Week 3
Course: Introduction to programming 2
Theme: Lists, tuples, sets, dictionaries

Task 1. Provide Answers for the following:

1. What will the following code display?

```
numbers = [1, 2, 3, 4, 5]
numbers[2] = 99
print(numbers)
```

2. What will the following code display?

```
numbers = [1, 2, 3, 4, 5]
print(numbers[-2])
```

3. How do you find the number of elements in a list?
4. What will the following code display?

```
numbers1 = [1, 2, 3]
numbers2 = [10, 20, 30]
numbers3 = numbers1 + numbers2
print(numbers1)
print(numbers2)
print(numbers3)
```

5. What will the following code display?

```
numbers1 = [1, 2, 3]
numbers2 = [10, 20, 30]
numbers2 += numbers1
print(numbers1)
print(numbers2)
```

6. What will the following code display?

```
numbers = [1, 2, 3, 4, 5]
my_list = numbers[1:3]
print(my_list)
```

7. What will the following code display?

```
numbers = [1, 2, 3, 4, 5]
my_list = numbers[:]
print(my_list)
```

8. What will the following code display?

```
numbers = [1, 2, 3, 4, 5]
my_list = numbers[-3:]
print(my_list)
```

9. Look at the following interactive session, in which a two-dimensional list is created. How many rows and how many columns are in the list?

```
numbers = [[1, 2], [10, 20], [100, 200], [1000, 2000]]
```

10. Which part of a dictionary element must be immutable?
11. Suppose a dictionary named employee has been created. What does the following statement do?

```
employee['id'] = 54321
```

12. What will the following code display?

```
stuff = {1 : 'aaa', 2 : 'bbb', 3 : 'ccc'}
print(stuff[3])
```

13. How can you determine whether a key-value pair exists in a dictionary?
14. Suppose a dictionary named inventory exists. What does the following statement do?

```
del inventory[654]
```

15. What will the following code display?

```
stuff = {1 : 'aaa', 2 : 'bbb', 3 : 'ccc'}
print(len(stuff))
```

16. What is the difference between the dictionary methods pop and popitem?
17. What does the items method return?
18. What does the keys method return?
19. What does the values method return?
20. What is object serialization?
21. What function do you call to retrieve and unpickle an object?

Task 2. Multiple choice Questions:

1. This term refers to an individual item in a list.
a. element
b. bin
c. cubby hole
d. slot
2. This is a number that identifies an item in a list.
a. element
b. index
c. bookmark
d. identifier
3. This is the first index in a list.
a. - 1
b. 1
c. 0
d. The size of the list minus one
4. This is the last index in a list.
a. 1
b. 99
c. 0
d. The size of the list minus one
5. This will happen if you try to use an index that is out of range for a list.
a. a ValueError exception will occur
b. an IndexError exception will occur
c. The list will be erased and the program will continue to run.
d. Nothing—the invalid index will be ignored
6. This function returns the length of a list.
a. length
b. size
c. len
d. lengthof
7. When the * operator's left operand is a list and its right operand is an integer, the operator becomes this.
a. The multiplication operator
b. The repetition operator
c. The initialization operator
d. Nothing—the operator does not support those types of operands.
8. This list method adds an item to the end of an existing list.
a. add
b. add_to
c. increase
d. append
9. This removes an item at a specific index in a list.
a. The remove method
b. The delete method
c. The del statement
d. The kill method

10. You can add one element to a set with this method.
a. append
b. add
c. update
d. merge
11. You can add a group of elements to a set with this method.
a. append
b. add
c. update
d. merge
12. This set method removes an element but does not raise an exception if the element is not found.
a. remove
b. discard
c. delete
d. erase
13. This set method removes an element and raises an exception if the element is not found.
a. remove
b. discard
c. delete
d. erase
14. This operator can be used to find the union of two sets.
a. |
b. &
c. -
d. ^
15. This operator can be used to find the difference of two sets.
a. |
b. &
c. -
d. ^
16. This operator can be used to find the intersection of two sets.
a. |
b. &
c. -
d. ^
17. This operator can be used to find the symmetric difference of two sets.
a. |
b. &
c. -
d. ^

True or False

1. Lists in Python are immutable.
2. Tuples in Python are immutable.
3. The del statement deletes an item at a specified index in a list.
4. Assume list1 references a list. After the following statement executes, list1 and list2 will reference two identical but separate lists in memory: `list2 = list1`
5. The dictionary method `popitem` does not raise an exception if it is called on an empty dictionary.
6. The following statement creates an empty dictionary:
`mydct = {}`
7. The following statement creates an empty set:
`myset = ()`
8. Sets store their elements in an unordered fashion.

9. You can store duplicate elements in a set.
10. The remove method raises an exception if the specified element is not found in the set.

Short Answer

1. What will the following code display?
```
dct = {'Monday':1, 'Tuesday':2, 'Wednesday':3}
print(dct['Tuesday'])
```
2. What will the following code display?
```
dct = {'Monday':1, 'Tuesday':2, 'Wednesday':3}
print(dct.get('Monday', 'Not found'))
```
3. What will the following code display?
```
dct = {'Monday':1, 'Tuesday':2, 'Wednesday':3}
print(dct.get('Friday', 'Not found'))
```
4. What will the following code display?
```
stuff = {'aaa' : 111, 'bbb' : 222, 'ccc' : 333}
print(stuff['bbb'])
```
5. How do you delete an element from a dictionary?
6. How do you determine the number of elements that are stored in a dictionary?
7. What will the following code display?
```
dct = {1:[0, 1], 2:[2, 3], 3:[4, 5]}
print(dct[3])
```
8. What values will the following code display? (Don't worry about the order in which they will be displayed.)
```
dct = {1:[0, 1], 2:[2, 3], 3:[4, 5]}
for k in dct:
print(k)
```
9. After the following statement executes, what elements will be stored in the myset set?
```
myset = set('Saturn')
```
10. After the following statement executes, what elements will be stored in the myset set?
```
myset = set(10)
```
11. After the following statement executes, what elements will be stored in the myset set?
```
myset = set('a bb ccc dddd')
```
12. After the following statement executes, what elements will be stored in the myset set?
```
myset = set([2, 4, 4, 6, 6, 6, 6])
```
13. After the following statement executes, what elements will be stored in the myset set?
```
myset = set(['a', 'bb', 'ccc', 'dddd'])
```
14. What will the following code display?
```
myset = set('1 2 3')
print(len(myset))
```
15. After the following code executes, what elements will be members of set3?
```
set1 = set([10, 20, 30, 40])
set2 = set([40, 50, 60])
set3 = set1.union(set2)
```
16. After the following code executes, what elements will be members of set3?
```
set1 = set(['o', 'p', 's', 'v'])
set2 = set(['a', 'p', 'r', 's'])
set3 = set1.intersection(set2)
```
17. After the following code executes, what elements will be members of set3?
```
set1 = set(['d', 'e', 'f'])
set2 = set(['a', 'b', 'c', 'd', 'e'])
set3 = set1.difference(set2)
```
18. After the following code executes, what elements will be members of set3?
```
set1 = set(['d', 'e', 'f'])
set2 = set(['a', 'b', 'c', 'd', 'e'])
set3 = set2.difference(set1)
```

19. After the following code executes, what elements will be members of set3?

```
set1 = set([1, 2, 3])
set2 = set([2, 3, 4])
set3 = set1.symmetric_difference(set2)
```

20. Look at the following code:

```
set1 = set([100, 200, 300, 400, 500])
set2 = set([200, 400, 500])
```

Which of the sets is a subset of the other?
Which of the sets is a superset of the other?

Task 3. Programming Exercise

1. When analysing data collected as part of a science experiment it may be desirable to remove the most extreme values before performing other calculations. Write a function that takes a list of values and an non-negative integer, n, as its parameters. The function should create a new copy of the list with the n largest elements and the n smallest elements removed. Then it should return the new copy of the list as the function's only result. The order of the elements in the returned list does not have to match the order of the elements in the original list.

Write a main program that demonstrates your function. Your function should read a list of numbers from the user and remove the two largest and two smallest values from it. Display the list with the outliers removed, followed by the original list. Your program should generate an appropriate error message if the user enters less than 4 values.

2. In this exercise, you will create a program that reads words from the user until the user enters a blank line. After the user enters a blank line your program should display each word entered by the user exactly once. The words should be displayed in the same order that they were entered. For example, if the user enters:

first
second
first
third
second

then your program should display:

first
second
third

3. Create a program that reads integers from the user until a blank line is entered. Once all of the integers have been read your program should display all of the negative numbers, followed by all of the zeros, followed by all of the positive numbers. Within each group the numbers should be displayed in the same order that they were entered by the user. For example, if the user enters the values 3, -4, 1, 0, -1, 0, and -2 then your program should output the values -4, -1, -2, 0, 0, 3, and 1. Your program should display each value on its own line.

4. When writing out a list of items in English, one normally separates the items with commas. In addition, the word "and" is normally included before the last item, unless the list only contains one item. Consider the following four lists:

apples
apples and oranges
apples, oranges and bananas
apples, oranges, bananas and lemons

Write a function that takes a list of strings as its only parameter. Your function should return a string that contains all of the items in the list formatted in the manner described previously as its only result. While the examples shown previously only include lists containing four elements or less, your function should behave correctly for lists of any length. Include a main program that reads several items from the user, formats them by calling your function, and then displays the result returned by the function.

5. In order to win the top prize in a particular lottery, one must match all 6 numbers on his or her ticket to the 6 numbers between 1 and 49 that are drawn by the lottery organizer. Write a program that generates a random selection of 6 numbers for a lottery ticket. Ensure that the 6 numbers selected do not contain any duplicates. Display the numbers in ascending order.

6. Write a function named `reverseLookup` that finds all of the keys in a dictionary that map to a specific value. The function will take the dictionary and the value to search for as its only parameters. It will return a (possibly empty) list of keys from the dictionary that map to the provided value.
Include a main program that demonstrates the `reverseLookup` function as part of your solution to this exercise. Your program should create a dictionary and then show that the `reverseLookup` function works correctly when it returns multiple keys, a single key, and no keys. Ensure that your main program only runs when the file containing your solution to this exercise has not been imported into another program.

7. In this exercise you will simulate 1,000 rolls of two dice. Begin by writing a function that simulates rolling a pair of six-sided dice. Your function will not take any parameters. It will return the total that was rolled on two dice as its only result. Write a main program that uses your function to simulate rolling two six-sided dice 1,000 times. As your program runs, it should count the number of times that each total occurs. Then it should display a table that summarizes this data. Express the frequency for each total as a percentage of the total number of rolls. Your program should also display the percentage expected by probability theory for each total.
Sample output is shown below.

Total	Simulated Percent	Expected Percent
2	2.90	2.78
3	6.90	5.56
4	9.40	8.33
5	11.90	11.11
6	14.20	13.89
7	14.20	16.67
8	15.00	13.89
9	10.50	11.11
10	7.90	8.33
11	4.50	5.56
12	2.60	2.78

8. Create a program that determines and displays the number of unique characters in a string entered by the user. For example, Hello, World! has 10 unique characters while zzz has only one unique character. Use a dictionary or set to solve this problem.

9. Two words are anagrams if they contain all of the same letters, but in a different order. For example, “evil” and “live” are anagrams because each contains one e, one i, one l, and one v. Create a program that reads two strings from the user, determines whether or not they are anagrams, and reports the result.

10. A Bingo card consists of 5 columns of 5 numbers. The columns are labeled with the letters B, I, N, G and O. There are 15 numbers that can appear under each letter. In particular, the numbers that can appear under the B range from 1 to 15, the numbers that can appear under the I range from 16 to 30, the numbers that can appear under the N range from 31 to 45, and so on. Write a function that creates a random Bingo card and stores it in a dictionary. The keys will be the letters B, I, N, G and O. The values will be the lists of five numbers that appear under each letter. Write a second function that displays the Bingo card with the columns labeled appropriately. Use these functions to write a program that displays a random Bingo card. Ensure that the main program only runs when the file containing your solution has not been imported into another program.
*You may be aware that Bingo cards often have a “free” space in the middle of the card. We won’t consider the free space in this exercise.