Started on	Monday, 10 June 2024, 7:30 PM
State	Finished
Completed on	Monday, 10 June 2024, 10:40 PM
Time taken	3 hours 10 mins
Marks	5.00/5.00
Grade	**100.00** out of 100.00

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

Answer: (penalty regime: 0 %)

```python
1  a=(input())
2  cnt=0
3  for i in range(len(a)):
4      if a[i]=='0' or a[i]=='1':
5          cnt=cnt+1
6  if cnt==len(a):
7      print('Yes')
8  else:
9      print('No')
```

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

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

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

Answer: (penalty regime: 0 %)

```python
1  nums=list(map(int,input().split()))
2  duplicate=sum(nums)-sum(set(nums))
3  print(duplicate)
```

	Input	Expected	Got	
✓	1 3 4 4 2	4	4	✓
✓	1 2 2 3 4 5 6 7	2	2	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Given a tuple and a positive integer k, the task is to find the count of distinct pairs in the tuple whose sum is equal to **K**.

Examples:

Input: t = (5, 6, 5, 7, 7, 8), K = 13

Output: 2

Explanation:

Pairs with sum K(= 13) are {(5, 8), (6, 7), (6, 7)}.

Therefore, distinct pairs with sum K(= 13) are { (5, 8), (6, 7) }.

Therefore, the required output is 2.

For example:

Input	Result
1,2,1,2,5 3	1
1,2 0	0

Answer: (penalty regime: 0 %)

```
1  a=input()
2  b=int(input())
3  c=a.split(',')
4  d=len(c)
5  e=[]
6  for i in range(d):
7      for k in range(1,d):
8          f=[]
9          g=(int(c[i])+int(c[k]))
10         if g==b:
11             f.append(int(c[i]))
12             f.append(int(c[k]))
13             f.sort()
14             f=tuple(f)
15             e.append(f)
16 h=set(e)
17 print(len(h))
```

	Input	Expected	Got	
✓	5,6,5,7,7,8 13	2	2	✓
✓	1,2,1,2,5 3	1	1	✓
✓	1,2 0	0	0	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

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
5 5 1 2 3 4 5 1 2 3 4 5	NO SUCH ELEMENTS

Answer: (penalty regime: 0 %)

```
1   a=input()
2   b=input().split()
3   c=input().split()
4   d=set(b)
5   e=set(c)
6   f=d^e
7   if len(f)!=0:
8       g=list(f)
9       g=list(map(int,g))
10      g.sort()
11      print(" ".join(map(str,g)))
12      print(len(g))
13   else:
14        print("NO SUCH ELEMENTS")
```

✓	3 3 10 10 10 10 11 12	11 12 2	11 12 2	✓
✓	5 5 1 2 3 4 5 1 2 3 4 5	NO SUCH ELEMENTS	NO SUCH ELEMENTS	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

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

Answer: (penalty regime: 0 %)

```
1  text=input().lower()
2  brokenLetters="".join(input().split())
3  count=0
4  for word in text.split():
5      if not any(letter in brokenLetters for letter in word):
6          count+=1
7  print(count)
```

	Input	Expected	Got	
✓	hello world ad	1	1	✓
✓	Welcome to REC e	1	1	✓
✓	Faculty Upskilling in Python Programming ak	2	2	✓

Passed all tests! ✓

Correct
Marks for this submission: 1.00/1.00.