

THE UNIVERSITY OF AUCKLAND

SUMMER SEMESTER, 2015

Campus: City

COMPUTER SCIENCE

TEST

Principles of Computer Science

(Time Allowed: 50 minutes)

Note:

- The use of calculators is NOT permitted.
- Compare the term test version number on the Teleform sheet supplied with the version number above. If they do not match, ask the supervisor for a new sheet.
- Enter your name and student ID on the Teleform sheet. Your name should be entered left aligned. If your name is longer than the number of boxes provided, truncate it.
- Answer all **Multiple-choice** questions on the Teleform answer sheet provided. Answer Section **B** in the space provided in this booklet. Attempt all questions.
- Use a dark pencil to mark your answers in the multiple choice answer boxes on the Teleform sheet. Check that the question number on the sheet corresponds to the question number in this question/answer book. If you spoil your sheet, ask the supervisor for a replacement.
- Write your answers in the space provided in the short answer section. Write as clearly as possible. The space provided will generally be sufficient but is not necessarily an indication of the expected length. Extra space is provided at the end of this exam book.

Surname:	
First Name(s):	
Student ID:	
Login Name(UPI):	
Lab Time:	

MARKERS ONLY

	Question	Mark	Out Of
1 – 23	Multiple Choice		25
24 – 27	Written		25
	TOTAL		50

CONTINUED

The `Rectangle` class below is used by the following SEVEN questions.

```
class Rectangle:
    def __init__(self, w, h):
        self.width = w
        self.height = h

    def perimeter(self):
        return (self.width + self.height) * 2

    def area(self):
        return self.width * self.height

    def scale(self, factor):
        self.width *= factor
        self.height *= factor

    def __lt__(self, other):
        if not isinstance(other, Rectangle):
            return False
        return self.area() < other.area()

    def __eq__(self, other):
        if not isinstance(other, Rectangle):
            return False
        return self.area() == other.area()
    ...
```

```
r = Rectangle(5, 10)
s = Rectangle(10, 5)
t = Rectangle(2, 4)
```

Question 1

[1 mark] Which of the following code fragments will cause a `TypeError` exception?

- (a) `print(r < s)`
- (b) `print(r.perimeter())`
- (c) `print(r.area())`
- (d) `print(r <= s)`
- (e) `print(r.scale(10))`

Question 2

[1 mark] What is the output of the following code?

```
r = Rectangle(5, 10)
s = Rectangle(10, 5)
r.scale(10)
print(r == s, r.perimeter())
```

- (a) False 150
- (b) True 300
- (c) False 300
- (d) False 30
- (e) True 30

Question 3

[1 mark] What is the output of the following code?

```
r = Rectangle(5, 10)
s = Rectangle(10, 5)
t = Rectangle(2, 4)
print(r == s, s < r, s < t)
```

- (a) False False False
- (b) False True False
- (c) True True False
- (d) False False True
- (e) True False False

Question 4

[1 mark] What is the output of the following code?

```
r = Rectangle(5, 10)
s = Rectangle(10, 5)
u = s
print(r == s, r is s, s == u, s is u)
```

- (a) True False False False
- (b) True True False False
- (c) True False True True
- (d) True False True False
- (e) False True False True

Question 5

[1 mark] What is the output of the following code?

```
r = Rectangle(5, 10)
s = Rectangle(10, 5)
a = [r, s]
b = a.copy()
print(a == b, a is b, a[0] == b[0], a[0] is b[0])
```

- (a) True True True True
- (b) True False True True
- (c) True False True False
- (d) True True False False
- (e) False True False True

Question 6

[1 mark] What is the output of the following code?

```
r = Rectangle(5, 10)
s = Rectangle(10, 5)
a = [r, s]
d = copy.deepcopy(a)
print(a == d, a is d, a[0] == d[0], a[0] is d[0])
```

- (a) True True False False
- (b) False True False True
- (c) True True True True
- (d) True False True False
- (e) True False True True

Question 7

[1 mark] What is the output of the following code?

```
r = Rectangle(5, 10)
print(r == 50)
```

- (a) True
- (b) False
- (c) None
- (d) AttributeError: 'int' object has no attribute 'area'
- (e) None of the above

Question 8

[1 mark] What is the output of the following code?

```
words = 'red green yellow blue'.split()
value = [(w, len(w)) for w in words]
print(value)
```

- (a) [('red', 3), ('green', 5), ('yellow', 6), ('blue', 4)]
- (b) [3, 5, 6, 4]
- (c) ['red', 'green', 'yellow', 'blue']
- (d) ['red', 3, 'green', 5, 'yellow', 6, 'blue', 4]
- (e) None of the above

Question 9

[1 mark] What is the output of the following code?

```
my_list = [x - 1 for x in range(0,10) if x % 2 == 1]
print(my_list)
```

- (a) [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
- (b) [1, 3, 5, 7, 9]
- (c) [0, 2, 4, 6, 8, 10]
- (d) [0, 2, 4, 6, 8]
- (e) None of the above

The `check_exceptions` function below is used by the following two questions:

```
def check_exceptions(input_value):
    result = ''
    try:
        num = int(input_value)
        result += 'a'
        try:
            num = 200 / 0
            result += 'b'
        except TypeError:
            result += 'c'
        except:
            result += 'd'
    finally:
        result += 'e'
    except ZeroDivisionError:
        result += 'f'
    except ValueError:
        result += 'g'
    finally:
        result += 'h'
    return result
```

Question 10

[1.5 marks] What is the output of the following statement to the `check_exceptions` function above?

```
print(check_exceptions('Hello'))
```

- (a) acegh
- (b) afgh
- (c) adeh
- (d) abcdefgh
- (e) gh

Question 11

[1.5 marks] What is the output of the following statement to the `check_exceptions` function above?

```
print(check_exceptions(0))
```

- (a) aeh
- (b) adebh
- (c) adeh
- (d) afgh
- (e) abefgh

Question 12

[1 mark] Consider the following code:

```
class A:
    def __init__(self, s):
        self.s = s

    def print(self):
        print(s)

a = A("Welcome")
a.print()
```

- (a) The program would run if you change `print(s)` to `print(self.s)`.
- (b) The program has an error because class `A` should have a `print` method with signature `print(s)`.
- (c) The program has an error because class `A` does not have a constructor.
- (d) The program has an error because class `A` should have a `print` method with signature `print(self, s)`.
- (e) None of the above

Question 13

[1 mark] Consider the following code:

```
class A:
    def __init__(self, s):
        self.s = s

    def print(self):
        print(self.s)

a = A()
a.print()
```

- (a) The program runs fine and prints nothing.
- (b) The program has an error because class `A` does not have a constructor.
- (c) The program has an error because the constructor is invoked without an argument.
- (d) The program has an error because `s` is not defined in `print(s)`.
- (e) None of the above

Question 14

[1 mark] What is the output of the following code fragment?

```
result = ''
try:
    list = 5 * [0]
    x = list[5]
    result += 'a'
except IndexError:
    result += 'b'
except:
    result += 'c'
finally:
    result += 'd'
print(result)
```

- (a) cd
- (b) bcd
- (c) abcd
- (d) d
- (e) bd

Question 15

[1 mark] What is the output of the following code fragment?

```
try:
    list = 10 * [0]
    x = list[9]
    print("Done", end=' ')
except IndexError:
    print("Index out of bound", end=' ')
else:
    print("Nothing is wrong", end=' ')
finally:
    print("Finally we are here", end=' ')
```

- (a) Index out of bound Nothing is wrong Finally we are here
- (b) Nothing is wrong Finally we are here
- (c) Done Nothing is wrong
- (d) Done Nothing is wrong Finally we are here
- (e) None of the above

Consider the following JSON documents:

- I. {"name":"Fred Flintstone";"occupation":"Miner";"wife":"Wilma"}
- II. {"title":"Star Wars", "quotes":["Use The Force","These are not the Droids you are looking for"],"director":"George Lucas"}
- III. {}
- IV. {"city":"New York", "population", 7999034, boros:{"queens", "manhattan", "staten island", "the bronx", "brooklyn"}}
- V. {"a":1, "b":{"b":1, "c":"foo", "d":"bar", "e":[1,2,4]}}

Question 16

[1 mark] Which of the above are **INVALID**?

- (a) III only
- (b) I and IV
- (c) IV only
- (d) I, III and IV
- (e) None of the above

Question 17

[1 mark] Which of the following is **INCORRECT** regarding to JSON Structures?

- (a) Each name is followed by ":" (colon) and the name/value pairs are separated by "," (comma)
- (b) A JSON object begins with "{" (left brace) and ends with "}" (right brace)
- (c) A JSON object is an ordered set of name/value pairs.
- (d) A collection of name/value pairs
- (e) None of the above

Question 18

[1 mark] Which of the following complexity is $O(n \log n)$?

- (a) $23n \log n + 50 + n * n$
- (b) $45n + 45n \log n + 503$
- (c) $n * n * n + n \log n$
- (d) $300n + 400n * n$
- (e) None of the above

Question 19

[1 mark] What is the big-O complexity of the following code fragment?

```
my_list2 = [2, 4, 6, 8, 10]
for i in range(len(my_list2)):
    print(my_list2.pop(-1))
```

- (a) $O(\log n)$
- (b) $O(n \log n)$
- (c) $O(n)$
- (d) $O(n^2)$
- (e) None of the above

Question 20

[1 mark] What is the big-O complexity of the following two functions?

```
def complexity1(n):
    for i in range(n):
        for j in range(n):
            for k in range(n):
                k = 2 + 2

def complexity2(n):
    for i in range(n):
        k = 2 + 2
    for j in range(n):
        k = 2 + 2
    for k in range(n):
        k = 2 + 2
```

- (a) complexity1: $O(n^3)$, complexity2: $O(n)$
- (b) complexity1: $O(n^3)$, complexity2: $O(\log n)$
- (c) complexity1: $O(n)$, complexity2: $O(n)$
- (d) complexity1: $O(n)$, complexity2: $O(n^3)$
- (e) None of the above

Question 21

[1 mark] What is the output of the following code fragment?

```
my_queue = Queue()
my_linked_list = UnorderedList()
my_queue.enqueue('1')
my_queue.enqueue('2')
my_queue.enqueue('3')
while not my_queue.is_empty():
    my_linked_list.add(my_queue.dequeue())
for elt in my_linked_list:
    print(elt, end = ' ')
```

- (a) 1 2 3 1 2 3
- (b) 1 2 3
- (c) 3 2 1
- (d) 1 2 3 3 2 1
- (e) None of the above

Question 22

[1.5 marks] Suppose `list1` is a Python List and `list2` is an `UnorderedList`. Both contains 1 million double values. Consider the following code fragment:

```
A:
for i in range(100):
    list1.append(i);
```

```
B:
for i in range(100):
    list2.add(i);
```

Which of the following is true?

- (a) Code fragment B runs faster than code fragment A because the time complexity for code fragment B is $O(n)$ and for A is $O(n^2)$.
- (b) Both code fragment A and B have the same time complexity, but A runs faster because `LinkedList` has more overhead on creating object for each node in the linked list.
- (c) Code fragment A runs as fast as code fragment B because both code fragment A and B have the same time complexity $O(n^2)$.
- (d) Code fragment A runs faster than code fragment B because the time complexity for code fragment A is $O(n)$ and for B is $O(n^2)$.
- (e) None of the above

Consider the following:

- I. A stack can be viewed as a special type of list, where the elements are accessed, inserted, and deleted only from the end, called the top, of the stack.
- II. A queue represents a waiting list. A queue can be viewed as a special type of list, where the elements are inserted into the end (tail) of the queue, and are accessed and deleted from the beginning (head) of the queue.
- III. Since the insertion and deletion operations on a stack are made only at the end of the stack, using an array list to implement a stack is more efficient than a linked list.
- IV. Since deletions are made at the beginning of the list, it is more efficient to implement a queue using a linked list than a list.

Question 23

[1.5 marks] Which of the above statements is/are **TRUE**?

- (a) I only
- (b) II only
- (c) I & II
- (d) III & IV
- (e) All of the above

THIS PAGE HAS BEEN INTENTIONALLY LEFT BLANK.

CONTINUED

SECTION B

Answer all questions in this section in the space provided. If you run out of space then please use the Overflow Sheet and indicate in the allotted space that you have used the Overflow Sheet.

Question 24:

[3 marks]

Given the following dictionary:

```
Phonebook = { 'Sue' : '333-3333', 'Ann' : '222-2222', 'Tom' : '555-5555' }
```

Consider the following `search_entry` function to search a name-phone number pair from the above dictionary:

```
def search_entry(self):  
    search_name = input('Enter name to search for: ')  
    if search_name in self.phonebook.keys():  
        print(search_name, ':', self.phonebook[search_name])  
    else:  
        print('entry not found')
```

Rewrite the above function using a try-except block.

```
def search_entry(self):  
    search_name = input('Enter name to search for: ')  
  

```

(3 marks)

Question 25:

[10 marks]

a) Consider the following function:

```
def stack_example(n):  
    my_stack = Stack()  
    while n > 0:  
        my_stack.push(n % 2);  
        n = n // 2;  
    while not my_stack.is_empty():  
        print(my_stack.pop(), end = '')  
    print()
```

What is the output of the following call to the `stack_example` function above? Show the status of the stack after each step.

```
stack_example(12)
```

(5 marks)

- b) Complete the function `reverse_stack()` below which accepts a stack of integers as a parameter and reverses the order of the numbers of the original stack. Use a queue to store the stack's content. Examples are given below:

Code fragment	Output
<pre>my_stack = Stack() my_stack.push('1') my_stack.push('2') my_stack.push('3') while not my_stack.is_empty(): print(my_stack.pop(), end = ' ')</pre>	3 2 1
<pre>my_stack = Stack() my_stack.push('1') my_stack.push('2') my_stack.push('3') reverse_stack(my_stack) while not my_stack.is_empty(): print(my_stack.pop(), end = ' ')</pre>	1 2 3

```
def reverse_stack(my_stack):
    backup = Queue()
```

(5 marks)

Question 26:

[5 marks]

Complete the function `create_list_and_queue` below which takes an integer, `n`, as a parameter and returns a Python List and a Queue as a Tuple. Your function should create a list of `n` natural numbers using list comprehension and create a queue using the elements in the Python List. For example, consider the following code fragment:

```
my_list, my_queue = create_list_and_queue(4)
for num in my_list:
    print(num, end = ' ')
print()
while not my_queue.is_empty():
    print(my_queue.dequeue(), end = ' ')
```

The output is:

```
0 1 2 3
0 1 2 3
```

```
def create_list_and_queue(n):
```

(5 marks)

Question 27:

[7 marks]

a) Consider the following code fragment:

```
my_list = UnorderedList()
my_list.add(12)
my_list.add(3)
my_list.add(7)
my_list.add(2)
```

Draw a diagram of the `UnorderedList` after the above code fragment has been executed.

(3 marks)

b) Complete the following table regarding the operations of Python Lists and LinkedLists.

Operations	Python List & Complexity	Singly Linked List with head and tail references & Complexity
adds an item to the END of the list	_____ Complexity: _____	addToTail (item) Complexity: _____
adds an item to the FRONT of the list	_____ Complexity: _____	addTohead (item) Complexity: _____
removes the item at the END of the list	_____ Complexity: _____	removeFromTail () Complexity: _____
removes the item at the FRONT of the list	_____ Complexity: _____	removeFromHead () Complexity: _____

(4 marks)

Rough Working – This page will not be marked