Introduction to python

Problem 1: Min of Three (you can compare only two at a time.)

Problem 2: If we list all the natural numbers below 10 that are multiples of 3 or 5, we get 3, 5, 6 and 9. The sum of these multiples is 23. Find the sum of all the multiples of 3 or 5 below 1000.

Problem 3: Calculate RD maturity amount

Problem 4: Fibonacci Number upto N

Problem 5: First N Fibinacci Number

Conditional Statements

Problem 6: Find the integral square root

Problem 7: Each new term in the Fibonacci sequence is generated by adding the previous two terms. By starting with 1 and 2, the first 10 terms will be :
1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ...

By considering the terms in the Fibonacci sequence whose values do not exceed four million, find the sum of the even-valued terms.

Problem 8: The prime factors of 13195 are 5, 7, 13 and 29.

Problem 8.1: What is the largest prime factor of the number 600851475143 ?

Problem 9: Check If given number is prime or not.

Problem 10: Print first N prime numbers.

List and Tuples

Problem 11: Define a 2D list and then find max element in it.

Problem 12: Write a program to find the frequency of vowels in given string.

Problem 13: Write a 1 line code to initialize a matrix of size n x n with all zero values.

Problem 14: Write a program to generate a new list using all the even numbers from input list.

Problem 15: Write a program to find even-parity of given no.

Problem 16: Write program to do right rotation of list by n times.

Problem 17: Given a non-negative number represented as a list of digits. Write a program to add one to it.

Problem 18:

Take two lists, say for example these two:

```
a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]
b = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]
```

and write a program that returns a list that contains only the elements that are common between the lists (without duplicates). Make sure your program works on two lists of different sizes.

Problem 19:

Given an unsorted list of integers. Find a pair with a given sum in it.
For **Example:**

```
Input :
arr = {8, 7, 2, 5, 3, 1}
sum = 10

Output :
Pair found at index 0 and 2 (8 + 2)
Pair found at index 1 and 4 (7 + 3)
```

Dictionary

Problem 20: Write a program to count the frequency of each character in given string.

Problem 21: With a given integral number n, write a program to generate a dictionary that contains (i, i*i) such that is an integral number between 1 and n (both included). and then the program should print the dictionary. Suppose the following input is supplied to the program:8Then, the output should be:{1: 1, 2: 4, 3: 9, 4: 16, 5: 25, 6: 36, 7: 49, 8: 64}

Problem 22: You have list of coordinates (x,y) eg. [(3,5),(6,7),(2,9),(-3,6),(9,1)] And your input is (a,b). Write a program to find out closest point in the list with (a,b).

Sets

Problem 23: Write one line python code to remove redundant elements from given list.

Problem 24: Find all the number less than n which are not multiple 7, 11 and 13.
Problem 25: If P = {multiples of 3 between 1 and 20} and Q = {even natural numbers up to 15}.

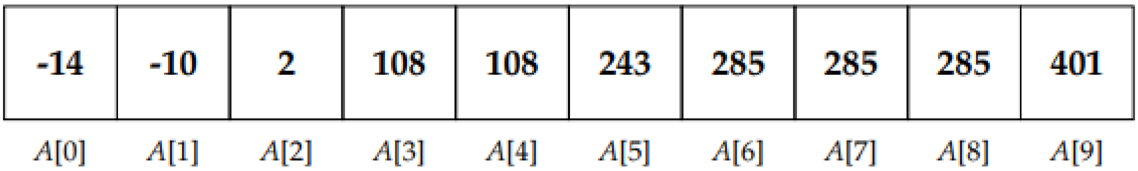
Problem 26: What are the elements which are present in both?

Problem 27: What are the elements which are present in P but not in Q?

Advanced:

Problem 1: Search a sorted array for the first occurrence of k

Binary search commonly asks for the index of any element of a sorted array A that is equal to a given element. The following problem has a slight twist on this.

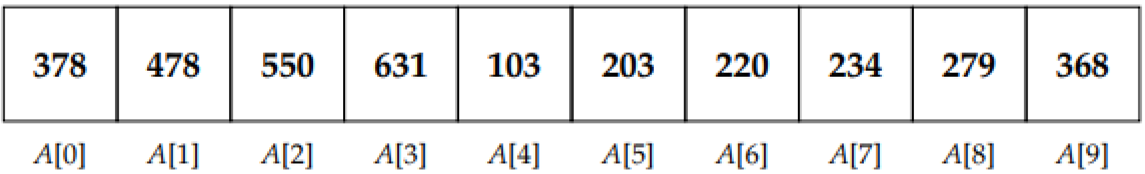


-14	-10	2	108	108	243	285	285	285	401
$A[0]$	$A[1]$	$A[2]$	$A[3]$	$A[4]$	$A[5]$	$A[6]$	$A[7]$	$A[8]$	$A[9]$

Write Python code that takes a sorted list A and a value k and prints the index of the first occurrence of k in A. Print−1 if k does not appear in A. For example, when applied to the list in the Figure your algorithm should print 3 if k = 108; if k = 285, your algorithm should print 6.

Problem 2: Search a cyclically sorted array

An array A of length n is said to be cyclically sorted if the smallest element in the array is at index i, and the sequence hA[i], A[i+1], . . . , A[n−1], A[0], A[1], . . . , A[i−1]i is sorted in increasing order as illustrated in the Figure below:



378	478	550	631	103	203	220	234	279	368
$A[0]$	$A[1]$	$A[2]$	$A[3]$	$A[4]$	$A[5]$	$A[6]$	$A[7]$	$A[8]$	$A[9]$

Design an O(log n) algorithm for finding the position of the smallest element in a cyclically sorted array. Assume all elements are distinct. For example, for the array in the Figure, your algorithm should return 4.

Problem 3: Find the missing IP address

The storage capacity of hard drives dwarfs that of RAM. This can lead to interesting space-time trade-offs.

Suppose you were given a file containing roughly one billion Internet Protocol (IP) addresses, each of which is a 32-bit unsigned integer. How would you programmatically find an IP address that is not in the file? Assume you have unlimited drive space but only two megabytes of RAM at your disposal.

Loops

Problem 28: Write a python program to print all the divisors of ***n***.

Hint:

1) You need % operator

2) You need conditional statement

3) You need nested loop

Example:

i) if ***n*** = 21, output = 1, 3 , 7, 21

ii) if ***n*** = 18, output= 1, 2, 3, 6, 9, 18

Problem 29: Write a python program to print a sequence same as range function. You will have three inputs **start**, **end** and **step**. These input variable have same meaning as range() function.
Note: Do not use range() function, you have to develop your own
Example:
i) if ***start*** = 2, **end** = 8, **step** = 2, Output = 2, 4, 6
ii) if ***start*** = 10, **end** = 101, **step** = 20, Output = 10, 30, 50, 70, 90
iii)
Hint:
1) Use of conditional statements
2) Use of comparison operator and logical operators.
3) Use of Single loop (While)

Problem 30: Write a python program to find the product of all the numbers in a sequence generated by range() function.

Example:
i) if ***start*** = 2, **end** = 8, **step** = 2, Output = 48 = (2 x 4 x 6)
iii) if ***start*** = 10, **end** = 101, **step** = 20, then output= 94500000 (10 x 30 x 50 x 70 x 90)

Hint:

1) Use of range function

2) Use of single loop

Functions

Problem 31:
Can you sort a list of tuples?
Each tuple in the list holds marks of 5 subject. Arrange each tuple based on the highest to lowest percentage.

Problem 32:
Write a function which will accept only two string parameters and you have to find the common characters in both the words.

- Prompt if both the argument are not strings
- Prompt if any of the string input has any other character than [a-zA-Z]

Problem 33:
Write encryption and decryption function for Caesar Cipher.

Problem 34:
Write a program to break the encryption of a given encrypted file. The objective is to get the decrypted content back without the use of the key. Please, find the attached file which needs to be decrypted.

Advanced

Problem 1: Rearrange positive and negative numbers

A list contains both positive and negative numbers in random order. Rearrange the array elements so that positive and negative numbers are placed alternatively. The number of positive and negative numbers need not be equal. If there are more positive numbers they appear at the end of the array. If there are more negative numbers, they too appear at the end of the array.

For example, if the input array is [-1, 2, -3, 4, 5, 6, -7, 8, 9], then the output should be [9, -7, 8, -3, 5, -1, 2, 4, 6]

Problem 2: Maximum Occurrence in a Given Range

Given an array of n integers in non-decreasing order. Find the number of occurrences of the most frequent value within a given range.

Input : arr[] = {-5, -5, 2, 2, 2, 2, 3, 7, 7, 7}
    Query 1: start = 0, end = 9
    Query 2: start = 4, end = 9
Output : 4
    3
Explanation:
Query 1: '2' occurred the most number of times
with a frequency of 4 within given range.
Query 2: '7' occurred the most number of times
with a frequency of 3 within given range.

Problem 3: K'th Smallest Element in Unsorted List

Given a list and a number k where k is smaller than size of list. we need to find the k'th smallest element in the given array. It is given that ll array elements are distinct

Input: arr[] = {7, 10, 4, 3, 20, 15}
    k = 3
Output: 7
Input: arr[] = {7, 10, 4, 3, 20, 15}
    k = 4
Output: 10

Problem 4: Find all elements in list which have at-least two greater elements

Given a list of n distinct elements, the task is to find all elements in list which have at-least two greater elements than themselves.

Input : arr[] = {2, 8, 7, 1, 5};
Output : 2 1 5

The output three elements have two or
more greater elements

Input : arr[] = {7, -2, 3, 4, 9, -1};
Output : -2 3 4 -1

Modules

Problem 35:

Create a module with name "support" and define three functions in it. Now in the main script try the following activities:

- Import all the functions in the main script and call each of them.
- Import only one of the functions in the current namespace of main script and call
- that function.
- Import all the functions in the current namespace of main script and call all of them.

Problem 36:

Create a module with name "support", define three functions in it also add some run able

code (not definitions) and prevent the run able code from getting executed during the

import operation.

Problem 37:

Create a package "support" and inside it, create three modules "support1","support2" and

"support3". Then, define two functions in each module. Now, do the following operations:

- Import only support1 and call all the imported functions
- Import only support2 and support3 and call the all the functions.
- Import only one function from support1 to the current name space.
- d. Create a sub-package "extra" with module "extra1" inside “support” package and the module "extra1" should have two functions inside it.
  - Import all the function defined in the "extra1" to the main script.
  - Import only one function in the "extra1" to the main script

Advanced

Problem 1: Activity Selection Problem

You are given n activities with their start and finish times. Select the maximum number of activities that can be performed by a single person, assuming that a person can only work on a single activity at a time.

Example 1 : Consider the following 3 activities sorted by
by finish time.
    start[] = {10, 12, 20};
    finish[] = {20, 25, 30};
A person can perform at most two activities. The
maximum set of activities that can be executed
is {0, 2} [These are indexes in start[] and
finish[]]

Example 2 : Consider the following 6 activities
sorted by by finish time.
    start[] = {1, 3, 0, 5, 8, 5};
    finish[] = {2, 4, 6, 7, 9, 9};
A person can perform at most four activities. The
maximum set of activities that can be executed
is {0, 1, 3, 4} [These are indexes in start[] and
finish[]]

Problem 2: Job Sequencing Problem

Given an array of jobs where every job has a deadline and associated profit if the job is finished before the deadline. It is also given that every job takes a single unit of time, so the minimum possible deadline for any job is 1. How to maximize total profit if only one job can be scheduled at a time.

Input: Four Jobs with following deadlines and profits

JobID	Deadline	Profit
a	4	20
b	1	10
c	1	40
d	1	30

Output: Following is maximum profit sequence of jobs
    c, a

Input: Five Jobs with following deadlines and profits

JobID	Deadline	Profit
a	2	100
b	1	19
c	2	27
d	1	25
e	3	15

Output: Following is maximum profit sequence of jobs
    c, a, e

Problem 3: Egyptian Fraction

Every positive fraction can be represented as sum of unique unit fractions. A fraction is unit fraction if numerator is 1 and denominator is a positive integer, for example 1/3 is a unit fraction. Such a representation is called Egyptian Fraction as it was used by ancient Egyptians.

Egyptian Fraction Representation of 2/3 is 1/2 + 1/6
Egyptian Fraction Representation of 6/14 is 1/3 + 1/11 + 1/231
Egyptian Fraction Representation of 12/13 is 1/2 + 1/3 + 1/12 + 1/156

Input/Output

Problem 38: Write a program to copy a file from one directory to another.

Problem 39: Write a program to convert all the alphabets in a file to upper case.

Problem 40: Write a program to identify the common words in two given files.

Problem 41: Write a program to split a file into n equal parts.

Problem 42: Implement run-length encoding compression algorithm (https://en.wikipedia.org/wiki/Run-length_encoding)
to compress a given file. Also, write code to decompress the same.

OOPS

Problem 43: Write a program to count the number of objects created for a given class.

Problem 44: Define a class called “Student” which can hold the name, age and marks of 3 subjects and also add functionality to calculate the percentage of all the marks.

Regular Expression

Problem 45: Write a regular expression for python identifier.

Problem 46: Write a regular expression to validate a mobile number with the format (+99-9999999999), where international code can be of any length but the mobile number must be of 10 digits.

Problem 47: Write a regular expression to validate an email address.