CSCI2202: Mid-Term Review

Finlay Maguire (finlay.maguire@dal.ca)
TA: Ehsan Baratnezhad (ethan.b@dal.ca)
TA: Precious Osadebamwen (precious.osadebamwen@dal.ca)

Object Types

Q1. What is the type of each of the following expressions?

```
'5'
5.0
5
[]
[5]
[5,10,15]
(5,10)
{5: 10}
{}
lambda x: x +5
```

Mutability/Immutability

Q2. Which of the following types are mutable and which are immutable?

```
Integers (int)
Strings (str)
Floating point numbers (float)
Lists (list)
Tuples (tuple)
Dictionaries (dict)
Boolean (bool)
```

Boolean Statements

Q3. What is the boolean value of the following?

```
"hello" in "hello world"
len([1,2,3]) == 4
3 > 2 > 1
[] == False
bool("False")
0.0 or True
False and False or True
```

Operator Precedence

Q4. What does z contain?

```
z = 2 + 2 * 2 ** 2
```

Aliasing and mutability

Q5. What is the value of x, y, and z after this code finishes?

```python
x = ['a', 'b']
y = x
z = x + ['c']
y.append('c')
```

Operator with different types

Q6. What will this code print?

```python
print(3 * [1, 2, 3])
```

Nested loops

Q7. What does the following code print?

```python
for x in range(3):
    for y in range(2):
        print(x, y)
```

Break/Continue

Q8. What will each of the following output? Why?

```python
for i in range(3):
    for j in range(3):
        if j == 1:
            break
        print(i, j)



for i in range(3):
    for j in range(3):
        if j == 1:
            continue
        print(i, j)
```

Recursion

Q9. What does rec print(3) print?

```python
def rec_print(x):
    print(x, end=' ')
    if x > 1:
        rec_print(x - 1)
        rec_print(x - 1)
```

Keyword and positional arguments

Q10 For this code, what do each of these inputs result in?

```python
def f(x, y=3, z=5):
    print(x + y + z)
f()
f(1)
f(1, 2)
f(1, x=2)
f(1, y=2)
f(1, z=2)
f(1, 2, 3)
f(1, 2, z=3)
f(1, 2, y=3)
```

Lazily zipping tuples

Q11. What does the following code return?

```
list(zip((1, 2), (3, 4)))
```

Comprehensions

Q12. What will this code print?

```python
print(sum([2 * x if x % 2 == 0 else 3 * x for x
in [2, 3]]))
```

Lambda and sorting

Q13. What will `my_list` contain after this code has been run?

```python
my_list = ["abx", "cdk", "ghy"]
my_list.sort(key=lambda x: x[2])
```

Class vs instance attributes

Q14. What does this print?

```python
class A:
    name = "My name"
    def __init__(self, s):
        self.name = s

a = A('Hugo')
print(a.name)
print(A.name)
```

Instance attributes with kwargs

Q15. What does z evaluate to after this code is run?

```python
class C:
    def __init__(self, x=1, y=2):
        self.x = x
        self.y = y
    def sum(self):
        return self.x + self.y

z = C(y=3).sum()
```

Inheritance

Q16. What does this print?

```python
class A:
    def say(self):
        print("Hi there")

class B(A):
    def say(self):
        print("Howdy")
    def __init__(self):
        self.say()
        super().say()

A()
B()
```

Dictionaries and aliasing

Q17. What does `data['z']` contain?

```
data = {'x': [1, 2, 3], 'y': [4, 5, 6]}
data['x'].append(4)
data['z'] = data['x']
data['x'][0] = 10
```

Modifying dictionaries

Q18. What does `my_dict` and `new_dict` contain?

```python
def modify_dict(d):
    d['a'] = 5
    d = {'a': 3, 'b': 4}
    return d

my_dict = {'a': 1, 'b': 2}
new_dict = modify_dict(my_dict)
```

Comprehension and dictionary iteration

Q19. What will this code print?

```python
prices = {'apple': 0.5,
          'banana': 0.3,
          'orange': 0.6}
quantities = {'apple': 5,
              'orange': 3,
              'banana': 2}
print(sum(prices[fruit] * quantities[fruit] for fruit
in prices.keys()))
```

Docstrings

Q20. Write a docstring for this function

```python
def calculate_discount(price, percentage):
    """
    [Write your docstring here]
    """
    if not isinstance(price, (int, float)) or not isinstance(percentage, (int, float)):
        raise TypeError("Price and percentage must be numbers")
    if percentage < 0 or percentage > 100:
        raise ValueError("Percentage must be between 0 and 100")

    discount = price * (percentage / 100)
    return price - discount
```