaluated? `n = m(2,1)`

(a) (2, -3)

(b) (3, 5)

(c) (4, 2)

(d) (2, 1)

(e) Error

Question 22. What is the value of n after the following is evaluated? n = m((4,3))

(a) (2, -14)

(b) (3, 25)

(c) (4, 10)

(d) (7, 28)

(e) Error

Question 23. What is the value of n after the following is evaluated? n = m(m((1,2)))

(a) (1, 26)

(b) (2, 18)

(c) (13, 31)

(d) (3, 23)

(e) None of the above

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.

[Austria]

Time zone = 3:00
Capital = Vienna

[Canada]

Time zone = 7:00
Capital = Ottawa

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, 'rU')
    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')
{'Austria': {'Time zone': '3:00', 'Capital': 'Vienna'}, 'Canada': {'Time zone': '7:00', 'Capital': 'Ottawa'}}
```

Question 24. 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

Question 25. 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

Question 26. What is the required code for ## line 3 ##?

Recall that a semi-colon allows you to write two or more statements on one line.

(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

Question 27. Which of the following statements about dictionaries is true?

(a) Dictionary keys can be any type.

(b) Only numbers or strings can be dictionary keys.

(c) Dictionary keys must be immutable types.

(d) Both dictionary keys and values must be immutable types.

(e) If the dictionary keys are mutable then the corresponding values must be immutable.

The next 3 questions refer to the following definition.

```
def f(xs):
    i = 0
    r = []
    while i >= 0 and i < len(xs):
        d,v = xs[i]
        r.append(v)
        i += d
    return r
```

Question 28. What is the value of zs after the following is evaluated?

```
zs = f([(1,'l'),(2,'m'),(3,'n'), (-1,'o')])
```

(a) ['l', 'm', 'o']

(b) ['l', 'm', 'o', 'n']

(c) ['l', 'm', 'n']

(d) Non-terminating execution (possibly resulting in an out of memory error)

(e) Error (other than non-termination/out of memory)

Question 29. What is the value of zs after the following is evaluated?

```
zs = f([(1,'l'),(-2,'m'),(3,'n'), (-2,'o')])
```

(a) ['l', 'm', 'n', 'o', 'n']

(b) ['l', 'm, 'n']

(c) ['l', 'm']

(d) Non-terminating execution (possibly resulting in an out of memory error)

(e) Error (other than non-termination/out of memory)

Question 30. What is the value of zs after the following is evaluated?

```
zs = f([(1,'l'),(-1,'m'),(4,'n'), (-2,'o')])
```

(a) ['l', 'm', 'n', 'o', 'n']

(b) ['l', 'm', 'o', 'n']

(c) ['l', 'm', 'n']

(d) Non-terminating execution (possibly resulting in an out of memory error)

(e) Error (other than non-termination/out of memory)

The next five questions refer to the class definitions and assignments given below.

```
class A(object):
    def __init__(self, x):
        self.x = x
    def g(self, x):
        return self.f(x)
    def f(self, x):
        return 3*x
class B(A):
    def g(self, y):
        return self.x + 2*y
class C1(B):
    def __init__(self, x, y):
        B.__init__(self, x)
        self.y = y
    def f(self, x):
        return self.x + self.y
class C2(B):
    def __init__(self, x, y):
        B.__init__(self, x)
        self.y = y
    def f(self, x):
        return x + self.x + 2* self.y
a = A(2)
b = B(1)
c1 = C1(2, 4)
c2 = C2(2, 4)
```

Question 31. What does the expression b.f(3) evaluate to?

(a) 6

(b) 7

(c) 8

(d) 9

(e) None of the above

Question 32. What does the expression a.f(3) evaluate to?

(a) 6

(b) 7

(c) 8

(d) 9

(e) None of the above

Question 33. What does the expression b.g(2) evaluate to?

(a) 5

(b) 6

(c) 7

(d) 8

(e) None of the above

Question 34. What does the expression c1.g(3) evaluate to?

(a) 6

(b) 7

(c) 8

(d) 9

(e) None of the above

Question 35. What does the expression c2.f(2) evaluate to?

(a) 10

(b) 11

(c) 12

(d) 13

(e) None of the above

Question 36. How do you create a root window?

(a) root = tk.rootWindow()
(b) root=tk.Window()
(c) root=tk.Frame()
(d) root=tk.Tk()

Question 37. To create a menu in a window, use _____

(a) menubar = tk.Menu(master)
(b) menubar = tk.MenuBar(master)
(c) menubar = tk.Menu()
(d) menubar = tk.MenuBar()

Question 38. How do you create an event loop?

(a) root.event()
(b) root.main()
(c) root.mainloop()
(d) root.eventloop()

This question relates to the incomplete recursive function definition below that provides indexing into a nested list. The following gives examples of calling the function.

```
>>> nested = [[[3, 6], 3], 11, [4, [3, 6]], [12, 2, 10]]

>>> recursive_index(nested, [])

[[[3, 6], 3], 11, [4, [3, 6]], [12, 2, 10]]

>>> recursive_index(nested, [2,1,0])

3
```

The first example above shows what happens when the second argument (the indexing list) is empty. The second example indexes the 0'th element of the 1'th element of the 2'th element of nested.

Question 39. What expression needs to replace #### in order to complete the following recursive definition for indexing a nested list

```
def recursiveIndex(nested, indexes):
    """Return the element of the required sublist of the nested list
    (nested) at the position specified by indexes
    Precondition: indexes specifies a "valid" element of the nested list """
    if indexes == []:
       return nested
    else:
        return ####
```

(a) recursive_index(nested[indexes[0]], indexes[:-1])

(b) recursive_index(nested[indexes[0]], indexes[1:])

(c) recursive_index(nested[indexes[-1]], indexes[:-1])

(d) recursive_index(nested[indexes[-1]], indexes[1:])

(e) None of the above

Question 40. What is the value of y after the following has been evaluated?

```
x = lambda u,v: (u*v)
y = lambda u,v: u-v
ug = [0, 1, 2]
z = [x(u,v) for u in ug for v in ug if y(u,v)]
```

(a) [0, 0, 2, 2]

(b) [0, 0, 2, 4]

(c) [0, 0, 0, 2, 0 2]

(d) [0, 0, 0, 2, 4, 6]

(e) Error

END OF EXAMINATION