.

**Department of Computer ScienceandEngineering**|**Rajalakshmi EngineeringCollege**

89



**05-ListinPython**

**Ex.No. : 5.1 Date:**



**Register No: 231401021 Name:DEVADHARSHINI K S**

# BalancedArray

Assumethatthegivenstringhasenoughmemory. Don't use any extra space(IN-PLACE)

## SampleInput1

a2b4c6

## SampleOutput1

aabbbbcccccc

# program

**defgenerate\_repeated\_chars(input\_str): result=[]**

**i=0**

**while i<len (input\_str): char=input\_str[i] count = 0**

**i+=1**

**while i<len(input\_str) and input\_str[i].isdigit(): count = count\*10+ int(input\_str[i])**

**i+=1**

**result.append(char \* count) return ''.join(result)**

**input\_str1=input() output\_str1=generate\_repeated\_chars(input\_str1) print (output\_str1)**

.

**Department of Computer ScienceandEngineering**|**Rajalakshmi EngineeringCollege**

90

.

**Department of Computer ScienceandEngineering**|**Rajalakshmi EngineeringCollege**

91



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Input** | **Expected** | **Got** |  |
|  | a2b4c6 | Aabbbbcccccc | aabbbbcccccc |  |
|  | a12b3d4 | aaaaaaaaaaaabbbdddd | aaaaaaaaaaaabbbdddd |  |

**-**



**Ex.No. : 5.2 Date:**



**Register No: 231401021 Name:DEVADHARSHINI K S**

# Checkpair with differencek

Robertishaving2stringsconsistofuppercase&lowercaseenglishletters.Nowhewantto compare those two strings lexicographically. The letters' case does not matter, that is an uppercase letter is considered equivalent to the corresponding lowercase letter.

## Input

Thefirst linecontains **T**.Then**T**test casesfollow.

Eachtestcasecontainsatwolinescontainsastring.Thestrings'lengthsrangefrom1to 100 inclusive. It isguaranteed thatthe strings are of the same lengthand also consist of uppercase and lowercase Latin letters.

## Output

Ifthefirststringislessthanthesecondone,print"-1". If the second string is less than the first one, print "1". If the strings are equal, print "0".

Notethattheletters'case isnot takenintoconsiderationwhenthestringsare compared.

## Constraints

**1**≤**T**≤**50**

**String length**≤**100**

**For example:**



|  |  |
| --- | --- |
| **Input** | **Result** |
|  | |
| 3 | 0 |
| aaaa | -1 |
| aaaA | 1 |
| abs |  |
| Abz |  |
| abcdefg |  |
| AbCdEfF |  |

# Program

**for \_ in range(int(input())): s1=input().lower() s2=input().lower()print((s1 > s2) - (s1 < s2))**

.

**Department of Computer ScienceandEngineering**|**Rajalakshmi EngineeringCollege**

92

.

**Department of Computer ScienceandEngineering**|**Rajalakshmi EngineeringCollege**

93



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Input** | **Expected** | **Got** |  |
|  | |  |  |  |
|  | 3 | 0 | 0 |  |
| aaaa | -1 | -1 |
| aaaA | 1 | 1 |
| abs |  |  |
| Abz |  |  |
| abcdefg |  |  |
| AbCdEfF |  |  |

.

**Department of Computer ScienceandEngineering**|**Rajalakshmi EngineeringCollege**

94



**Ex.No. : 5.3 Date:**

**Register No: 231401021 Name:DEVADHARSHINI K S**

# CountElements

GiventwoStringss1ands2,removeallthecharactersfroms1whichispresentins2.

## Constraints

1<=stringlength<=200

## SampleInput1

experienceenc

## SampleOutput1

xpri

PROGRAM

s1=input() s2=input() result = ""

forcharins1:

ifcharnotins2: result+= char

print(result)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | | |  |
|  | **Input** | **Expected** | **Got** |
|  |  |  |  |  |
|  | experience | xpri | xpri |

.

**Department of Computer ScienceandEngineering**|**Rajalakshmi EngineeringCollege**

95



enc

**Got**

**Expected**

**Input**

.

**Department of Computer ScienceandEngineering**|**Rajalakshmi EngineeringCollege**

96



**Ex.No. : 5.4 Date:**

**Register No: 231401021 Name:DEVADHARSHINI K S**

# Distinct ElementsinanArray

Stringshouldcontainonlythewordsarenotpalindrome.

## SampleInput1

Malayalamismymothertongue

## SampleOutput1

ismymothertongue

program

defis\_palindrome(word):

returnword==word[::-1]

deffilter\_non\_palindromic\_words(input\_string): words = input\_string.split()

non\_palindromic\_words=[wordforwordinwordsifnotis\_palindrome(word)] return ' '.join(non\_palindromic\_words)

input\_string=input().lower()

output\_string=filter\_non\_palindromic\_words(input\_string) print(output\_string)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Input** | **Expected** | **Got** |  |  |
|  | Malayalamismymothertongue | ismymother tongue | ismymother tongue |  |



**Ex.No. : 5.4 Date:**

**Register No: 231401021 Name:DEVADHARSHINI K S**

Questiontext

GivenastringS,whichcontainsseveralwords,printthecountCofthewordswhoselength is atleast L. (You can include punctuation marks like comma, full stop also as part of the word length. Space alone must be ignored)

### InputFormat:

The first line contains S. ThesecondlinecontainsL.

### OutputFormat:

ThefirstlinecontainsC

### BoundaryConditions:

2<=Lengthof S<=1000

### ExampleInput/Output1:

Input:

DuringandafterKenyattasinaugurationpoliceelsewhereinthecapital,Nairobi,triedto stop the opposition from holding peaceful demonstrations.

5

Output:

13

Explanation:

Thewordsofminimumlength5are During

.

**Department of Computer ScienceandEngineering**|**Rajalakshmi EngineeringCollege**

97

after Kenyattas inauguration police elsewhere capital, Nairobi,tried opposition holding peaceful

.

**Department of Computer ScienceandEngineering**|**Rajalakshmi EngineeringCollege**

98



demonstrations

Program

S=input()

L=int(input()) words=S.split() count=0

forwordinwords: iflen(word)>=L:

count+=1 print(count)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | | |  |
|  | **Input** | **Expected** | **Got** |
|  |  |  |  |  |
|  | DuringandafterKenyattasinaugurationpoliceelsewhereinthe capital, Nairobi, tried to stop the opposition from holding peaceful demonstrations.  5 | 13 | 13 |

**Ex.No. : 5.6 Date:**

.

**Department of Computer ScienceandEngineering**|**Rajalakshmi EngineeringCollege**

99



**Register No: 231401021 Name:DEVADHARSHINI K S**

# FindtheFactor

Find if a String2 is substring of String1. If it is, return the index of the first occurrence. else return -1.

**Sample Input 1** thistest123string123

## SampleOutput1

8

**Program**

**x=input() y=input() z=x.find(y) print(z)**

**output**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Input** | **Expected** | **Got** |  |
|  | thistest123string 123 | 8 | 8 |  |

**Ex.No. : 5.7 Date:**



**Register No: 231401021 Name:DEVADHARSHINI K S**

# MergeList

Writeaprogramthattakesasinputastring(sentence),andreturnsitssecondwordin uppercase.

For example:

Ifinputis“WiproTechnologiesBangalore”thefunctionshouldreturn“TECHNOLOGIES” If input is “Hello World” the function should return “WORLD”

Ifinputis“Hello”theprogramshouldreturn“LESS”

NOTE1:Ifinputisasentencewithlessthan2words,theprogramshouldreturntheword “LESS”.

NOTE2:Theresultshouldhavenoleadingortrailingspaces.

### For example:



|  |  |
| --- | --- |
| **Input** | **Result** |
| WiproTechnologiesBangalore | TECHNOLOGIES |
| HelloWorld | WORLD |
| Hello | LESS |
|  |  |

Program

defsecond\_word\_uppercase(sentence): words = sentence.split()

if len(words) < 2: return"LESS"

else:

returnwords[1].upper()

.

**Department of Computer ScienceandEngineering**|**Rajalakshmi EngineeringCollege**

100

sentence=input()

.

**Department of Computer ScienceandEngineering**|**Rajalakshmi EngineeringCollege**

101



result=second\_word\_uppercase(sentence) print(result)

output



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Input** | **Expected** | **Got** |  |
|  | WiproTechnologiesBangalore | TECHNOLOGIES | TECHNOLOGIES |  |
|  | HelloWorld | WORLD | WORLD |  |
|  |  |  |  |  |
|  | Hello | LESS | LESS |  |

.

**Department of Computer ScienceandEngineering**|**Rajalakshmi EngineeringCollege**

102



**Ex.No. : 5.8 Date:**



**Register No: 231401021 Name:DEVADHARSHINI K S**

# MergeTwoSortedArraysWithoutDuplication

Writeapythontoreadasentenceand printits longestwordanditslength

### For example:



|  |  |
| --- | --- |
| **Input** | **Result** |
|  | |
| Thisisasampletexttotest | sample 6 |

Program

deflongest\_word(sentence): words = sentence.split() max\_length = 0 longest\_word = ""

forwordin words:

if len(word) >max\_length: max\_length = len(word) longest\_word = word

returnlongest\_word,max\_length

sentence=input()

result=longest\_word(sentence)

print( result[0]) print(str(result[1]))

.

**Department of Computer ScienceandEngineering**|**Rajalakshmi EngineeringCollege**

103

.

**Department of Computer ScienceandEngineering**|**Rajalakshmi EngineeringCollege**

104



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Input** | **Expected** | **Got** |  |
|  | Thisisasampletexttotest | sample 6 | sample 6 |  |
|  | RajalakshmiEngineeringCollege,approvedbyAICTE | Rajalakshmi11 | Rajalakshmi11 |  |
|  | CseITCSBSMCT | CSBS 4 | CSBS 4 |  |

**Ex.No. : 5.9 Date:**



**Register No: 231401021 Name:DEVADHARSHINI K S**

# PrintElementLocation

TwostringvaluesS1,S2arepassedastheinput.TheprogrammustprintfirstNcharacters present in S1 which are also present in S2.

### InputFormat:

The first line contains S1. ThesecondlinecontainsS2. The third line contains N.

### OutputFormat:

ThefirstlinecontainstheNcharacters present inS1whicharealso present inS2.

### BoundaryConditions:

2<= N<= 10

2<=Lengthof S1,S2 <= 1000

### ExampleInput/Output1:

Input:

abcbdecdefghbb3

Output:

bcd

### Note:

boccurs twiceincommonbut mustbeprintedonlyonce.

Program

defextract\_common\_chars(s1,s2,n):

.

**Department of Computer ScienceandEngineering**|**Rajalakshmi EngineeringCollege**

105

common\_chars=[] for char in s1:

.

**Department of Computer ScienceandEngineering**|**Rajalakshmi EngineeringCollege**

106



ifcharins2andcharnotincommon\_chars: common\_chars.append(char)

iflen(common\_chars)==n: break

return ''.join(common\_chars) # Input

s1 = input().strip() s2=input().strip()

n=int(input().strip()) # Output

print(extract\_common\_chars(s1,s2,n))

output



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Input** | **Expected** | **Got** |  |
|  | | | | |
|  | Abcbdecdefghbb3 | bcd | bcd |  |

**Ex.No. : 5.10 Date:**



**Register No: 231401021 Name:DEVADHARSHINI K S**

# Strictlyincreasing

Writeaprogramtocheckiftwostringsarebalanced.Forexample,stringss1 and s2are balancedifallthecharactersinthes1arepresentins2.Thecharacter’spositiondoesn’t matter. If balanced display as "true" ,otherwise "false".

**For example:**



|  |  |
| --- | --- |
| **Input** | **Result** |
|  | |
| YnPYnative | True |

Program

defcheck\_balance(s1, s2): s1\_set = set(s1)

s2\_set=set(s2)

if s1\_set.issubset(s2\_set): return True

else:

returnFalse

s1=input() s2=input()

result=check\_balance(s1,s2)

if result: print("True")

else:

print("False")

.

**Department of Computer ScienceandEngineering**|**Rajalakshmi EngineeringCollege**

107

Output

.

**Department of Computer ScienceandEngineering**|**Rajalakshmi EngineeringCollege**

108



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Input** | **Expected** | **Got** | | |
|  | YnPYnative | True | True |  |
|  | YnfPYnative | False | False |  |

.

**Department of Computer ScienceandEngineering**|**Rajalakshmi EngineeringCollege**

109



**06-StringsinPython**