Let's learn about list comprehensions! You are given three integers  and  representing the dimensions of a cuboid along with an integer . Print a list of all possible coordinates given by  on a 3D grid where the sum of  is not equal to . Here, . Please use list comprehensions rather than multiple loops, as a learning exercise.

**Example**

All permutations of  are:

Print an array of the elements that do not sum to .

**Input Format**

Four integers  and , each on a separate line.

**Constraints**

Print the list in lexicographic increasing order.

**Sample Input 0**

1

1

1

2

**Sample Output 0**

[[0, 0, 0], [0, 0, 1], [0, 1, 0], [1, 0, 0], [1, 1, 1]]

Solution:

if \_\_name\_\_ == '\_\_main\_\_':

    x = int(raw\_input())

    y = int(raw\_input())

    z = int(raw\_input())

    n = int(raw\_input())

    numbers1=[]

    for i in range(x+1):

        for j in range(y+1):

            for k in range(z+1):

                if (i+j+k!=n):

                    numbers1.append([i,j,k])

    print(numbers1)

2-

Given the participants' score sheet for your University Sports Day, you are required to find the runner-up score. You are given  scores. Store them in a list and find the score of the runner-up.

**Input Format**

The first line contains . The second line contains an array   of  integers each separated by a space.

**Constraints**

**Output Format**

Print the runner-up score.

**Sample Input 0**

5

2 3 6 6 5

**Sample Output 0**

5

**Explanation 0**

Given list is . The maximum score is , second maximum is . Hence, we print  as the runner-up score.

Solution:

if \_\_name\_\_ == '\_\_main\_\_':

    n = int(raw\_input())

    arr = map(int, raw\_input().split())

    arr.sort()

    arr2=[]

    last= arr[-1]

    for x in arr:

        if(x!=last):

            arr2.append(x)

    print(arr2[-1])