Question-1:

In this challenge, you are required to calculate and print the sum of the elements in an array, keeping in mind that some of those integers may be quite large.

Function Description

Complete the aVeryBigSum function in the editor below. It must return the sum of all array elements.

aVeryBigSum has the following parameter(s):

- int ar[n]: an array of integers .

Return

- long: the sum of all array elements

Input Format

The first line of the input consists of an integer n.

The next line contains n space-separated integers contained in the array.

Output Format

Return the integer sum of the elements in the array.

Constraints

$1 \leq n \leq 10$

$0 \leq ar[i] \leq 10^{10}$

Sample Input

```
5
1000000001 1000000002 1000000003 1000000004 1000000005
```

Output

```
5000000015
```

Note:

The range of the 32-bit integer is (-2^{31}) *to* $(2^{31} - 1)$ *or* $[-2147483648, 2147483647]$.

When we add several integer values, the resulting sum might exceed the above range.

You might need to use long int C/C++/Java to store such sums.

Code:

```
def aVeryBigSum(ar):
    sum = 0
    for i in range(len(ar)):
      sum+=ar[i]
    return sum
```

Question-2:

Different Operations In-Sets

Sumanth having N number of elements and he decided to create a set for it. After creating the set he wants to add some elements to the set, remove elements from the set, find the difference between two sets, and he wants to know the intersection elements in two sets.

All the process will be shown in input and output format specifications.

So can you please help him to write a program to add(), remove(), intersection(), difference() to the sets.

Input Format Specifications:

- The first line of input creates set 1 with Strings (str).
- The second line of input creates set 2 with Strings (str).

- The next line will choose one option in which operation you want in sets.
- Next line will be choose set for operation
- The sample input showed the sample input format.
- Note that print the elements in sorted order

Output Format Specifications:

- The output consists of a single line as per your choice.
- Display ‘invalid choice’, if the entered choice is wrong.
- Sample Input and Output showed below

Sample Input1:
banana,apple,sony
mango,apple,samsung
choose one option 1)add 2)remove 3)difference 4)intersection
1
choose one 1)set1 2)set2
1
orange
Sample Output 1:
set(['apple','banana',orange','sony',])

Sample Input 2:
banana,apple,sony
mango,apple,samsung
choose one option 1)add 2)remove 3)difference 4)intersection
2
choose one 1)set1 2)set2
1
apple
Sample Output2:
set(['banana','sony'])

Sample Input 3:
banana,apple,sony
mango,apple,samsung
choose one option 1)add 2)remove 3)difference 4)intersection
3
choose one 1)set1 2)set2
2

Sample Output3:
set(['mango','samsung'])

Sample Input 4:
banana,apple,sony
mango,apple,samsung
choose one option 1)add 2)remove 3)difference 4)intersection
4
choose one 1)set1 2)set2
1
Sample Output 4:
set(['apple'])

Sample Input 5:
banana,apple,sony
mango,apple,samsung
choose one option 1)add 2)remove 3)difference 4)intersection
3
choose one 1)set1 2)set2
3
Sample Output 5:
invalid choice

Code:

```
i1=input()
i2=input()
set1 = set(map(str,i1.split(',')))
set2 = set(map(str,i2.split(',')))
set3={}
print("choose one option",end=" ")
print("1)add",end=" ")
print("2)remove",end=" ")
print("3)difference",end=" ")
print("4)intersection ")
ch1=int(input())
print("choose one",end=" ")
print("1)set1",end=" ")
print("2)set2")
```

```
ch2=int(input())
if(ch1==1):
    if(ch2==1):
        ele=input()
        set1.add(ele)
        print("set({})".format(sorted(set1)))
    elif(ch2==2):
        ele=input()
        set2.add(ele)
        print("set({})".format(sorted(set2)))
    else:
        print("invalid choice")
if(ch1==2):
    if(ch2==1):
        ele=input()
        set1.remove(ele)
        print("set({})".format(sorted(set1)))
    elif(ch2==2):
        ele=input()
        set2.remove(ele)
        print("set({})".format(sorted(set2)))
    else:
        print("invalid choice")
if(ch1==3):
    if(ch2==1):
        set3=set1-set2
        print("set({})".format(sorted(set3)))
    elif(ch2==2):
        set3=set2-set1
        print("set({})".format(sorted(set3)))
    else:
        print("invalid choice")
elif(ch1==4):
    if(ch2==1):
        set3=set1 & set2
        print("set({})".format(sorted(set3)))
    elif(ch2==2):
```

```
        set3=set1 & set2
        print("set({})".format(sorted(set3)))
    else:
        print("invalid choice")
```

Question-3:

Validation of Password

Sitha wanted to purchase some clothes for her sister's wedding. One of her friends suggested purchasing online through fashion app so that she can get the clothes with discount.

Now she downloads the 'fashion' app since she is the new user to 'fashion app', she has to create new user account. In order to create new account, app asked her to enter a password.
The password should contain following rules.
At least 1 letter between [a-z] and 1 letter between [A-Z].
At least 1 number between [0-9].
At least 1 character from [$#@].
Minimum length 6 characters.
Maximum length 16 characters.

Help her to create Password.

Input format
Input is a string which indicates password.

Output format
The output is a string containing 'Valid Password' or 'Invalid Password'.

Sample Input and Output 1:
W3r@100a
Valid Password

Sample Input and Output 2:
W3r@100a123rttrddxsali
Invalid Password

Code:

```
password = input()
if len(password) < 6 or len(password) > 16:
    print("Invalid Password")
else:
    has_lower = False
    has_upper = False
    has_digit = False
    has_special = False
    special_characters = {'$', '#', '@'}
    for char in password:
        if char.islower():
            has_lower = True
        elif char.isupper():
            has_upper = True
        elif char.isdigit():
            has_digit = True
        elif char in special_characters:
            has_special = True
    if has_lower and has_upper and has_digit and has_special:
        print("Valid Password")
    else:
        print("Invalid Password")
```

Question-4:

Symmetric_Difference

Gokul have two different sets and he wants to find the Symmetric_Difference between two sets. The symmetric difference of two sets A and B is the set of elements that are in either of the sets A or B but not in both.

So can u please help to write a program to find the symmetric_Difference between two sets.
Input and Output format will be shown below.

Input Format Specifications:

- Firstline contains to enter the elements to set1(integers).
- Second-line contains to enter the elements to set2(integers).

- Note that print the elements in sorted order.

Output Format Specifications:

- Output Consists of Symmetric_difference between set1 and set2 (Integers)
- If both set elements are equal to print ‘invalid set’.

Sample Input1:
1,2,3,4,5,6
2,3,5,7,8,9
Sample Output1:
{1, 4, 6, 7, 8, 9}

Sample input2:
1,2,3
1,2,3
Sample Output2:
invalid set

Code:

```
def symmetric_difference(set1, set2):
    sym_diff = set1.symmetric_difference(set2)
    if not sym_diff:
        return 'invalid set'
    else:
        return sym_diff
set1 = set(map(int, input().split(',')))
set2 = set(map(int, input().split(',')))
result = symmetric_difference(set1, set2)
if result == 'invalid set':
    print(result)
else:
    sorted_result = sorted(result)
    print("{" + ", ".join(map(str, sorted_result)) + "}")
```

Question-5:

Push All Zero

Write a program to push all zeros to the end of a given tuple. The order of the elements should not change. Refer sample input and output for formatting.

Input Format:

Integer elements in a tuple need to be separated by a space.
Refer sample input and output for formatting specifications

[All text in bold corresponds to the input and the rest corresponds to the output.]

SAMPLE INPUT AND OUTPUT:

0 4 0 6 0 7 0 8
Initial Value: (0, 4, 0, 6, 0, 7, 0, 8)
Final value: (4, 6, 7, 8, 0, 0, 0, 0)

Code:

```
t=tuple(map(int,input().split()))
print("Initial Value:",t)
y=list(t)
x=[]
for i in range(len(y)):
    if y[i]!=0:
        x.append(y[i])
for j in range(len(y)):
    if y[j]==0:
        x.append(y[j])
z=tuple(x)
print("Final value:",z)
```