08 – Tuple/Set

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

Register No.: 230701362 **Name: THARUNRAJ I**

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
0101010101	No

CODE:

```
s=input()
f=0
for i in s:
    if(i!='0' and i!='1'):
        f=1
        break
if(f==1):
    print("No")
else:
    print("Yes")
```

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

Register No.: 230701362 **Name: THARUNRAJ I**

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

CODE:

```
l=list(map(int,input().split()))
s=set(l)
for i in s:
    if(l.count(i)>1):
        print(i)
        break
```

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

Register No.: 230701362 **Name: THARUNRAJ I**

Remove repeated

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

CODE:

```
n,m=map(int,input().split())
s1=set(map(int,input().split()))
s2=set(map(int,input().split()))
s3=s1^s2
if(len(s3)>0):
    for i in s3:
        print(i,end=" ")
    print("")
    print(len(s3))
else:
    print("NO SUCH ELEMENTS")
```

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

Register No.: 230701362 **Name: THARUNRAJ I**

Malfunctioning Keyboard

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

CODE:

```
s=list(map(str,input().split()))
bs=input()
co=0
for i in s:
```

```
    f=0
    for j in bs:
        if(j in i.lower()):
            f=1
    if(f==1):
        co+=1
print(len(s)-co)
```

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

Register No.: 230701362 **Name: THARUNRAJ I**

American keyboard

Given an array of strings words, return *the words that can be typed using letters of the alphabet on only one row of American keyboard like the image below.*

In the **American keyboard**:

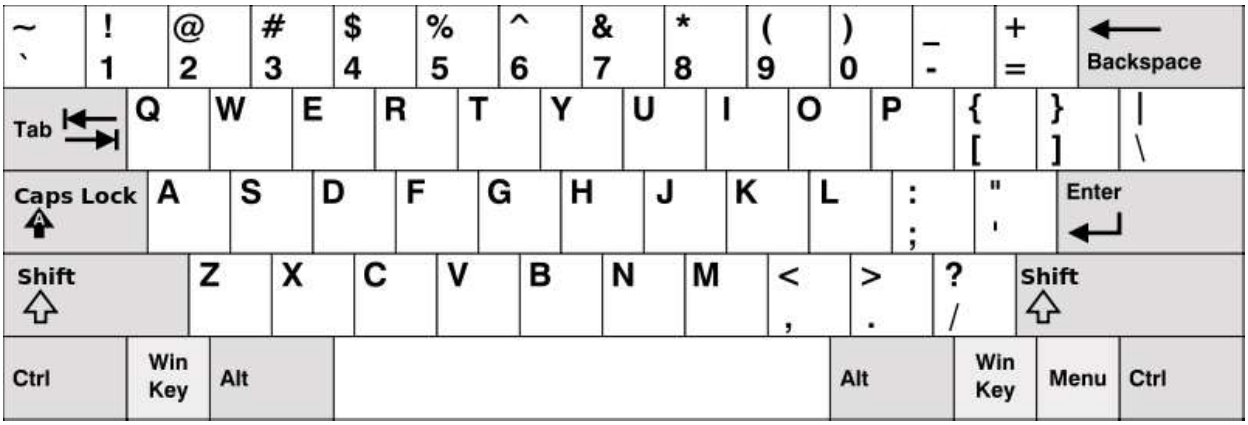
- the first row consists of the characters "qwertyuiop",
- the second row consists of the characters "asdfghjkl", and

the third row consists of the characters "zxcvbnm".

CODE:

```
n=int(input())

l=[]

f=0

for i in range(n):

    l.append(input())

for i in l:

    r1,r2,r3=0,0,0

    for j in i:

        if(j.lower() in "qwertyuiop"):

            r1+=1

        elif(j.lower() in "asdfghjkl"):

            r2+=1

        else:

            r3+=1


    if(r1+r2==0 or r1+r3==0 or r3+r2==0):
```

```
        print(i)
        f=1
if(f==0):
    print("No words")
```



Example 1:
Input: words = ["Hello","Alaska","Dad","Peace"]
Output: ["Alaska","Dad"]
Example 2:

Input: words = ["omk"]
Output: []
Example 3:

Input: words = ["adsdf","sfd"]
Output: ["adsdf","sfd"]

For example:

Input	Result
4 Hello Alaska Dad Peace	Alaska Dad