

CS23336-Introduction to Python Programming

Started on	Monday, 11 November 2024, 9:46 PM
State	Finished
Completed on	Monday, 11 November 2024, 11:20 PM
Time taken	1 hour 34 mins
Marks	10.00/10.00
Grade	100.00 out of 100.00

Question 1

Correct
Mark 1.00 out of 1.00
Flag question

Question text

String should contain only the words are not palindrome.

Sample Input 1

Malayalam is my mother tongue

Sample Output 1

is my mother tongue
Answer:(penalty regime: 0 %)

```
1 def isPalindrome(word):
2     i=0
3     j=len(word)-1
4     while i<j:
5         if word[i]!=word[j]:
6             return False
7         i+=1
8         j-=1
9     return True
10 words=input().lower().split(" ")
11 for word in words:
12     if not isPalindrome(word):
13         print(word,end=" ")
14
15
16
```

Feedback

Input	Expected	Got
Malayalam is my mother tongue	is my mother tongue	is my mother tongue

Passed all tests!
Correct
Marks for this submission: 1.00/1.00.

Question 2

Correct
Mark 1.00 out of 1.00
Flag question

Question text

Two string values S1, S2 are passed as the input. The program must print first N characters present in S1 which are also present in S2.

Input Format:

The first line contains S1.
The second line contains S2.
The third line contains N.

Output Format:

The first line contains the N characters present in S1 which are also present in S2.

Boundary Conditions:

2 <= N <= 10
2 <= Length of S1, S2 <= 1000

Example Input/Output 1:

Input:

abcbde
cdefghbb
3

Output:

bcd

Note:

b occurs twice in common but must be printed only once.

Answer:(penalty regime: 0 %)

```
1 def fun(s1,s2,n):
2     res=[]
3     seen=set()
4     for char in s1:
5         if char in s2 and char not in seen:
6             res.append(char)
7             seen.add(char)
8         if len(res)==n:
9             break
10    return ''.join(res)
11    s1=input()
12    s2=input()
13    n=int(input())
14    print(fun(s1,s2,n))
```

Feedback

Input	Expected Got
abcbde cdefghbb 3	bcd

Passed all tests!
Correct
Marks for this submission: 1.00/1.00.

Question 3

Correct
Mark 1.00 out of 1.00
Flag question

Question text

Balanced strings are those that have an equal quantity of 'L' and 'R' characters.

Given a balanced string *s*, split it in the maximum amount of balanced strings.

Return the maximum amount of split balanced strings.

Example 1:

Input:

RLRRLRLRL

Output:

4

Explanation: *s* can be split into "RL", "RRLL", "RL", "RL", each substring contains same number of 'L' and 'R'.

Example 2:

Input:

RLLLLRRRLR

Output:

3

Explanation: *s* can be split into "RL", "LLLR", "RR", each substring contains same number of 'L' and 'R'.

Example 3:

Input:

LLLLRRRR

Output:

1

Explanation: *s* can be split into "LLLLRRRR".

Constraints:

$1 \leq s.length \leq 1000$

s[*i*] is either 'L' or 'R'.

s is a balanced string.

For example:

Test	Result
<code>print(BalancedStrings('RLRRLRLRL'))</code>	4
<code>print(BalancedStrings('RLLLLRRRLR'))</code>	3

Answer:(penalty regime: 0 %)

Reset answer

```
1 def BalancedStrings(s):
2     b=0
3     c=0
4     for char in s:
5         if char=='L':
6             b+=1
7         else:
8             b-=1
9         if b==0:
10            c+=1
11     return c
12
```

Feedback

Test	Expected	Got
print(BalancedStrings('RLRRLLRLRL'))	4	4
print(BalancedStrings('RLLLLRRRLR'))	3	3

Passed all tests!

Correct
Marks for this submission: 1.00/1.00.

Question 4

Correct
Mark 1.00 out of 1.00
Flag question

Question text

Given two Strings s1 and s2, remove all the characters from s1 which is present in s2.

Constraints

1<= string length <= 200

Sample Input 1

experience
enc

Sample Output 1

xpri
Answer:(penalty regime: 0 %)

```
1 s1=input()
2 s2=input()
3 s2=set(s2)
4 for i in s1:
5     if i not in s2:
6         print(i,end='')
7
```

Feedback

Input	Expected	Got
experience enc	xpri	xpri

Passed all tests!

Correct
Marks for this submission: 1.00/1.00.

Question 5

Correct
Mark 1.00 out of 1.00
Flag question

Question text

Write a Python program for binary search.

For example:

Input	Result
1, 2, 3, 5, 8 6	False
3, 5, 9, 45, 42 42	True

Answer:(penalty regime: 0 %)

```
1 def search(arr,t):
2     arr.sort()
3     l,r=0,len(arr)-1
4     while l<=r:
5         m=(l+r)//2
6         if arr[m]==t:
7             return True
8         elif arr[m]<t:
9             l=m+1
10        else:
11            r=m-1
12        return False
13 arr=list(map(int,input().split(',')))
14 t=int(input())
15 print(search(arr,t))
```

Feedback

Input	Expected	Got
1, 2, 3, 5, 8 6	False	False
3, 5, 9, 45, 42 42	True	True
52, 45, 89, 43, 11 11	True	True

Passed all tests!
Correct
Marks for this submission: 1.00/1.00.

Question 6

Correct
Mark 1.00 out of 1.00
Flag question

Question text

Given an array of integers nums which is sorted in ascending order, and an integer target, write a function to search target in nums. If target exists, then return its index. Otherwise, return -1.
You must write an algorithm with $O(\log n)$ runtime complexity.

Example 1:
Input: nums = [-1,0,3,5,9,12], target = 9
Output: 4
Explanation: 9 exists in nums and its index is 4

Example 2:

Input: nums = [-1,0,3,5,9,12], target = 2

Output: -1

Explanation: 2 does not exist in nums so return -1

Constraints:

- $1 \leq \text{nums.length} \leq 10^4$
- $-10^4 < \text{nums}[i], \text{target} < 10^4$
- All the integers in nums are **unique**.
- nums is sorted in ascending order.

For example:

Test

Result

print(search([-1,0,3,5,9,12],9)) 4

Answer:(penalty regime: 0 %)

Reset answer

```
1 def search(nums, target):
2     l, r = 0, len(nums) - 1
3     while l <= r:
4         m = l + (r - l) // 2
5         if nums[m] == target:
6             return m
7         elif nums[m] < target:
8             l = m + 1
9         else:
10            r = m - 1
11    return -1
12
```

Feedback

Test

Expected Got

print(search([-1,0,3,5,9,12],9)) 4 4

print(search([-1,0,3,5,9,12],2)) -1 -1

Passed all tests!

Correct

Marks for this submission: 1.00/1.00.

Question 7

Correct

Mark 1.00 out of 1.00

Flag question

Question text

Given an list, find peak element in it. A peak element is an element that is greater than its neighbors.

An element $a[i]$ is a peak element if

$A[i-1] \leq A[i] \geq A[i+1]$ for middle elements. $[0 < i < n-1]$

$A[i-1] \leq A[i]$ for last element $[i=n-1]$

$A[i] \geq A[i+1]$ for first element $[i=0]$

Input Format

The first line contains a single integer n , the length of A .

The second line contains n space-separated integers, $A[i]$.

Output Format

Print peak numbers separated by space.

Sample Input

5
8 9 10 2 6

Sample Output

10 6

For example:

Input	Result
4 12 3 6 8	12 8

Answer:(penalty regime: 0 %)

```
1 def find(n,arr):
2     peaks=[]
3     for i in range(n):
4         if i==0:
5             if n==1 or arr[i]>=arr[i+1]:
6                 peaks.append(arr[i])
7         elif i==n-1:
8             if arr[i]>=arr[i-1]:
9                 peaks.append(arr[i])
10        else:
11            if arr[i]>arr[i-1] and arr[i]>=arr[i+1]:
12                peaks.append(arr[i])
13    return peaks
14 n=int(input())
15 arr=list(map(int,input().split()))
16 peaks=find(n,arr)
17 peaks[0]=peaks[0][0]
18 print(" ".join(map(str,peaks)))
```

Feedback

Input	Expected	Got
7 15 7 10 8 9 4 6	15 10 9 6 15	10 9 6
4 12 3 6 8	12 8	12 8

Passed all tests!
Correct
Marks for this submission: 1.00/1.00.

Question 8

Correct
Mark 1.00 out of 1.00
Flag question

Question text

An list contains N numbers and you want to determine whether two of the numbers sum to a given number K. For example, if the input is 8, 4, 1, 6 and K is 10, the answer is yes (4 and 6). A number may be used twice.

Input Format

The first line contains a single integer n , the length of list
The second line contains n space-separated integers, list[i].
The third line contains integer k.

Output Format

Print Yes or No.

Sample Input

7
0 1 2 4 6 5 3

Sample Output

Yes

For example:

Input	Result
5 8 9 12 15 3 11	Yes
6 2 9 21 32 43 43 1 4	No

Answer:(penalty regime: 0 %)

```
1 def fun(n,arr,k):
2     seen=set()
3     for num in arr:
4         if (k-num) in seen:
5             return "Yes"
6         seen.add(num)
7     return "No"
8 n=int(input())
9 arr=list(map(int,input().split()))
10 k=int(input())
11 print(fun(n,arr,k))
```

Feedback

Input	Expected	Got
5 8 9 12 15 3 11	Yes	Yes
6 2 9 21 32 43 43 1 4	No	No
6 13 42 31 4 8 9 17	Yes	Yes

Passed all tests!

Correct
Marks for this submission: 1.00/1.00.

Question 9

Correct
Mark 1.00 out of 1.00
Flag question

Question text

You are given an $m \times n$ integer matrix `matrix` with the following two properties:

- Each row is sorted in non-decreasing order.
- The first integer of each row is greater than the last integer of the previous row.

Given an integer `target`, return `True` *if target is in matrix* or `False` *otherwise*.

You must write a solution in $O(\log(m * n))$ time complexity.

Example 1:

Input: matrix = [[1,3,5,7],[10,11,16,20],[23,30,34,60]], target = 3
Output: True

Example 2:

Input: matrix = [[1,3,5,7],[10,11,16,20],[23,30,34,60]], target = 13
Output: False

For example:

Test	Result
print(searchMatrix([[1,3,5,7],[10,11,16,20],[23,30,34,60]], 13))	False
print(searchMatrix([[1,3,5,7],[10,11,16,20],[23,30,34,60]], 3))	True

Answer:(penalty regime: 0 %)

Reset answer

```
1 def searchMatrix(matrix, target):
2     for i in range(3):
3         for j in range(4):
4             if matrix[i][j]==target:
5                 return True
6     else:
7         return False
8
9
```

Feedback

Test	Expected	Got
print(searchMatrix([[1,3,5,7],[10,11,16,20],[23,30,34,60]], 13))	False	False
print(searchMatrix([[1,3,5,7],[10,11,16,20],[23,30,34,60]], 3))	True	True

Passed all tests!
Correct
Marks for this submission: 1.00/1.00.

Question 10

Correct
Mark 1.00 out of 1.00
Flag question

Question text

Given an array nums containing n distinct numbers in the range [0, n], return the only number in the range that is missing from the array.

Example 1:

Input: nums = [3,0,1]
Output: 2
Explanation: n = 3 since there are 3 numbers, so all numbers are in the range [0,3]. 2 is the missing number in the range since it does not appear in nums.

Example 2:

Input: nums = [0,1]
Output: 2
Explanation: n = 2 since there are 2 numbers, so all numbers are in the range [0,2]. 2 is the missing number in the range since it does not appear in nums.

Example 3:

Input: nums = [9,6,4,2,3,5,7,0,1]

Output: 8

Explanation: n = 9 since there are 9 numbers, so all numbers are in the range [0,9]. 8 is the missing number in the range since it does not appear in nums.

For example:

Test	Result
print(missingNumber([3,0,1]))	2
print(missingNumber([0,1]))	2

Answer:(penalty regime: 0 %)

Reset answer

```
1 def missingNumber(numbers):
2     l=[]
3     num=len(numbers)+1
4     for i in range(num):
5         if i not in numbers:
6             l.append(i)
7     print(*l)
8     return ''
```

Feedback

Test	Expected	Got
print(missingNumber([3,0,1]))	2	2
print(missingNumber([0,1]))	2	2
print(missingNumber([9,6,4,2,3,5,7,0,1]))	8	8

Passed all tests!
Correct
Marks for this submission: 1.00/1.00.

Finish review
[Skip Quiz navigation](#)

Quiz navigation

[Question 1 This page](#) [Question 2 This page](#) [Question 3 This page](#) [Question 4 This page](#) [Question 5 This page](#) [Question 6 This page](#) [Question 7 This page](#) [Question 8 This page](#) [Question 9 This page](#) [Question 10 This page](#)
[Show one page at a time](#) Finish review