

# Python for Computer Science and Data Science 1 (CSE 3651)

## MINOR ASSIGNMENT-4: LIST AND TUPLES

---

1. Write a Python program to create a list of size N and store the random values in it and find the sum and average.
2. Define a function **rotate** that receives three arguments and returns a tuple in which the first argument is at index 1, the second argument is at index 2 and the third argument is at index 0. Define variables a, b and c containing 'Doug', 22 and 1984. Then call the function three times. For each call, unpack its result into a, b and c, then display their values.
3. Design and develop a menu-driven Python program for the following list operations.
  - a. Create a list of N integers
  - b. Display the list elements
  - c. Insert an element at a specific position
  - d. Delete an element at a given position
  - e. Exit
4. Write a Python program that removes all occurrences of a specific element from a list.
5. Are there any mistakes in these codes? State the outputs:
  - a) `day, high_temperature = ('Monday', 87, 65)`
  - b) `numbers = [1, 2, 3, 4, 5]`  
`numbers[10]`
  - c) `name = 'amanda'`  
`name[0] = 'A'`
  - d) `numbers = [1, 2, 3, 4, 5]`  
`numbers[3.4]`
  - e) `student_tuple = ('Amanda', 'Blue', [98, 75, 87])`  
`student_tuple[0] = 'Ariana'`
  - f) `('Monday', 87, 65) + 'Tuesday'`
  - g) `'A' += ('B', 'C')`
  - h) `x = 7`  
`del x`  
`print(x)`
  - i) `numbers = [1, 2, 3, 4, 5]`  
`numbers.index(10)`
  - j) `numbers = [1, 2, 3, 4, 5]`  
`numbers.extend(6, 7, 8)`
  - k) `numbers = [1, 2, 3, 4, 5]`  
`numbers.remove(10)`
  - l) `values = []`  
`values.pop()`
6. Input 10 integers from the keyboard into a list. The number to be searched is entered through the keyboard by the user. Write a Python program to find if the number to be searched is present in the list and if it is present, display the number of times it appears in the list.
7. Write a function that takes a list of numbers as input from the user and produces the corresponding cumulative list where each element in the list at index i is the sum of elements at index  $j \leq i$ .

8. Write a function that takes n as an input and creates a list of n lists such that  $i^{th}$  list contains the first five multiples of i.
9. Given the following inputs, indicate in each case (1) to (16), whether the statements will execute successfully. If, so, give what will be the outcome of execution? Also give the output of print statements (where applicable):

```
address = 'B-6, Lodhi road, Delhi'
```

```
list1 = [1, 2, 3]
```

```
list2 = ['a', 1, 'z', 26, 'd', 4]
```

```
tuple1 = ('a', 'e', 'i', 'o', 'u')
```

```
tuple2 = ([2,4,6,8], [3,6,9], [4,8], 5)
```

1. list1[3] = 4
  2. print(list1 \* 2)
  3. print(min(list2))
  4. print(max(list1))
  5. print(list(address))
  6. list2.extend(['e', 5])  
print(list2)
  7. list2.append(['e', 5])  
print(list2)
  8. names = ['rohan', 'mohan', 'gita']  
names.sort(key= len)  
print(names)
  9. list3 = [(x \* 2) for x in range(1, 11)]  
print(list3)
  10. del list3[1:]  
print(list3)
  11. list4 = [ x+y for x in range(1,5) for y in range(1,5)]  
print(list4)
  12. tuple2[3] = 6
  13. tuple2.append(5)
  14. t1 = tuple2 +(5)
  15. ' , ' .join(tuple1)
  16. list(zip(['apple', 'orange'], ('red','orange')))
10. Write a Python function that takes a tuple of tuples and prints the sum of all numeric elements in the inner tuples.
  11. Write a Python program to print M-by-N list in the tabular format.
  12. Define a function that returns the sum of all the elements in a specified column in a matrix. Write a Python program that reads a 3-by-4 matrix and displays the sum of each column. Here is a sample run:  
Enter a 3-by-4 matrix row by row:  
1.5 2 3 4  
5.5 6 7 8  
9.5 1 3 1
  13. Write a Python function that sorts a list of tuples based on the second element of each tuple.

14. Write a Python program that generates a tuple where each element is the square of an integer from 1 to 10.
15. Write a Python program that randomly fills in 0s and 1s into a 4-by-4 matrix, prints the matrix, and finds the first row and column with the most 1s. Here is a sample run of the program:
- ```
0011
0011
1101
1010
The largest row index: 2
The largest column index: 2
```
16. Write a Python program that prompts the user to enter a list and displays whether the list is sorted or not. Here is a sample run. Note that the first number in the input indicates the number of elements in the list. This number is not part of the list. Here is the sample run:
- ```
Enter list: 8 10 1 5 16 61 9 11 1
The list is not sorted
Enter list: 10 1 1 3 4 4 5 7 9 11 21
The list is already sorted
```
17. You can compute the standard deviation with the following formula; you have to store the individual numbers using a list so that they can be used after the mean is obtained.

$$\text{mean} = \frac{\sum_{i=1}^n x_i}{n} = \frac{x_1 + x_2 + x_3 + \cdots + x_n}{n}$$

$$\text{deviation} = \sqrt{\frac{\sum_{i=1}^n (x_i - \text{mean})^2}{n - 1}}$$

Write a Python program that prompts the user to enter ten numbers and displays the mean and standard deviation, as shown in the following sample run:

```
Enter ten numbers: 1.9 2.5 3.7 2 1 6 3 4 5 2
The mean is 3.11
The standard deviation is 1.55738.
```

18. Write a Python program to create a new list that contains the square of every element in a given list using list comprehension.
19. Write a Python function to demonstrate the difference between shallow and deep copy of lists. For Example:-
- ```
Original List: [['Shallow', 2, 3], [4, 5, 6]]
Shallow Copy: [['Shallow', 2, 3], [4, 5, 6]]
Deep Copy: [[1, 2, 3], ['Deep', 5, 6]]
```
20. Twenty students were asked to rate on a scale of 1 to 5 the quality of the food in the student cafeteria, with 1 being “awful” and 5 being “excellent.” Place the 20 responses in a list.
- ```
1, 2, 5, 4, 3, 5, 2, 1, 3, 3, 1, 4, 3, 3, 3, 2, 3, 3, 2, 5
```
- Determine and display the frequency of each rating. Use the built-in (or user-defined) functions and statistics module functions to display the following response statistics: minimum, maximum, range, mean, median, mode, variance and standard deviation.

21. With regard to the following code:

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

```
list(map(lambda x: x ** 2, filter(lambda x: x % 2 != 0, numbers)))
```

a) How many times does the filter operation call its lambda argument?

b) How many times does the map operation call its lambda argument?

c) If you reverse the filter and map operations, how many times does the map operation call its lambda argument?

22. When combining filter and map operations, the order in which they're performed matters. Consider a list numbers containing 10, 3, 7, 1, 9, 4, 2, 8, 5, 6 and the following code:

```
In [1]: numbers = [10, 3, 7, 1, 9, 4, 2, 8, 5, 6]
```

```
In [2]: list(map(lambda x: x * 2,
```

```
...: filter(lambda x: x % 2 == 0, numbers)))
```

```
...:
```

```
Out[3]: [20, 8, 4, 16, 12]
```

Reorder this code to call map first and filter second. What happens and why?

23. Using the list given below

1, 2, 5, 4, 3, 5, 2, 1, 3, 3, 1, 4, 3, 3, 3, 2, 3, 3, 2, 5

Display a bar chart showing the response frequencies and their percentages of the total responses.

24. Given a list of tuples, remove all the tuples with length K, where K is user-defined.