07 – Tuple/Set

Ex. No. : 7.1 **Date: 18.05.24**

Register No.: 230701385 **Name: S. Vishwak**

.

Binary String

Coders here is a simple task for you, Given string str. Your task is to check whether it is a binary string or not by using python set.

Examples:

Input: str = "01010101010"

Output: Yes

Input: str = "REC101"

Output: No

For example:

Input	Result
01010101010	Yes
010101 10101	No

Program:

```
str1=input()
s1={'0','1'}
s2=' '
for i in s1:
    if i in str1 and str1.isdigit():
        print('Yes')
        break
    else:
```


```
print('No')
break
```

	Input	Expected	Got	
✔	01010101010	Yes	Yes	✔
✔	REC123	No	No	✔
✔	010101 10101	No	No	✔

Ex. No. : 7.2 **Date: 18.05.24**

Register No.: 230701385 **Name: S. Vishwak**

.

DNA Sequence

The **DNA sequence** is composed of a series of nucleotides abbreviated as 'A', 'C', 'G', and 'T'.

For example, "ACGAATTCCG" is a **DNA sequence**.

When studying **DNA**, it is useful to identify repeated sequences within the DNA.

Given a string s that represents a **DNA sequence**, return all the **10-letter-long** sequences (substrings) that occur more than once in a DNA molecule. You may return the answer in **any order**.

Example 1:

Input: s = "AAAAACCCCCAAAAACCCCCCAAAAAGGGTTT"
Output: ["AAAAACCCCC","CCCCCAAAAA"]

Example 2:

Input: s = "AAAAAAAAAAAAA"
Output: ["AAAAAAAAAA"]

For example:

Input	Result
AAAAACCCCCAAAAACCCCCCAAAAAGGGTTT	AAAAACCCCC CCCCCAAAAA

Program:

```
n = input()
length = 10
for i in range(len(n)):
    if(n.count(n[0])==len(n)):
        print(n[0:length])
        break
    else:
        substring = n[i:i+length]
        if n.count(substring) > 1:
            print(substring)
            print(substring[::-1])
            break
```

	Input	Expected	Got	
✔	AAAAACCCCCAAAAACCCCCCAAAAAGGGTTT	AAAAACCCCC CCCCCAAAAA	AAAAACCCCC CCCCCAAAAA	✔
✔	AAAAAAAAAAAAA	AAAAAAAAAA	AAAAAAAAAA	✔

Ex. No. : 7.3 **Date: 18.05.24**

Register No.:230701385 **Name: S. Vishwak**

There is a malfunctioning keyboard where some letter keys do not work. All other keys on the keyboard work properly.

Given a string text of words separated by a single space (no leading or trailing spaces) and a string brokenLetters of all distinct letter keys that are broken, return the number of words in text you can fully type using this keyboard.

Example 1:

Input: text = "hello world", brokenLetters = "ad"

Output:

1

Explanation: We cannot type "world" because the 'd' key is broken.

For example:

Input	Result
hello world ad	1
Faculty Upskilling in Python Programming ak	2

Program:

```
str1=input()

bl=input()

count=0

j=1

for i in range(len(bl)):
```

```
    if bl[i:j] in str1:
        count+=1
        j+=1
print(count)
```

Ex. No. : 7.4 **Date: 18.05.24**

Register No.: 230701385 **Name S. Vishwak**

.

Print repeated no

Given an array of integers nums containing n + 1 integers where each integer is in the range [1, n] inclusive.There is only **one repeated number** in nums, return *this repeated number*. Solve the problem using set.

Example 1:

Input: nums = [1,3,4,2,2]
Output: 2

Example 2:

Input: nums = [3,1,3,4,2]
Output: 3

For example:

Input	Result
1 3 4 4 2	4

Program:

```
n = input()
n=n.split()
t = []
for i in n:
    if i.isdigit():
        t.append(int(i))
t=tuple(t)
for i in t:
```

```
if(t.count(i)>1):
    print(i)
    break
else:
    continue
```

Ex. No. : 7.5 **Date: 18.05.24**

Register No.:230701385 **Name: S. Vishwak**

Write a program to eliminate the common elements in the given 2 arrays and print only the non-repeating

elements and the total number of such non-repeating elements.

Input Format:

The first line contains space-separated values, denoting the size of the two arrays in integer format respectively.

The next two lines contain the space-separated integer arrays to be compared.

Sample Input:

5 4

1 2 8 6 5

2 6 8 10

Sample Output:

1 5 10

3

Sample Input:

5 5

1 2 3 4 5

1 2 3 4 5

Sample Output:

NO SUCH ELEMENTS

For example:

Input	Result
5 4 1 2 8 6 5 2 6 8 10	1 5 10 3

Program:

```
n = input().split()
t = []
for i in n:
    if i.isdigit():
        t.append(int(i))
s1=input().split()
a=[]
for i in s1:
    if i.isdigit():
        a.append(int(i))
s2 = input().split()
b = []
for i in s2:
    if i.isdigit():
        b.append(int(i))
c=set(a)^set(b)
d=set(a)&set(b)
count=1
for i in c:
    if count<len(c):
        print(i,end=' ')
        count+=1
    else:
        print(i,end='\n')
print(len(c))
```