

# LIST

1. **Write a program to find the length of a list in 4 ways.**

2. **Write python program to find the cumulative sum of a list.**

For e.g.

Input: list = [10, 20, 30, 40, 50]

Output: [10, 30, 60, 100, 150]

3. **Write a python program to remove the duplicate elements of a list.**

For e.g.

Input: list = [5, 9, 8, 3, 5, 2, 3, 9, 8, 11]

Output: [5, 9, 8, 3, 2, 11]

4. **Write a python program to remove common elements from two list.**

For e.g.

Input: list1 = [1, 2, 3, 4, 5] & list2 = [2, 4, 6, 8, 10]

Output: list1 = [1, 3, 5] & list2 = [6, 8, 10]

5. **Write a python program to print the first letter of each word.**

For e.g.

Input: list = ["Avul", "Avul Pakir", "Avul Pakir Jainulabdeen", "Avul Pakir JainulabdeenAbdul", "Avul Pakir Jainulabdeen Abdul Kalam"]

Output: ["A","AP","APJ", "APJA", "APJAK"]

6. **Write a Python program that removes and prints every third number from a list of numbers until the list is empty.**

Input : input\_numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9]

Output:

Removed: 3

Removed: 6

Removed: 9

Final List: [1, 2, 4, 5, 7, 8]

7. **Write a Python program to make combinations of 3 digits.**

Input:

input\_digits = [1, 2, 3, 4]

Output: Combinations of 3 Digits:

(1, 2, 3)

(1, 2, 4)  
(1, 3, 4)  
(2, 3, 4)

8. **Write a Python program to check the sum of three elements (each from an array) from three arrays is equal to a target value. Print all those three-element combinations.**
9. **Write a Python program that generates a list of all possible permutations from a given collection of distinct numbers.**
10. **Write a Python program to get all possible two-digit letter combinations from a 1-9 digit string.**
11. **Write a Python program to find the number of notes (Samples of notes: 10, 20, 50, 100, 200, 500) against an amount.**

Range: Number of notes (n):  $n$  ( $1 \leq n \leq 1000000$ ).

12. **Write a Python program to create a sequence where the first four members of the sequence are equal to one. Each successive term of the sequence is equal to the sum of the four previous ones. Find the Nth member of the sequence.**
13. **Write a python program to find the second largest number in a list without using any built-in methods.**  
For e.g.  
Input: list = [78, 45, 56, 33, 97, 23, 99]  
Output: 97
14. **Write a python program to break a list into chunks of size N in Python and find sum.**  
For e.g.  
Input: list = [10, 12, 23, 32, 22, 21, 90, 98, 89, 99] & N = 2  
Output: [[10, 12], [23, 32], [22, 21], [90, 98], [89, 99]] □ [22, 55, 43, 188, 188]
15. **Write a python program to find the sum of length of strings at given indices.**  
For e.g.

Input: list = ["Hi", "I", "am","a","DataScientist"] & index\_list = [0, 1, 4]

Output: 2 + 1 + 13 = 16

**16. Write a Python program to identify unique triplets whose three elements sum to zero from an array of n integers.**

For e.g

Input: input\_numbers = [-1, 0, 1, 2, -1, -4]

Output: Unique Triplets Whose Sum is Zero:

(-1, 0, 1)

(-1, -1, 2)

(-4, 0, 4)

**17. Write a Python program that accepts a positive number and subtracts from it the sum of its digits, and so on. Continue this operation until the number is positive.**

**18. Write a Python program to find the total number of even or odd divisors of a given integer.**

**19. Write a Python program to find the digits that are missing from a given mobile number.**

**20. Write a Python program to compute the summation of the absolute difference of all distinct pairs in a given array (non-decreasing order).**

**21. Write a Python program to find the type of progression (arithmetic progression / geometric progression) and the next successive member of the three successive members of a sequence.**

**22. Write a Python program to find the number of combinations that satisfy  $p + q + r + s = n$  where n is a given number  $\leq 4000$  and p, q, r, s are between 0 to 1000.**

Input a positive integer: (ctrl+d to exit)

252

Number of combinations of a, b, c, d: 2731135

**23. Write a Python program that adds up the columns and rows of the given table as shown in the specified figure.**

Input number of rows/columns (0 to exit)

4

Input cell value:

25 69 51 26

68 35 29 54

54 57 45 63

61 68 47 59

Result:

25 69 51 26 171

8 35 29 54 186

54 57 45 63 219

61 68 47 59 235

208 229 172 202 811

Input number of rows/columns (0 to exit)

**24. Given a list of numbers and a number k, write a Python program to check whether the sum of any two numbers from the list is equal to k or not.**

For example, given [1, 5, 11, 5] and k = 16, return true since 11 + 5 is 16.

Sample Input:

([12, 5, 0, 5], 10)

([20, 20, 4, 5], 40)

([1, -1], 0)

([1, 1, 0], 0)

Sample Output:

True

True

True

False

**25. Write a Python program that removes duplicate elements from a given array of numbers so that each element appears only once and returns the new length of the array.**

Sample Input:

[0,0,1,1,2,2,3,3,4,4,4]

[1, 2, 2, 3, 4, 4]

Sample Output:

5

4

**26. Write a Python program to calculate the maximum profit from selling and buying values of stock. An array of numbers represent the stock prices in chronological order.**

For example, given [8, 10, 7, 5, 7, 15], the function will return 10, since the buying value of the stock is 5 dollars and sell value is 15 dollars.

Sample Input:

([8, 10, 7, 5, 7, 15])

([1, 2, 8, 1])

([])

Sample Output:

10

7

0

**27. Write a Python program to remove all instances of a given value from a given array of integers and find the length of the newly created array.**

Sample Input:

([1, 2, 3, 4, 5, 6, 7, 5], 5)

([10,10,10,10,10], 10)

([10,10,10,10,10], 20)

([], 1)

Sample Output:

6

0

5

0

**28. Write a Python program to find the starting and ending position of a given value in a given array of integers, sorted in ascending order.**

If the target is not found in the array, return [0, 0].

Input: [5, 7, 7, 8, 8, 8] target value = 8

Output: [0, 5]

Input: [1, 3, 6, 9, 13, 14] target value = 4

Output: [0, 0]

**29. Write a Python program to compute the largest product of three integers from a given list of integers.**

Sample Input:

[-10, -20, 20, 1]

[-1, -1, 4, 2, 1]

[1, 2, 3, 4, 5, 6]

Sample Output:

4000

8

120

**30. Write a program to find the multiplication of two matrix.**

```

Ex 1: A  = [[1 3 4]
            [2 5 7]
            [5 9 6]]
      B  = [[1 0 0]
            [0 1 0]
            [0 0 1]]
      A*B = [[1 3 4]
            [2 5 7]
            [5 9 6]]

```

```

Ex 2: A  = [[1 2]
            [3 4]]
      B  = [[1 2 3 4 5]
            [5 6 7 8 9]]
      A*B = [[11 14 17 20 23]
            [23 30 36 42 51]]

```

```

Ex 3: A  = [[1 2]
            [3 4]]
      B  = [[1 4]
            [5 6]
            [7 8]
            [9 6]]
      A*B =Not possible

```

**31. Write a python program to find the sum of digits till single digit.**

For e.g.

Input: list = [121, 220, 12330, 78940, 235650]

Output: [4, 4, 9, 21] □ [4, 4, 9, 3]

**32. Write a Python program to print the length of the series and the series from the given 3rd term, 3rd last term, and the sum of a series.**

Sample Data:

Input third term of the series: 3

Input 3rd last term: 3

Input Sum of the series: 15

Length of the series: 5

Series:

1 2 3 4 5

**33. Write a Python program to find common divisors between two numbers in a given pair.**

**34. Write a Python program to find the maximum sum of a contiguous subsequence from a given sequence of numbers  $a_1, a_2, a_3, \dots, a_n$ . A subsequence of one element is also a continuous subsequence.**

Input:

You can assume that  $1 \leq n \leq 5000$  and  $-100000 \leq a_i \leq 100000$ .

Input numbers are separated by a space.

Input 0 to exit.

Input number of sequence of numbers you want to input (0 to exit):

3

Input numbers:

2

4

6

Maximum sum of the said contiguous subsequence: 12

Input number of sequence of numbers you want to input (0 to exit):

0

**35. Write a Python program that reads  $n$  digits (given) chosen from 0 to 9 and prints the number of combinations where the sum of the digits equals another given number (s). Do not use the same digits in a combination.**

Input:

Two integers as number of combinations and their sum by a single space in a line.

Input 0 0 to exit.

Input number of combinations and sum, input 0 0 to exit:

5 6

2 4

0 0

2

**36. Write a Python program that determines the difference between the largest and smallest integers created by 8 numbers from 0 to 9. The number that can be rearranged shall start with 0 as in 00135668.**

Input:

Input an integer created by 8 numbers from 0 to 9.

2345

Difference between the largest and the smallest integer from the given integer:

3087

**37. Write a Python program to compute the sum of the first  $n$  prime numbers.**

Input:

$n$  ( $n \leq 10000$ ). Input 0 to exit the program.

Input a number ( $n \leq 10000$ ) to compute the sum :(0 to exit)

25

Sum of first 25 prime numbers:

1060

**38. Arrange integers (0 to 99) as narrow hilltop, as illustrated in Figure 1. Reading such data representing huge, when starting from the top and proceeding according to the next rule to the bottom. Write a Python program that compute the maximum value of the sum of the passing integers.**

Input:

A series of integers separated by commas are given in diamonds. No spaces are included in each line. The input example corresponds to Figure 1. The number of lines of data is less than 100 lines.

Output:

The maximum value of the sum of integers passing according to the rule on one line.

Input the numbers (ctrl+d to exit):

8

4, 9

9, 2, 1

3, 8, 5, 5

5, 6, 3, 7, 6

3, 8, 5, 5

9, 2, 1

4, 9

8

Maximum value of the sum of integers passing according to the rule on one line.

64

**39 .From Wikipedia, the free encyclopedia:**

**A happy number is defined by the following process:**

**Starting with any positive integer, replace the number by the sum of the squares of its digits, and repeat the process until the number equals 1 (where it will stay), or it loops endlessly in a cycle which does not include 1. Those numbers for which this process ends in 1 are happy numbers, while those that do not end in 1 are unhappy numbers.**

**Write a Python program to check whether a number is "happy" or not.**

Sample Input:

(7)

(932)

(6)

Sample Output:

True

True

False



**40. from Wikipedia,**

**A happy number is defined by the following process:**

**Starting with any positive integer, replace the number by the sum of the squares of its digits, and repeat the process until the number equals 1 (where it will stay), or it loops endlessly in a cycle which does not include 1. Those numbers for which this process ends in 1 are happy numbers, while those that do not end in 1 are unhappy numbers.**

**Write a Python program to find and print the first 10 happy numbers.**

Sample Input:

`[ :10]`

Sample Output:

`[1, 7, 10, 13, 19, 23, 28, 31, 32, 44]`

**41. Write a Python program that counts the number of prime numbers that are less than a given non-negative number.**

Sample Input:

`(10)`

`(100)`

Sample Output:

`4`

`25`

**42. The price of a given stock on each day is stored in an array.**

**Write a Python program to find the maximum profit in one transaction i.e., buy one and sell one share of the stock from the given price value of the said array. You cannot sell a stock before you buy one.**

Input (Stock price of each day): `[224, 236, 247, 258, 259, 225]`

Output: `35`

Explanation:

`236 - 224 = 12`

`247 - 224 = 23`

`258 - 224 = 34`

`259 - 224 = 35`

`225 - 224 = 1`

`247 - 236 = 11`

`258 - 236 = 22`

`259 - 236 = 23`

`225 - 236 = -11`

`258 - 247 = 11`

`259 - 247 = 12`

225 - 247 = -22  
259 - 258 = 1  
225 - 258 = -33  
225 - 259 = -34

**43. Write a Python program to print a given N by M matrix of numbers line by line in forward > backwards > forward >... order.**

Input matrix:

[[1, 2, 3, 4],  
[5, 6, 7, 8],  
[0, 6, 2, 8],  
[2, 3, 0, 2]]

Output:

1  
2  
3  
4  
8  
7  
6  
5  
0  
6  
2  
8  
2  
0  
3  
2