



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

ADVANCED SUB ARRAY PROBLEM

Description

You are competing in a basketball contest. In this contest the score for each successful shot depends on both the distance from the basket and the player's position. The ball is shot N times, successfully. You are given an array A containing the distance of a player from basket for N shots. The index of array represents the position of the player. Score is calculated by multiplying the position with the distance from the basket.

Your task is to find and return an integer value, representing the maximum possible score you can achieve by choosing a contiguous subarray of size K from the given array.

Note:

- * A subarray is a contiguous part of array.
- * Assume 1 based indexing.
- * The array contains both negative and positive values.
- * Assume the player is standing on a cartesian plane.

Input Format

- **input1**: An integer value N representing the number of shots made by the player
- **input2** : An integer K representing the size of subarray
- **input3** : An array of integers

Sample Input

5
2
1 2 3 4 5

Sample Output

14

Source Code:

```
goals=int(input())
size=int(input())
l=list(map(int,input().split()))
max=0
for i in range(0,len(l)):
    sub=l[i:i+size]
    k=1
    s=0
    for j in sub:
        s+=(j*k)
        k+=1
    if s>max:
        max=s
print(max)
```

RESULT

5 / 5 Test Cases Passed | 100 %



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

ANT ON RAIL

Description

There is a ant on your balcony. It wants to leave the rail so sometimes it moves right and sometimes it moves left until it gets exhausted. Given an integer array A of size N which consists of integer 1 and -1 only representing ant's moves.

Where 1 means ant moved unit distance towards the right side and -1 means it moved unit distance towards the left. Your task is to find and return the integer value representing how many times the ant reaches back to original starting position.

Note:

- Assume 1-based indexing
- Assume that the railing extends infinitely on the either sides

Input Format:

input1 : An integer value N representing the number of moves made by the ant.

input2 : An integer array A consisting of the ant's moves towards either side

Sample Input

5

1 -1 1 -1 1

Sample Output

2

Source Code:

```
n=int(input())
arr=list(map(int,input().split()))
summ=0
count=0
for i in range(len(arr)):
    summ+=arr[i]
    if summ==0:
        count+=1
print(count)
```

RESULT

5 / 5 Test Cases Passed | 100 %

KUB

SE01

B23C

2KU
SE01

CSE0
JB23

UB23
J2K

D12
3C2K



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

CHOCOLATE JAR

Description

You are given an integer array of size N, representing jars of chocolates. Three students A, B, and C respectively, will pick chocolates one by one from each chocolate jar, till the jar is empty, and then repeat the same with the rest of the jars. Your task is to find and return an integer value representing the total number of chocolates that student A will have, after all the chocolates have been picked from all the jars.

Note: Once a jar is done A will start taking the chocolates from the new jar.

Input Format :

input1: An integer value N representing the number of jars.

input2: An integer array representing the quantity of chocolates in each jar.

Output Format:

Return an integer value representing the total number of chocolates that student A will have, after all the chocolates are picked.

Example:**Input:**

3

10 20 30

Output:

21

Explanation:

Jar 1: 10 chocolates -> A-4, B-3,C-3

Jar 2: 20 chocolates -> A-7, B-7, C-6

Jar 3: 30 chocolates -> A-10, B-10,C-10

so A gets a total of $4+7+10=21$ chocolates.

Source Code:

```
n=int(input())
arr=list(map(int,input().split()))
summ=0
for i in arr:
    if i%3==0:
        summ+=(i//3)
    elif i%3>0:
        summ+=(i//3)+1
print(summ)
```

RESULT

5 / 5 Test Cases Passed | 100 %



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

DOG AGE

Description

Max has a dog, which is an integer N years old. Now he wants the age of his dog in human years. The internet says that 1 dog year equals to 7 human years. Your task is to find and return an integer value representing the age of Max's dog in human years.

Input Format:

input1: An integer value N representing the age of Max's dog

Output Format:

Return an integer value representing the age of Max's dog in human years

Example:**Input:**

4

Output:

28

Source Code:

```
n=int(input())
dog=n*7;
print(dog)
```

RESULT

5 / 5 Test Cases Passed | 100 %



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

ELECTIONS

Description

You are the head of the election committee in your village. Each Political party is associated with a unique number and the votes are represented as an integer array A. where each element contains the party number voted for by the villagers. For a party to win, they must have a majority of votes. our task is to find and return an integer value denoting the winning party's number. Return -1 if there is no party with a majority.

Note: If only one vote is there he is the winner.

Input Format :

input1: An integer value representing the number the number of voters

input2: An integer array A representing the votes of the voters.

output Format:

Return an integer value denoting the winning party's number.Return -1 there is no party with a majority

Example 1:**Input:**

6

1 1 2 2 2 3

Output:

2

Explanation:

As 2 got the most number of votes i.e 3.

Example 2:**Input:**

6

1 2 1 1 2 2

Output:

-1

Explanation:

As both the contestants got same votes there is no majority.

Source Code:

```
n = int(input())
arr = list(map(int, input().split()))
d = {}
for num in arr:
    if num not in d:
        d[num] = 1
    else:
        d[num] += 1
if n == 1:
    print(arr[0])
else:
    x = sorted(d.items(), key=lambda x: x[1], reverse=True)
    if len(x)>1 and x[0][1] == x[1][1]:
        print("-1")
    else:
        print(x[0][0])
```

RESULT

6 / 6 Test Cases Passed | 100 %



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

VOWEL REPETITION PROBLEM

Description

Given a string s print the most frequent vowel that is present in the string as a output.

Input Format:

A single line containing the string s.

The input will be read from the STDIN by the candidate

Output Format:

Print a single character which represents the most frequent vowel in the given string.

Example:**Input:**

helloworld

Output:

o

Source Code:

```
def most_frequent_vowel(s):
    vowels = 'aeiou'
    vowel_count = {vowel: 0 for vowel in vowels}

    for char in s:
        if char.lower() in vowels:
            vowel_count[char.lower()] += 1

    max_count = max(vowel_count.values())
    most_frequent = [vowel for vowel, count in vowel_count.items() if count == max_count]

    return most_frequent[0]

s = input()
print(most_frequent_vowel(s))
```

RESULT

5 / 5 Test Cases Passed | 100 %

KU,
^3CSE0,
, KUB23-
SE012 K
^823CSE
^UB23C
^2KU012^
^3CSE0
^3CSL
^UB23
^KUB23^012
^CSE012
^3CSE0,
^UB23^012
^CSE0 KI,



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

SPACE COUNTER

Description

You have been given the task of making the content on a social media platform more user-friendly. Your task is to find and return an integer value representing the count of the number of spaces in a given string S.

Input:

A string S

Output :

Return an integer value representing the count of the number of spaces in a given string S.

Example:**Input:**

Hello World Hey

Output:

2

Source Code:

```
def count_spaces(s):
    return s.count(' ')
# Example usage
s = input()
space_count = count_spaces(s)
print(space_count)
```

RESULT

5 / 5 Test Cases Passed | 100 %



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

MINIMUM ARRAY SUM

Description

Paul is given an array A of length N. He must perform the following Operations on the array sequentially:

- * Choose any two integers from the array and calculate their average.
- * If an element is less than the average, update it to 0. However, if the element is greater than or equal to the average, he need not update it.

Your task is to help Paul find and return an integer value, representing the minimum possible sum of all the elements in the array by performing the above operations.

Note: An exact average should be calculated, even if it results in a decimal.

Input Format:

input1: An integer value N, representing the size of the array A.

input2: An integer array A.

Output Format:

Return an integer value, representing the minimum possible sum of all the elements in the array by

Sample Input

5
1 2 3 4 5

Sample Output

5

Source Code:

```
def min_sum(arr):
    arr.sort(reverse=True)
    total = arr[0]
    avg = arr[0]

    for i in range(1, len(arr)):
        if arr[i] < avg:
            break
        total += arr[i]
        avg = (total) / (i + 1)

    return total

n = int(input())
arr = list(map(int, input().split()))

result = min_sum(arr)
print(result)
```

RESULT

5 / 5 Test Cases Passed | 100 %



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

MATHS TEST

Description

Alice has a mathematics test for which she is underprepared. She has to do at least one question correctly to pass the test. He decides to do a question which needs her to find the smallest prime number which is larger than a given integer N. Your task is to find and return an integer value representing the smallest prime number larger than N.

Input Format:**input1:** An integer value N**Output Format:**

Return an integer value representing the smallest prime number larger than N.

Sample Input

6

Sample Output

7

Source Code:

```
def next_prime(N):
    num = N + 1

    while True:

        is_prime = True
        for i in range(2, int(num**0.5) + 1):
            if num % i == 0:
                is_prime = False
                break

        if is_prime:
            return num

    num += 1

N = int(input())

result = next_prime(N)

print(result)
```

RESULT

5 / 5 Test Cases Passed | 100 %



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

MAGIC STRING

Description

Eva has a string S containing lowercase English letters. She wants to transform this string into a Magic String, where all the characters in the string are the same. To do so, she can replace any letter in the string with another letter present in that string.

Your task is to help Eva find and return an integer value, representing the minimum number of steps required to form a Magic String. Return 0, if S is already a Magic String.

Input Specification:

input1: A string S, containing lowercase English letters.

Output Specification:

Return an integer value, representing the minimum number of steps required to form a Magic String. Return 0, if S is already a Magic String.

Sample Input:

aabbbccddd

Sample Output:

8

Source Code:

```
from collections import Counter

def min_steps_to_magic_string(S):
    if len(set(S)) == 1:
        return 0

    freq = Counter(S)

    max_freq = max(freq.values())

    return len(S) - max_freq

S = input()

result = min_steps_to_magic_string(S)
print(result)
```

RESULT

5 / 5 Test Cases Passed | 100 %

KU,
^3CSE0,
, KUB23-
SE012 K
^823CSE
^UB23C
^2KU012^
^3CSE0
^3CSL
^UB23
^KUB23^012
^CSE012
^3CSE0,
^UB23^012
^CSE0 KI,



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

ENCODE THE NUMBER

Description

You work in the message encoding department of a national security agency. Every message that is sent from or received in your office is encoded. You have an integer N, and each digit of N is squared and the squares are concatenated together to encode the original number. Your task is to find and return an integer value representing the encoded value of the number.

input1: An integer value N representing the number to be encoded.

Output :

Return an integer value representing the encoded value of the number.

Sample Input:

167

Sample Output:

13649

Source Code:

```
def encode_number(N):
    str_N = str(N)
    encoded_str = ""

    for digit in str_N:
        squared_digit = int(digit) ** 2 # Square the digit
        encoded_str += str(squared_digit)

    encoded_value = int(encoded_str)

    return encoded_value

# Input reading
N = int(input())

result = encode_number(N)
print(result)
```

RESULT

5 / 5 Test Cases Passed | 100 %



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

EQUILIBRIUM

Description

You are given an array A of N integers. An equilibrium position is a position where the sum of all integers on its left is equal to the sum of all integers on its right in the array A. Print the index of the equilibrium position.

Note:For any given array there is only a single equilibrium position, if no equilibrium position is found then print "NOT FOUND" without quotes.

The array is 1 indexed.

Input Format:

The input consists of two lines:

The first line contains an integer denoting N.

The second line contains N space-separated integers denoting the elements of the array A.

Input will be read from the STDIN by the candidate

Output Format:

Print the index of the equilibrium position. If no index is found, print "NOT FOUND"

Sample Input

```
5
2 4 7 3 3
```

Sample Output

```
3
```

Source Code:

```
def find_equilibrium_position(N, A):
    total_sum = sum(A)
    left_sum = 0

    for i in range(N):
        right_sum = total_sum - left_sum - A[i]

        if left_sum == right_sum:
            return i + 1

        left_sum += A[i]

    return "NOT FOUND"

# Input reading
N = int(input())
A = list(map(int, input().split()))
result = find_equilibrium_position(N, A)
print(result)
```

RESULT

5 / 5 Test Cases Passed | 100 %



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

SIGNATURE FOR LCM

Description

Given two numbers a and b. Find the GCD and LCM of a and b.

Input:

- Two positive integers a and b (1 <= a, b <= 1000)

Output:

For GCD function, an integer representing the GCD of a and b

For LCM function, an integer representing the LCM of a and b

Sample Input:

12 18

Output:

6

36

Explanation:

The GCD of 12 and 18 is 6. The LCM of 12 and 18 is 36.

Source Code:

```
import math

def gcd(a, b):
    return math.gcd(a, b)

def lcm(a, b):
    return (a * b) // gcd(a, b)

# Input reading
a, b = map(int, input().split())

# Calculate GCD and LCM
gcd_value = gcd(a, b)
lcm_value = lcm(a, b)

print(gcd_value)
print(lcm_value)
```

RESULT

5 / 5 Test Cases Passed | 100 %



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

MISSING ALPHABETS

Description

Pangram is a sentence containing every letter in the English alphabet. Given a string, find all characters that are missing from the string, Le., the characters that can make the string a Pangram We need to print output in alphabetic order.

For example,

Input: welcome to geeksforgeeks

Output: abdhijnpquvxyz

Source Code:

```
def missing_characters_to_pangram(input_string):
    # Define the full alphabet
    alphabet = set('abcdefghijklmnopqrstuvwxyz')

    input_chars = set(input_string.lower())

    missing_chars = alphabet - input_chars

    sorted_missing_chars = sorted(missing_chars)

    return ''.join(sorted_missing_chars)

# Input reading
input_string = input()
result = missing_characters_to_pangram(input_string)
print(result)
```

RESULT

5 / 5 Test Cases Passed | 100 %



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

REVERSE THE ORDER OF STRING

Description

You are given a string containing words separated by spaces. Your task is to write a function or program that reverses the order of words in the string.

Sample Input:

Hello World

Sample Output:

World Hello

Source Code:

```
def reverse_words(string):
    words = string.split()
    words.reverse()
    reversed_string = " ".join(words)
    return reversed_string
input_string = input()
reversed_string = reverse_words(input_string)
print(reversed_string)
```

RESULT

5 / 5 Test Cases Passed | 100 %



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

REVERSE THE ORDER OF STRING

Description

You are given a string containing words separated by spaces. Your task is to write a function or program that reverses the order of words in the string.

Sample Input:

Hello World

Sample Output:

World Hello

Source Code:

```
def reverse_words(string):
    words = string.split()
    words.reverse()
    reversed_string = " ".join(words)
    return reversed_string
input_string = input()
reversed_string = reverse_words(input_string)
print(reversed_string)
```

RESULT

5 / 5 Test Cases Passed | 100 %



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

ARDUINO

Description

Tom is an Arduino Programmer. He has designed a program to run his robocar on a horizontal number line. Initially, the car is parked at: 0. Given an array A of N integers which can be A, B, C... the robocar runs as follows as per the designed program

First the robocar moves A units in specified direction(right in case the integer is positive and left if the integer is negative).

Then robocar first moves A units and then B units in a specified direction.

In the next step, the robocar moves A units, B units, and then C units in a specified direction.

This process keeps on repeating as per the number of integers in the sequence..

Your task is to find and return an integer value, representing the farthest coordinate reached by the robocar from the beginning to the end of the process.

Sample Input:

1 -2 3 4

Sample Output:

6

Source Code:

```
def find_farthest_coordinate(arr):
    current_position = 0
    max_distance = 0

    for i in range(len(arr)):
        current_position += arr[i]
        max_distance = max(max_distance, abs(current_position))

    return max_distance
arr = list(map(int,input().split()))
result = find_farthest_coordinate(arr)
print(result)
```

RESULT

5 / 5 Test Cases Passed | 100 %

KUB

SE01

B23C

2KU
SE01

CSE0
JB23

UB23
J212

D12
3C2K



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

CANDIES

Description

Let's consider a scenario where there are K candies to be distributed among N children, each uniquely numbered from 1 to N. The distribution commences with Child A, followed by a sequential allocation to the subsequent children in the order: A, A+1, A+2, ..., N. The query at hand is to identify which child will be the last recipient of a candy. In more explicit terms, after Child x (where $1 \leq x < N$) receives a candy, the subsequent candy is granted to Child x+1. Upon Child N receiving a candy, the distribution cycle restarts, and Child 1 becomes the next recipient. The primary objective is to ascertain the identity of the child who will receive the last candy in this cyclic distribution.

Note: Each child receives only 1 candy.

Input Format:

The first line of input contains 3 space separated integers N, K and A.

Output Format:

Print the friend who will be the final recipient of the candy.

Constraints:

$1 \leq N \leq K \leq 10^8$

Sample Input:

5 2 1

Sample Output:

2

Source Code:

```
def last_candy_recipient(N, K, A):
    last_child = (A - 1 + K - 1) % N + 1
    return last_child

# Example usage:
N, K, A = map(int, input().strip().split())
print(last_candy_recipient(N, K, A))
```

RESULT

6 / 6 Test Cases Passed | 100 %

KUB, SE01, B23C, 12KU, SEO1, CSE0, JB23, UB23, 112K, 012C, 3CK,



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

MINIMUM NUMBER OF KEY PRESSES

Description

George has a setup which includes a special keyboard and a monitor , that initially displays 0. The special keyboard has 11 numeric keys (0,1,2,3,4,5,6,7,8,9,00). If he presses 00, the previously displayed value will be multiplied by 100. Whereas, if he presses any other numeric key, the previously displayed value will be firstly multiplied by 10 and then the number on the key will be added to it

You are given a numeric string S. Your task is to help George find and return an integer value, representing the minimum number of key presses to reach the number.

Input Specification:

input: A numeric string s. representing the final number,

Output Specification:

Return an integer value, representing the minimum number of key presses to reach the number.

Sample Input:

100

Sample Output:

2

Source Code:

```
def min_key_presses(s):
    target = int(s)
    presses = 0

    while target > 0:
        if target % 100 == 0:
            target /= 100
        else:
            target /= 10
            presses += 1

    return presses

# Example usage:
s = input().strip()
print(min_key_presses(s))
```

RESULT

6 / 6 Test Cases Passed | 100 %



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

FINDING COMMAS

Description

Liam works as a data analyst for a company that stores massive amounts of numerical data. He has been tasked with determining how many commas are used when writing numbers in the range of 1 to N (inclusive) in a specific format.

In this format, if numbers are more than four digits long, commas are used to separate the numbers into groups of three, starting from the right for the representation of the number. Your task is to help Liam find and return an integer value, representing the total number of commas used when writing each integer in the range of 1 to N.

Input Specification:

Input: An integer value N. representing the number range.

Output Specification:

Return an integer value, representing total number of commas used when writing each integer in the range of 1 to N.

Sample Input:

5000

Sample Output:

4001

Source Code:

```
def count_commas(N):
    total_commas = 0
    for i in range(1, N+1):
        num_str = str(i)
        total_commas += (len(num_str) - 1) // 3
    return total_commas

# Test the function
N = int(input())
print(count_commas(N))
```

RESULT

5 / 5 Test Cases Passed | 100 %

KUB

SE01

B23C

2KU
SE01

CSE0
JB23

UB23
J2K

D12
3C2K



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

REVERSE PACK

Description

Given an array of positive integers, you need to create a new list where:

Each element represents the frequency count of occurrence of all unique numbers in the original array. Each frequency count occurs the number of times in the new list equal to the value of the corresponding unique number in the original array.

Finally, Sort the new list and display.

Input Format:

The first line contains an integer n, denoting the size of the array.

The second line contains n space-separated integers, representing the elements of the array.

Sample Input:

6

3 3 1 1 1 2

Sample Output:

[1, 1, 2, 2, 2, 3]

Explanation:

[3, 3, 1, 1, 2] we have {3:2, 1:3, 2:1}. So now 2 has to appear 3 times and 3 has to appear 1 time and 1 has to appear 2 times.

So the list we get is [2, 2, 2, 3, 1, 1] sorting the list we have [1, 1, 2, 2, 2, 3]

Source Code:

```
def create_frequency_list(n, arr):
    # Step 1: Count frequencies
    frequency_count = {}
    for num in arr:
        if num in frequency_count:
            frequency_count[num] += 1
        else:
            frequency_count[num] = 1

    # Step 2: Build the new list
    new_list = []
    for num, count in frequency_count.items():
        new_list.extend([count] * num)

    # Step 3: Sort the new list
    new_list.sort()

    return new_list

# Input reading
n = int(input())
arr = list(map(int, input().split()))

# Get the result
result = create_frequency_list(n, arr)

# Print the output
print(result)
```

RESULT

5 / 5 Test Cases Passed | 100 %



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

TRIO

Description**Problem Statement:**

Imagine you have three magical crystals, each with a number on it: X, Y, and Z.

You're on a magical adventure where you pick two crystals from a table. Your goal is to find the highest sum you can get by choosing two of these magical crystals, using their numbers to create the most powerful combination.

Input Format:

The first line of input contains three space-separated integers X, Y and Z.

Output Format:

Print the highest possible sum of the numerals inscribed on the chosen two crystals.

Constraints:

$1 \leq X, Y, Z \leq 10^8$

Sample Input:

1 2 3

Sample Output:

5

Source Code:

```
def highest_sum(X, Y, Z):
    # Calculate the sums of the pairs
    sum1 = X + Y
    sum2 = X + Z
    sum3 = Y + Z

    # Find the maximum sum
    max_sum = max(sum1, sum2, sum3)

    return max_sum

# Input reading
X, Y, Z = map(int, input().strip().split())

# Get the result
result = highest_sum(X, Y, Z)

# Print the output
print(result)
```

RESULT

5 / 5 Test Cases Passed | 100 %



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

SPECIAL FIBONACCI

Description

Alex is exploring a series and she came across a special series, in which

$$f(N) = f(N-1)*f(N-1) + f(N-2)*f(N-2) \bmod 47$$

where $f(0) = 1$. $f(1) = 1$

Your task is to help Alex find and return an integer value, representing the Nth number in this special series.

Input Specification:

input1: An integer value N.

Output Specification:

Return an integer value, representing the Nth number in this special fibonacci series.

Sample Input:

4

Sample Output:

29

Source Code:

```
def special_series(N):
    MOD = 47

    # Base cases
    if N == 0:
        return 1
    if N == 1:
        return 1

    # Create an array to store results up to N
    f = [0] * (N + 1)
    f[0] = 1
    f[1] = 1

    # Fill the array using the defined recurrence
    for i in range(2, N + 1):
        f[i] = (f[i - 1] * f[i - 1] + f[i - 2] * f[i - 2]) % MOD

    return f[N]

# Input reading
N = int(input().strip())

# Get the result
result = special_series(N)

# Print the output
print(result)
```

RESULT

5 / 5 Test Cases Passed | 100 %



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

PYRAMID SUM

Description

Adam has a pyramid of numbers. The pyramid structure is formed by arranging the numbers in the following pattern

1

212

32123

4321234

543212345

N____212____N

The first row contains the number 1. The second row contains the numbers 2, 1, and 2. The third row contains the numbers 3, 2, 1, 2, and 3. This pattern continues for subsequent rows, until it reaches N, which represents the height of the pyramid.

Given height of pyramid N find the sum of the numbers in the pyramid and return the sum as the output.

Sample Input:

3

Sample Output:

17

Explanation:

1

212

32123

$$\text{sum} = 1 + 2 + 1 + 2 + 3 + 2 + 1 + 2 + 3 = 17$$

Source Code:

```
def pyramid_sum(N):
    total_sum = 0

    for i in range(1, N + 1):
        # Calculate the sum of the i-th row
        row_sum = 0

        # Descending part
        for j in range(i, 0, -1):
            row_sum += j

        # Ascending part
        for j in range(2, i + 1):
            row_sum += j

        total_sum += row_sum

    return total_sum

# Input reading
N = int(input().strip())

# Get the result
result = pyramid_sum(N)

# Print the output
print(result)
```

RESULT

5 / 5 Test Cases Passed | 100 %



STUDENT REPORT

DETAILS

Name

APOORVA ASHOK TEGGINAMANI

Roll Number

KUB23CSE012

EXPERIMENT

Title

PREFIX-SUFFIX BALANCE

Description

You are given an array A of N integers. The array A can be divided into two parts: the first part consists of the first 'i' elements of A (where i ranges from 1 to N), and the second part consists of the last (N-i) elements of A.

Your task is to find and return a new array named result of the same size as A, where each element of result[i] represents the absolute difference between the sum of the elements in the first part of A and the sum of the elements in the second part of A.

Note: For $i = N, N-i=0$. So, consider the sum of last $N-i$ integers as 0 in this case.

Input Specifications:

input1: An integer value representing the size of the array A.

input2: An integer array A.

Output Specification:

Return a new integer array named result of the same size as A, where each element of result[i] represents the absolute difference between the sum of the elements in the first part A and the sum of the elements in the second part of A.

Sample Input:

5

1 2 3 4 5

Sample Output:

[13, 9, 3, 5, 15]

Source Code:

```
def calculate_absolute_differences(N, A):
    result = [0] * N # Initialize result array
    total_sum = sum(A) # Total sum of the array

    current_sum = 0 # To keep track of the sum of the first part

    for i in range(N):
        current_sum += A[i] # Sum of the first part up to index i
        # The second part is total_sum - current_sum
        second_part_sum = total_sum - current_sum

        # Calculate the absolute difference
        result[i] = abs(current_sum - second_part_sum)

    return result

# Input reading
N = int(input().strip())
A = list(map(int, input().strip().split()))

# Get the result
result = calculate_absolute_differences(N, A)

# Print the output
print(result)
```

RESULT

5 / 5 Test Cases Passed | 100 %