Question **1**

Correct

Mark 1.00 out of 1.00

Flag question

Question text

Write a program to check if two strings are balanced. For example, strings s1 and s2 are balanced if all the characters in the s1 are present in s2. The character’s position doesn’t matter. If balanced display as "true" ,otherwise "false".

**For example:**

| **Input** | **Result** |
| --- | --- |
| Yn  PYnative | True |

Answer:(penalty regime: 0 %)

a=input()

b=input()

if a in b:

print("True")

else:

print("False")

Feedback

|  | **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | Yn  PYnative | True | True |  |
|  | Ynf  PYnative | False | False |  |

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

Assume that the given string has enough memory.

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

**Sample Input 1**

a2b4c6

**Sample Output 1**

aabbbbcccccc

Answer:(penalty regime: 0 %)

a = list(input())

l = []

rep = []

i = 0

while i < len(a):

if a[i].isalpha():

l.append(i)

i += 1

else:

n = 0

while i < len(a) and not a[i].isalpha():

n = n \* 10 + int(a[i])

i += 1

rep.append(n)

s=''

j=0

for i in a:

if i.isalpha():

s+=i\*rep[j]

j+=1

print(s)

Feedback

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

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

**Reverse**a string **without affecting special characters**  
 Given a string **S**, containing special characters and all the alphabets, reverse the string without affecting the positions of the special characters.  
**Input:**A&B  
**Output:**B&A  
**Explanation**: As we ignore '&' and  
As we ignore '&' and then reverse, so answer is "B&A".

**For example:**

| **Input** | **Result** |
| --- | --- |
| A&x# | x&A# |

Answer:(penalty regime: 0 %)

def reverse\_string(s):

s = list(s)

left, right = 0, len(s) - 1

while left < right:

if not s[left].isalnum():

left += 1

elif not s[right].isalnum():

right -= 1

else:

s[left], s[right] = s[right], s[left]

left += 1

right -= 1

return ''.join(s)

input\_string = input()

reversed\_string = reverse\_string(input\_string)

print(reversed\_string)

Feedback

|  | **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | A&B | B&A | B&A |  |

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

 Write a python to read a sentence and print its longest word and its length

**For example:**

| **Input** | **Result** |
| --- | --- |
| This is a sample text to test | sample  6 |

Answer:(penalty regime: 0 %)

s=input()

w=s.split()

Lw=max(w,key=len)

LLw=len(Lw)

print(Lw)

print(LLw)

Feedback

|  | **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | This is a sample text to test | sample  6 | sample  6 |  |
|  | Rajalakshmi Engineering College, approved by AICTE | Rajalakshmi  11 | Rajalakshmi  11 |  |
|  | Cse IT CSBS MCT | CSBS  4 | CSBS  4 |  |

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

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 %)

17

s1=input()

s2=input()

N=int(input())

s2set=set(s2)

cc=[]

c=0

for char in s1:

if char in s2set and char not in cc:

cc.append(char)

c=c+1

if c==N:

break

x=''.join(cc)

print(x)

Feedback

|  | **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | abcbde  cdefghbb  3 | bcd | bcd |  |

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

Robert  is having 2 strings consist of uppercase & lowercase english letters. Now he want to 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

The first line contains **T**. Then **T** test cases follow.

Each test case contains a two lines contains a string. The strings' lengths range from 1 to 100 inclusive. It is guaranteed that the strings are of the same length and also consist of uppercase and lowercase Latin letters.

Output

If the first string is less than the second one, print "-1".  
If the second string is less than the first one, print "1".  
If the strings are equal, print "0".  
Note that the letters' case is not taken into consideration when the strings are compared.

Constraints

**1**≤**T**≤**50**

**String length**≤**100**

**For example:**

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

Answer:(penalty regime: 0 %)

def compare\_strings(str1, str2):

# Convert both strings to lowercase

str1\_lower = str1.lower()

str2\_lower = str2.lower()

# Compare the strings lexicographically

if str1\_lower < str2\_lower:

return -1

elif str1\_lower > str2\_lower:

return 1

else:

return 0

# Input number of test cases

T = int(input())

# Iterate through each test case

for \_ in range(T):

# Input two strings

str1 = input().strip()

str2 = input().strip()

# Compare the strings and print the result

result = compare\_strings(str1, str2)

print(result)

Feedback

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

Consider the below words as key words and check the given input is key word or not.

keywords: {break, case, continue, default, defer, else, for, func, goto, if, map, range, return, struct, type, var}

Input format:

Take string as an input from stdin.

Output format:

Print the word is key word or not.

Example Input:

break

Output:

break is a keyword

Example Input:

IF

Output:

IF is not a keyword

**For example:**

| **Input** | **Result** |
| --- | --- |
| break | break is a keyword |
| IF | IF is not a keyword |

Answer:(penalty regime: 0 %)

a={"break","case","continue","default", "defer", "else", "for", "func", "goto", "if", "map", "range", "return", "struct", "type", "var"}

b=input()

if b in a:

print(b,"is a keyword")

else:

print(b,"is not a keyword")

Feedback

|  | **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | break | break is a keyword | break is a keyword |  |
|  | IF | IF is not a keyword | IF is not a keyword |  |

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

Given a string S, which contains several words, print the count C of the words whose length is atleast L. (You can include punctuation marks like comma, full stop also as part of the word length. Space alone must be ignored)

**Input Format:**

The first line contains S.  
The second line contains L.

**Output Format:**

The first line contains C

**Boundary Conditions:**

2 <= Length of S <= 1000

**Example Input/Output 1:**

Input:

During and after Kenyattas inauguration police elsewhere in the capital, Nairobi, tried to stop the opposition from holding peaceful demonstrations.  
5

Output:

13

Explanation:

The words of minimum length 5 are  
During  
after  
Kenyattas  
inauguration  
police  
elsewhere  
capital,  
Nairobi,  
tried  
opposition  
holding  
peaceful  
demonstrations.

Answer:(penalty regime: 0 %)

def count\_words\_with\_length(s, l):

# Split the string into words

words = s.split()

# Initialize count

count = 0

# Iterate through words and count those with length at least L

for word in words:

if len(word) >= l:

count += 1

return count

# Input the string

s = input("")

# Input the minimum length L

l = int(input())

# Count the words with length at least L

count = count\_words\_with\_length(s, l)

# Output the count

print(count)

Feedback

|  | **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | During and after Kenyattas inauguration police elsewhere in the capital, Nairobi, tried to stop the opposition from holding peaceful demonstrations.  5 | 13 | 13 |  |

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

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 %)

s1=str(input())

s2=str(input())

x="".join(char for char in s1 if char not in s2)

print(x)

Feedback

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

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

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 %)

a=input().lower()

b=a.split()

s=''

for i in range(len(b)):

c=b[i][::-1]

if c==b[i]:

continue

s+=b[i]+' '

print(s)

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.