NATIONAL UNIVERSITY OF SINGAPORE

Department of Computer Science, School of Computing

IT5001 – *Software Development Fundamentals*

Academic Year 2024/2025, Semester 2

Mid-Term Assessment

SOLUTIONS MANUAL

12 March 2025 **Time allowed:** 1 hour

MULTIPLE CHOICE QUESTIONS [75 marks]

1. The following Python expression is entered into a fresh Python shell with no prior import statements. Determine the result from evaluating the following expression. **[3 marks]**

```
print(True or False and True)
```

A. True
B. False
C. None
D. 0
E. Evaluating this expression yields an error

2. The following Python expression is entered into a fresh Python shell with no prior import statements. Determine the result from evaluating the following expression. **[3 marks]**

```
print(3 + 20 / 4 * 12)
```

A. 63.0
B. 63
C. 69.0
D. 69
E. 69.0000019

3. The following Python expression is entered into a fresh Python shell with no prior import statements. Determine the result from evaluating the following expression. **[3 marks]**

```
print(True or 'p'*2 and 1 > 3 or 1 / 0)
```

A. True
B. False
C. None
D. 0
E. Evaluating this expression yields an error

4. The following Python expression is entered into a fresh Python shell with no prior import statements. Determine the result from evaluating the following expression. **[3 marks]**

```
print(len('Kevin and Onion King'))
```

A. `17`
B. `18`
C. `19`
D. `20`
E. `21`

5. The following Python expression is entered into a fresh Python shell with no prior import statements. Determine the result from evaluating the following expression. **[3 marks]**

```
print(('dx'[1] + 'dy'[0]) * 2)
```

A. `'dddd'`
B. `'dydy'`
C. `'xdxd'`
D. `'xyxy'`
E. Evaluating this expression yields an error

6. The following Python expression is entered into a fresh Python shell with no prior import statements. Determine the result from evaluating the following expression. **[3 marks]**

```
print('cucumbernimbus'[-1::-1][:3] + 'marine')
```

A. `'marine'`
B. `'busmarine'`
C. `'cucmarine'`
D. `'minmarine'`
E. `'submarine'`

7. What is the output of the following code? **[3 marks]**

```
x = 15 / 5
if x > 5:
    print (x - 1)
elif x > 4:
    print(x - 2)
else:
    print(x)
```

A. 0.0

B. 1.0

C. 2.0

D. 3.0

E. 4.0

8. What is the output of the following code? **[3 marks]**

```
if True:
    x = 30
    if x < 100:
        x += 2
    elif x == 100:
        x += 1
    else:
        x -= 2
else:
    x = 15
print(x)
```

A. 15

B. 28

C. 32

D. 31

E. Executing this code yields an error

9. What is the output of the following code? **[3 marks]**

```
a, b, c = 'apple', 'bear', 'carrot'
if a and c:
    a = a[:2] + a[-1]
    c = c[:2] + c[-1]
elif not b:
    b = b[1:]
print(a, b, c)
```

A. ape bear cat

B. apple bear cat

C. apple ear cat

D. apple ear carrot

E. Executing this code yields an error

10. What is the output of the following code? **[3 marks]**

```
def fn(t):
    x, y = t**2, t**3
    if x > 1 and y > 1:
        return True
    elif x > 1 or y > 1:
        return x*y > 0
    elif y < 1:
        return False
    return x * y


print(fn(1), fn(0), fn(2))
```

A. 1 True False
B. 1 False True
C. 1.0 True False
D. 1.0 False True
E. 1.0 False False

11. What is the output of the following code? **[3 marks]**

```
x = 10
while x < 100:
    x += x

print(x)
```

A. 160
B. 80
C. 40
D. 20
E. 10

12. What is the output of the following code? **[3 marks]**

```
balance, interest = 5000, 1.6
for _ in range(2):
    balance *= (1 + interest/100)
print(int(balance))
```

A. 5000
B. 5161
C. 5243
D. 5286
E. 5327

13. Which of the following is a possible value of num, such that it would make the code below print out only the first 5 digits? **[3 marks]**

```
num = ___BLANK____
for digit in str(num):
    if int(digit) > 5:
        break
    print(digit)
```

A. 15520164

B. 15520514

C. 15520814

D. 15526814

E. 15560814

14. What is the output of the following code? **[3 marks]**

```
num = 1932102

total = 0
while num != 0:
    digit = num % 10
    num //= 10
    if digit >= 5:
        continue
    if int(digit) % 2 != 0:
        total += int(digit ** 2)
    else:
        total += digit

print(total, sep="\n")
```

A. 9

B. 12

C. 13

D. 15

E. 18

15. The following Python expression is entered into a fresh Python shell with no prior import statements. Determine the result from evaluating the following expression. **[3 marks]**

```
print(len([x for x in range(0, 6, 1)]))
```

A. 4

B. 5

C. 6

D. 7

E. 8

16. The following Python expression is entered into a fresh Python shell with no prior import statements. Determine the result from evaluating the following expression. **[3 marks]**

```
print([i for i in range(1, 10, 2)][::-1][2])
```

A. 9
B. 5
C. 7
D. 3
E. 1

17. What is the output of the following code? **[3 marks]**

```
ali, abu, achan = 3, 7, 4
tom = 10 * 2 / 2 // 1

l_ori = [ali, abu, achan]
l_copy = l_ori

ali = 1
l_ori.append(tom)
print(l_copy)
```

A. [1, 7, 4, 10.0]
B. [3, 7, 4, 10.0]
C. [1, 7, 4, 10]
D. [3, 7, 4, 10]
E. Executing this code yields an error

18. The following Python expression is entered into a fresh Python shell with no prior import statements. Determine the result from evaluating the following expression. **[3 marks]**

```
print(len(((1, 2), 3, (4, 5), (6,))))
```

A. 2
B. 3
C. 4
D. 5
E. 6

19. What is the output of the following code? **[3 marks]**

```
scores = [90, 75, 60]
score_diff = (
    abs(scores[0]-scores[1]),
    abs(scores[2]-scores[0]),
    abs(scores[2]-scores[1]),
)
print(score_diff)
```

A. (15, 30, 15)

B. (15, 30, -15)

C. (15, -30, -15)

D. (-15, -30, -15)

E. (-15, -30, 15)

20. What is the output of the following code? **[3 marks]**

```
inbox = [9, 2, 5, 0, 0, 4, 3, 3, 0, 1, 8, 7, 0]
outbox = tuple()
temp = []

for item in inbox:
    if item != 0:
        temp.append(item)
    elif len(temp) != 0:
        smallest = temp[0]
        for comp in temp[1:]:
            smallest = min(smallest, comp)
            outbox = outbox + (smallest,)
            temp.clear()

print(outbox)
```

A. (1,)

B. (2, 3, 1)

C. (2, 0, 3, 1)

D. (2, 0, 3, 0, 1, 0)

E. Executing this code yields an error

21. The following Python expression is entered into a fresh Python shell with no prior import statements. Determine the result from evaluating the following expression. **[3 marks]**

```
print(set([3, 4, 7, 6, 0, 0, 1]))
```

A. `{[0, 1, 3, 4, 6, 7]}`
B. `{[1, 3, 4, 6, 7]}`
C. `{0, 0, 1, 3, 4, 6, 7}`
D. `{0, 1, 3, 4, 6, 7}`
E. `{1, 3, 4, 6, 7}`

22. The following Python expression is entered into a fresh Python shell with no prior import statements. Determine the result from evaluating the following expression. **[3 marks]**

```
print(len({(1, 3), (3, 5), (5, 7), 9}))
```

A. `3`
B. `4`
C. `6`
D. `7`
E. Evaluating this expression yields an error

23. The following program prints out a set named `result` at the end for each item in list `all`. **[3 marks]**

```
set_a, set_b, set_c = set("one"), set("two"), set("three")
all = [set_a ^ set_b, set_a ^ set_c, set_b ^ set_c]

for this_set in all:
    the_rest = all[:]
    the_rest.remove(this_set)

    for item in the_rest:
        result = this_set & item
        print(result)
```

Out of all the possible values of `result` printed from this program, what is the greatest number of elements possible?

A. 3 elements
B. 2 elements
C. 1 element
D. 0 elements
E. Unable to determine (program yields an error)

24. Which of the following lambda functions is equal to equation A below? **[1 mark]**

$$A = P\left(1 + \frac{r}{100}\right)^n$$

A. `lambda P, r, n: P * 1 + (r / 100)**n`

B. `lambda P, r, n: P * 1 + r / 100**n`

C. `lambda P, r, n: P * (1 + r / 100)**n`

D. `lambda P, r, n: P * ((1 + r) / 100)**n`

E. `lambda P, r, n: P (1 + r / 100)**n`

25. The program below contains variables whose values are defined with lambda functions. **[3 marks]**

```
f1 = (lambda x: x + x)
f2 = (lambda x: int(x ** 3))
f3 = (lambda x: x // 2)
f4 = (lambda x: x * 4)
```

If `num = 5`, which of the following produces the smallest value?

A. `f4(f3(f1(f2(num))))`

B. `f4(f1(f2(f3(num))))`

C. `f3(f2(f1(f4(num))))`

D. `f2(f3(f2(f3(num))))`

E. `f1(f4(f2(f3(num))))`

NOTE: *This was an oopsie – 3 out of 5 options will get you the same smallest value, which is 64.*

FILL-IN-THE-BLANKS QUESTIONS [25 marks]

26. After the user enters an integer `n`, the code is supposed to print out a list of integers from 1 to `n` inclusively. e.g., if the user inputs 10, the code should print:

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

Complete the code below by filling in the blank. **[5 marks]**

```
n = int(input("Please enter a positive number: "))
lst = []
for i in range(___BLANK_1___):
    lst.append(i)
print(lst)
```

Answer:

```
n = int(input("Please enter a positive number: "))
lst = []
for i in range(1,n+1):
    lst.append(i)
print(lst)
```

27. After the user enters an integer n, the code is supposed to print out a list of integers from 1 to n that has at least one digit '8' in it inclusively. e.g., if the user input 30, the code should print:

```
[8, 18, 28]
```

Complete the code below by filling in the blanks. **[5 marks]**

```
n = int(input("Please enter a positive number: "))
lst = []
for i in range(___BLANK_1___):
    if ___BLANK_2___:
        lst.append(i)

print(lst)
```

Answer:

```
n = int(input("Please enter a positive number: "))
lst = []
for i in range(1,n+1):
    if '8' in str(i):
        lst.append(i)

print(lst)
```

28. For an n x n chess board, we can label each square as an xy-coordinate pair. We will use zero-based indexing, namely counting starts with 0 instead of 1. The function `checker_coord(n)` returns a list of all coordinates in the board.

Sample output:

```
>>> print(checker_coord(2))
[(0, 0), (1, 0), (0, 1), (1, 1)]
>>> print(checker_coord(3))
[(0, 0), (1, 0), (2, 0), (0, 1), (1, 1), (2, 1), (0, 2), (1, 2), (2, 2)]
```

Complete the code below by filling in the blanks. The order of each item in the output list must match that in the sample output. **[5 marks]**

```
def checker_coord(n):
    output = []
    for i in range(___BLANK_1___):
        for j in range(___BLANK_2___):
            output.append(___BLANK_3___):
    return output
```

Answer:

```
def checker_coord(n):
    output = []
    for i in range(n):
        for j in range(n):
            output.append((j,i)):
    return output
```

29. The function `is_power_of_two(n)` will return `True` if `n` is a power of 2 (i.e., `n = 2**k` for some `k`). Otherwise, it returns `False`. For example,

```
print(is_power_of_two(1024)) # will print True
print(is_power_of_two(1026)) # will print False
```

Complete the code below by filling in the blank. Your code must be recursive and not use loop structures or list comprehension. **[5 marks]**

```
def is_power_of_two(n):
    if n == 1:
        return True
    return ___BLANK_1___
```

Answer:

```
def is_power_of_two(n):
    if n == 1:
        return True
    return n % 2 == 0 and is_power_of_two(n//2)
```

30. Create a function named `make_function(n)` which will return another function that checks if a number is divisible by `n`.

Sample output:

```
>>> f3 = make_function(3)
>>> print(f3(6))
True
>>> print(f3(31))
False
>>> f7 = make_function(7)
>>> print(f7(17))
False
>>> print(f7(63))
True
```

Complete the code below by filling in the blank. **[5 marks]**

```
def make_function(n):
    return ____BLANK_1____
```

Answer:

```
def make_function(n):
    return lambda x:x%n==0
```

-- End of Solutions Manual --