



X

Your A Very Big Sum submission got 10.00 points.

nare F

Post

You are now 69 points away from the 2nd star for your problem solving badge.

Try the next challenge | Try a Random Challenge

Problem

Submissions

Leaderboard

Editorial 🛆

In this challenge, you need to calculate and print the sum of elements in an array, considering that some integers may be very large.

Function Description

Complete the **aVeryBigSum** function with the following parameter(s):

• int ar[n]: an array of integers

Return

• long: the sum of the array elements

Input Format

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

The next line contains \boldsymbol{n} space-separated integers contained in the array.

Output Format

Return the integer sum of the elements in the array.

Constraints

 $1 \le n \le 10$

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

Sample Input

STDIN				Function	
5				arr[] size	n = 5
1000000001	1000000002	1000000003	1000000004	1000000005	arr[]

Output

5000000015

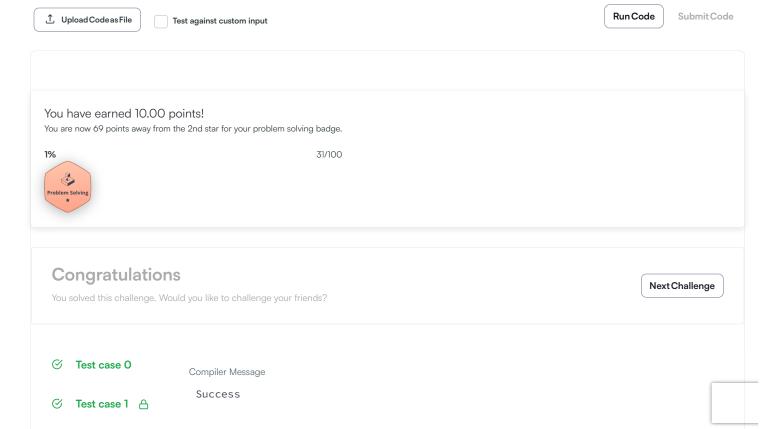
Note:

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.

```
1
     #!/bin/python3
 2
 3
     import math
     import os
 4
     import random
     import re
     import sys
 8
 9
10
     # Complete the 'aVeryBigSum' function below.
11
     # The function is expected to return a LONG_INTEGER.
12
     \ensuremath{\mathtt{\#}} The function accepts LONG_INTEGER_ARRAY ar as parameter.
13
14
15
16
     def aVeryBigSum(ar):
17
         # Write your code here
         return sum(ar)
18
19
     if __name__ == '__main__':
20
         fptr = open(os.environ['OUTPUT_PATH'], 'w')
21
22
         ar_count = int(input().strip())
23
24
25
         ar = list(map(int, input().rstrip().split()))
26
         result = aVeryBigSum(ar)
27
         fptr.write(str(result) + '\n')
29
30
31
         fptr.close()
32
```

EMACS Line: 32 Col: 1



Inpu	t (stdin)	Download
1	5	
2	1000000001 1000000002 1000000003 1000000004 1000000005	
Ехре	ected Output	Download
1	500000015	

Blog | Scoring | Environment | FAQ | About Us | Helpdesk | Careers | Terms Of Service | Privacy Policy