**Q.1 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.**

**Ans:**

    x = int(input())

    y = int(input())

    z = int(input())

    n = int(input())

list1 = [[i,j,k] for i in range(x+1) for j in range(y+1) for k in range(z+1) if i+j+k != n]

print(list1)

**Q.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.**

**Ans:**

    n = int(input())

    arr = map(int, input().split())

    print(sorted(set(arr), reverse=True)[1])

**Q.3. You are given the firstname and lastname of a person on two different lines. Your task is to read them and print the following:**

**Hello firstname lastname! You just delved into python.**

**Function Description**

**Complete the *print\_full\_name* function in the editor below.**

***print\_full\_name* has the following parameters:**

* ***string first:* the first name**
* ***string last:* the last name**

**Prints**

* ***string:* 'Hello  ! You just delved into python' where  and  are replaced with  and .**

**Input Format**

**The first line contains the first name, and the second line contains the last name.**

**Ans:**

def print\_full\_name(first, last):

    print ("Hello", first, last + "! You just delved into python.")

if \_\_name\_\_ == '\_\_main\_\_':

    first\_name = input()

    last\_name = input()

    print\_full\_name(first\_name, last\_name)

**Q.4. We have seen that lists are mutable (they can be changed), and tuples are immutable (they cannot be changed).**

**Let's try to understand this with an example.**

**You are given an immutable string, and you want to make changes to it.**

**Ans:**

def mutate\_string(string, position, character):

    return string[:position] + character + string[(position + 1):]

s = input()

i, c = input().split()

s\_new = mutate\_string(s, int(i), c)

print(s\_new)

**Q.5In this challenge, the user enters a string and a substring. You have to print the number of times that the substring occurs in the given string. String traversal will take place from left to right, not from right to left.**

**NOTE: String letters are case-sensitive.**

**Input Format**

**The first line of input contains the original string. The next line contains the substring.**

**Constraints**

**Each character in the string is an *ascii* character.**

**Ans:**

def count\_substring(string, sub\_string):

    counter = 0

    for i,c in enumerate(string):

        if i + len(sub\_string) > len(string):

            break

        if string[i:i+len(sub\_string)] == sub\_string:

            counter += 1

    return counter

string = input().strip()

sub\_string = input().strip()

count = count\_substring(string, sub\_string)

print(count)