########################## Easy Difficulty ############################

1. Imagine you have the following list in your code (Copy paste it into the code).

Now extract "baba" from this list. (copy the list directly onto your code)

list1= [5, 6, 7.34, [4,[6,"hi"], ["hello", ["abbaba"], False], 23]]

2. Reverse the following list without any built-in method.

Sample List: (copy the list directly onto your code)

[5, -6, 2, 1, 0, 16]

Output:

[16, 0, 1, 2, -6, 5]

3. Take the following input from the user, turn it into a list and

find the maximum, minmum and average of all the numbers of that list.

Sample Input: (take input from user)

[5, -6, 2, 1, 0, 16]

Output:

Maximum: 16

Minimum: -6

Average: 3.0

4. Keep taking inputs from the user until he gives 4 unique odd-positive numbers ranging between 0-12 and find their sum.

Print the list. However, there are 5 unique odd-positive numbers within 0 to 12.

Find and print that one missing odd-positive number.

Sample Input:

2

1

3

4

3

5

7

11

Output:

[1, 3, 5, 7, 11]

Missing: 9

5. Take two lists and create a SET Union (If you do not know how set union or intersection works, google the theory)

Sample Lists: (copy the lists directly onto your code)

L1 = [1, 5, 3]

L2 = [2, 5, 4]

Output:

L12 = [1,5,3,2,4]

Sample Lists: (copy the lists directly onto your code)

L1 = [1, 5]

L2 = [5]

Output:

L12 = [1,5]

6. Take two lists and create a SET Intersection

Sample Lists: (copy the lists directly onto your code)

L1 = [1, 5, 3]

L2 = [2, 5, 4]

Output:

L12 = [5]

Sample Lists: (copy the lists directly onto your code)

L1 = [1, 5]

L2 = [2, 6]

Output:

L12 = [1,5]

7. Write a Python program that takes a list from the user. Then finds the first even and the last odd number in the list.

Sample Input1: (take input from user)

[1, 3, 5, 7, 4, 1, 6, 8]

Sample Output1:

First even: 4 and Last odd: 1

Sample Input2:

[55, 5, 7, 9, 11]

Sample Output2:

First even: not found and Last odd: 11

8. Trace the following code

myList = [10, 20, 30, 40, 50, 40, 30, 20, 10]

p = 5

```
q = 0
r = 0
while q < 9:
    r = (p + q - r) % 9
    while r < q:
        p = p + r
        myList[q] = myList[q] - myList[r] - int(str(p) + "3")
        r += 1
    print(myList[q])
    p = p - q
    q += 25 % 12
print(len(myList) == q)
```

9. Trace the following code

```
l1 = [5, 4, 3, 4, 5, 4, 8, 2, 1]
j = 1

while j <= 3:
    for i in range (1,j+1):
        k= int(str(i)+str(j))%9
        l1[j] = l1[j] - l1[k]
        print(l1[i])
    j+=1
    print(l1[j])
print(j%3!=0)
```

########## Medium Difficulty ##########

10. Merge the following lists to create the resultant list accordingly.

Sample input 1: (copy the lists directly onto your code)

L1 = [1,2,3,4,5,6,7]

L2 = [A,B,C,D]

Output:

L12 = [1,2,A,3,4,B,5,6,C,7,D]

Sample input 2: (copy the lists directly onto your code)

L1 = [1,2,3,4]

L2 = [A,B,C,D]

Output:

L12 = [1,2,A,3,4,B,C,D]

Sample input 3: (copy the lists directly onto your code)

L1 = [1,2,3,4,5,6,7,8]

L2 = [A,B]

Output:

L12 = [1,2,A,3,4,B,5,6,7,8]

11. Assume you have the following two lists. Now add the elements of the same indices of both lists and append them in a new list.

Now print the new list. The length of both lists may not be the same.

Sample Lists #1: (copy the lists directly onto your code)

L1 = [1, 5, 3]

L2 = [2, 5, 4, 6, 2]

Output #1:

[3, 10, 7, 6, 2]

Sample Lists #2: (copy the lists directly onto your code)

L1 = [1, 5, 3, 0, 4]

L2 = [2, 5, 4]

Output #2:

[3, 10, 7, 0, 4]

12. Imagine you have a nested list like the following. You have to print all the numbers within this list.

Sample List: [[1, 2, 3], [6, 7], [0, -1]]

Output:

1

2

3

6

7

0

-1

13. Write a python function that takes a string with multiple numbers separated by space as input from the user.

Extract the numbers from the string and make a list containing the integer values and print it.

Multiply the numbers of the list and print the product. [N.B: You can not use split()]

Sample Input:

1 2 3 4 5

Sample Output:

[1, 2, 3, 4, 5]

Product: 120

14. Write a python code that will take a string as an input from the user where multiple numbers are separated by commas. Make a list of numbers using that string & print it.

Then create and print a new list consisting only of the numbers which are greater than the average of the previous list.

Sample Input 1:

"10, 12, 2, 17, 14"

Sample Output 1:

Input list: [10, 12, 2, 17, 14]

Average of input list: 11.0

New list: [12, 17, 14]

15. Take a fully smaller-cased word from the user and count the number of occurances of each vowel in that word.

Input:

alphanumeric

Output:

a has occurred 2 time(s)

e has occurred 1 time(s)

i has occurred 1 time(s)

o has occurred 0 time(s)

u has occurred 1 time(s)

16. a) What is the difference between append() and insert() function?

b) What is the difference between pop() and remove() function?

c) What is the difference between count() and index() function?

d) What happends when we use find() function?

###################### Pass by Reference ##########################

17. Trace the following code

```
myList = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
b = []
index1 = 0
index2 = 0
b = myList
while (index1 < 10):
  myList[index1] += myList[index2 % 10] + 5
  index2 = 1
  while (index2 < index1):
     myList[index1] = b[index2 % 5] - index1
     index2 += 3
  print(myList[index1])
  index1 += 1
```

18. Trace the following code

```
myList1 = [5,6,1,4,6]
var1 = 2
var3 = myList1
while(var1 < 5):
    var2 = 1
    while(var2 <= var1):
       print(myList1[var2])
       var3[var2] = myList1[var2] + var3[var1] - 9
```

```
        var2 = var2 + 1
    var1 = var1 + 1
print(len(var3) in myList1)
```

19. Trace the following code

```
givenList = [10, 4, 20, 9, 30, 10, 5, 40, 3, 7]
a = 1
b = 5
c = 2
newList = givenList
while(a<10):
  b = (b + a) % 2
  while(b<a):
    givenList[a] = newList[a]- givenList[b]+c
    b = b+1
  print(givenList[a])

  a = a+1
  c += 2
print(10 in givenList)
```