.

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

141



08–Dictionary

.

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

142



**Ex.No. : 8.1 Date:25.05.24**

**RegisterNo.:230701360 Name:TEJUSHREE SANJEEVIKUMAR**

# Sort DictionarybyValuesSummation

Giveadictionarywithvaluelists,sortthekeysby summationofvaluesinvaluelist.

**Input**:test\_dict={‘Gfg’:[6,7,4],‘best’:[7,6,5]}

**Output**:{‘Gfg’:17,‘best’:18}

**Explanation** : Sorted by sum, andreplaced. **Input** : test\_dict = {‘Gfg’ : [8,8], ‘best’ : [5,5]} **Output** : {‘best’: 10, ‘Gfg’: 16}

**Explanation** : Sorted by sum, and replaced. Sample Input:

2

Gfg674

Best 7 6 5 SampleOutput Gfg 17

Best18

**Forexample:**



|  |  |
| --- | --- |
| **Input** | **Result** |
|  | |
| 2  Gfg674  Best765 | Gfg17  Best18 |

.

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

143



## Program:

n=int(input()) d = {}

foriinrange(n):

s=input().split()

d[s[0]]=list(map(int,s[1:]))

d1={k:sum(v)fork,vind.items()}

sorted\_d = dict(sorted(d1.items(), key=lambda x: x[1])) for k, v in sorted\_d.items():

print(k,v)



**Ex.No. : 8.2 Date:25.05.24**



**RegisterNo.:230701360 Name:TEJUSHREE SANJEEVIKUMAR**

# StudentRecord

Create a student dictionaryfor n students with the student name askey and their testmarkassignmentmarkandlabmarkasvalues.Dothefollowingcomputations and display the result.

1. Identifythestudentwiththehighestaverage score
2. IdentifythestudentwhoasthehighestAssignmentmarks
3. IdentifythestudentwiththeLowestlabmarks
4. Identify the student with the lowest average score Note:

Ifmorethanonestudenthasthesamescoredisplayallthestudentnames Sample input:

4

James678956

Lalith894545

Ram898989

Sita 70 70 70 SampleOutput: Ram

JamesRam Lalith Lalith

## Forexample:



|  |  |
| --- | --- |
| **Input** | **Result** |
|  | |
| 4 | Ram |
| James678956 | JamesRam |
| Lalith894545 | Lalith |
| Ram898989 | Lalith |
| Sita707070 |  |

**Program:**

.

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

144

n=int(input()) d={}



for i in range(n): na=input().split()

d[na[0]]=[int(na[1]),int(na[2]),int(na[3])] l=int(na[3])

h=0

fori ind:

ifh<sum(d[i]):

h=sum(d[i]) j=i h1=sum(d[i])

print(j) h=0

for i in d: if(h<d[i][1]):

h=d[i][1] j=i

for i in d: if(h==d[i][1]):

print(i,end="") l1=[]

k=[]

print()foriind:

.

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

145

if(l>d[i][2]): l=d[i][2] j=i

.

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

146



for i in d: if(l==d[i][2]):

l1.append(i)

foriinrange(-1,-len(l1)-1,-1): print(l1[i],end="")

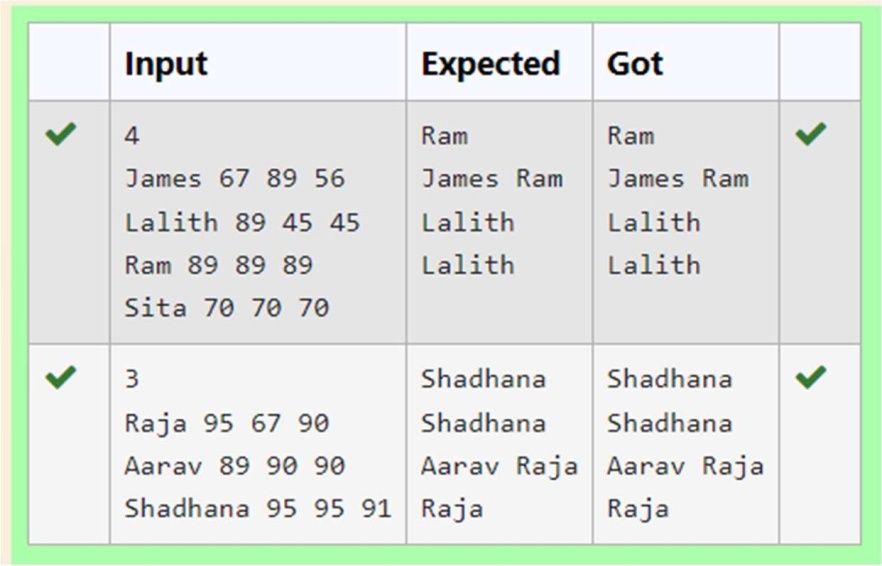
print()

fori ind:

ifh1>sum(d[i]):

h1=sum(d[i]) j=i

print(j)



**Ex.No. : 8.3 Date:25.05.24**



**RegisterNo.:230701360 Name:TEJUSHREE SANJEEVIKUMAR**

# ScrambleScore

In the game of Scrabble™, each letter has points associated with it. The total score of aword is the sum of the scores of its letters. More common letters are worth fewer points while less common letters are worth more points.

Write a program that computes and displays the Scrabble™ score for a word. Create a dictionarythatmapsfromletterstopointvalues.Thenusethedictionaryto computethe score.

AScrabble™boardincludessomesquaresthatmultiplythevalueofaletterorthevalue of an entire word. We will ignore these squares in this exercise.

The points associated with each letter are shown below: Points Letters

1. A,E,I,L,N,O,R,S,TandU
2. DandG
3. B,C,MandP
4. F,H,V,WandY
5. K

8 J and X 10QandZ

SampleInput REC

SampleOutput

RECisworth5points.

## Forexample:

.

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

147



|  |  |
| --- | --- |
| **Input** | **Result** |
|  | |
| REC | RECisworth5points. |

**Program:**

defcalculate\_scrabble\_score(word):

#Dictionarymappingletterstopoints letter\_points = {

'A':1,'B':3,'C':3,'D':2,'E':1,'F':4,'G':2,'H':4,

'I':1,'J':8,'K':5,'L':1,'M':3,'N':1,'O':1,'P':3,

'Q':10,'R':1,'S':1,'T':1,'U':1,'V':4,'W': 4,'X':8,

'Y':4,'Z':10

}

score=0

for letter in word:letter=letter.upper()

score+=letter\_points.get(letter,0)#Addthepointsforeachletter,defaultingto0ifnot found

returnscore

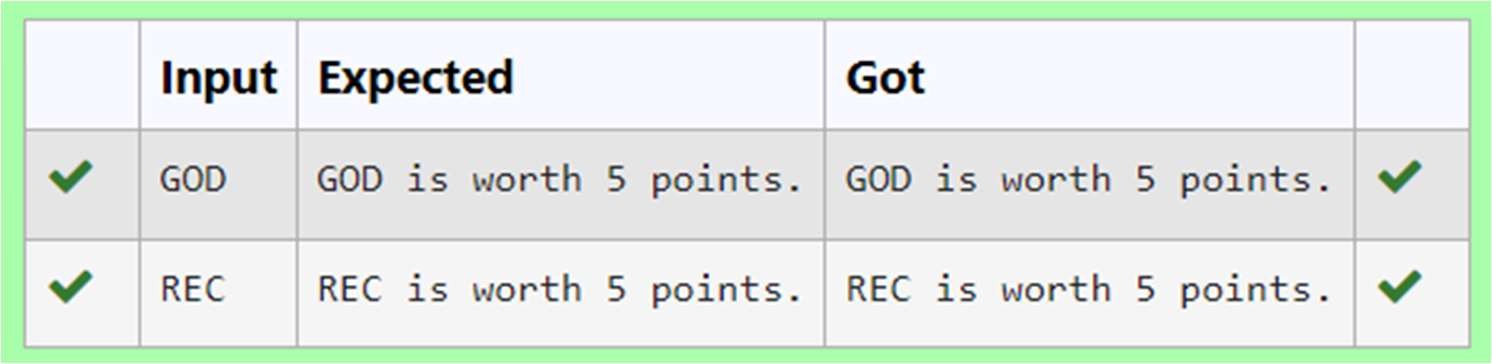
word=input()

score = calculate\_scrabble\_score(word) print(f"{word}isworth{score}points.")

.

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

148





**Ex.No. : 8.4 Date:25.05.24**

**RegisterNo.:230701360 Name:TEJUSHREE SANJEEVIKUMAR**

# Uncommonwords

Asentenceisastringofsingle-spaceseparatedwordswhereeachwordconsistsonlyof lowercaseletters.Awordisuncommonifitappearsexactlyonceinoneofthesentences, and does not appear in the other sentence.

Given two sentences s1 and s2, return a list of all the uncommon words. You may return the answer in any order.

Example1:

Input: s1 = "this apple is sweet", s2 = "this apple is sour" Output: ["sweet","sour"]

Example2:

Input: s1 = "apple apple", s2 = "banana" Output: ["banana"]

Constraints:

1 <=s1.length, s2.length<=200

s1 and s2 consist of lowercase English letters and spaces. s1 and s2 do not have leading or trailing spaces.

All the words in s1 and s2 are separated by a single space.Note:

.

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

149

Usedictionarytosolvetheproblem



**Forexample:**



|  |  |
| --- | --- |
| **Input** | **Result** |
|  | |
| this apple is sweet this apple is sour | sweet sour |

## Program:

s1=input().split() s2=input().split() d = {}

fori ins1:

ifinotind: d[i] = 1

else:

d[i]+=1

fori ins2:

ifinotind: d[i] = 1

else:

d[i]+=1

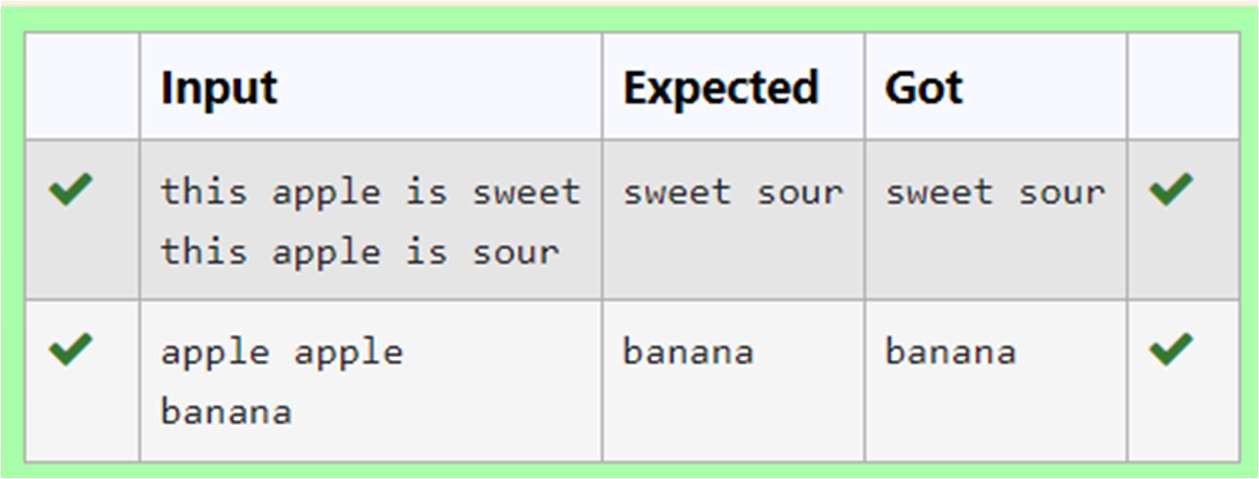
fori ind:

if d[i] == 1: print(i,end="")

.

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

150





**Ex.No. : 8.5 Date:25.05.24**

**RegisterNo.:230701360 Name:TEJUSHREE SANJEEVIKUMAR**

# WinnerofElection

Given an array of names of candidates in an election. A candidate name in the array representsavotecasttothecandidate.PrintthenameofcandidatesreceivedMaxvote. If there is tie, print a lexicographically smaller name.

### Examples:

Input:votes[]={"john","johnny","jackie", "johnny", "john", "jackie",

"jamie","jamie","john",

"johnny", "jamie", "johnny", "john"};

Output:John

We have four Candidates with name as 'John', 'Johnny', 'jamie', 'jackie'. The candidates John and Johny get maximum votes. Since John is alphabetically smaller, we print it. Use dictionary to solve the above problem

### SampleInput:

10

John

.

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

151

John Johny Jamie Jamie Johny Jack Johny Johny Jackie



### SampleOutput:

Johny

### Forexample:



|  |  |
| --- | --- |
| **Input** | **Result** |
|  | |
| 10  John John Johny Jamie Jamie Johny Jack Johny Johny Jackie | Johny |

**Program:** n=int(input()) d={}

foriinrange(n): s=input()

ifsnotind: d[s]=1

else:

d[s]+=1

.

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

152

h=0

.

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

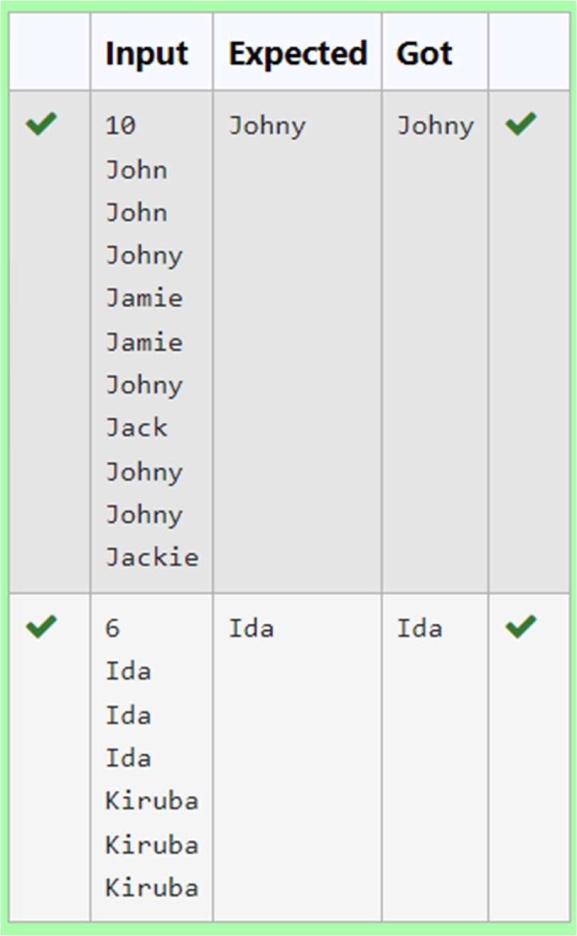
153



fori ind:

ifh<d[i]: h=d[i] j=i

print(j)



.

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

154

