List Comprehensions

Problem | Submissions | Leaderboard | Editorial | Tutorial

Let's learn about list comprehensions! You are given three integers x, y and z representing the dimensions of a cuboid along with an integer n. Print a list of all possible coordinates given by (i, j, k) on a 3D grid where the sum of $i + j + k$ is not equal to n. Here, $0 \leq i \leq x; 0 \leq j \leq y; 0 \leq k \leq z$. Please use list comprehensions rather than multiple loops, as a learning exercise.

Example

$x = 1$

$y = 1$

$z = 2$

$n = 3$

All permutations of $[i, j, k]$ are:

$[[0, 0, 0], [0, 0, 1], [0, 0, 2], [0, 1, 0], [0, 1, 1], [0, 1, 2], [1, 0, 0], [1, 0, 1], [1, 0, 2], [1, 1, 0], [1, 1, 1], [1, 1, 2]]$.

Print an array of the elements that do not sum to $n = 3$.

$[[0, 0, 0], [0, 0, 1], [0, 0, 2], [0, 1, 0], [0, 1, 1], [1, 0, 0], [1, 0, 1], [1, 1, 0], [1, 1, 2]]$

Input Format

Four integers x, y, z and n, 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]]
```

Explanation 0

Each variable x, y and z will have values of 0 or 1. All permutations of lists in the form $[i, j, k] = [[0, 0, 0], [0, 0, 1], [0, 1, 0], [0, 1, 1], [1, 0, 0], [1, 0, 1], [1, 1, 0], [1, 1, 1]]$. Remove all arrays that sum to $n = 2$ to leave only the valid permutations.

Sample Input 1

```
2
2
```

```
2
2
```

Sample Output 1

```
[[0, 0, 0], [0, 0, 1], [0, 1, 0], [0, 1, 2], [0, 2, 1], [0, 2, 2], [1, 0, 0], [1, 0, 2], [1, 1, 1], [1, 1, 2], [1, 2, 0], [1, 2, 1], [1, 2, 2], [2, 0, 1], [2, 0, 2], [2, 1, 0], [2, 1, 1], [2, 1, 2], [2, 2, 0], [2, 2, 1], [2, 
```

Change Theme Language Pypy 3

```python
# Link - https://www.hackerrank.com/challenges/list-comprehensions/problem?isFullScreen=false

if __name__ == '__main__':

    x_value = int(input())
    y_value = int(input())
    z_value = int(input())
    n_value = int(input())

    my_array = []

    for i in range(0, x_value + 1):
        for j in range(0, y_value + 1):
            for k in range(0, z_value + 1):
                if i + j + k != n_value:
                    my_array.append([i, j, k])

    my_array.sort()
    print(my_array)

```

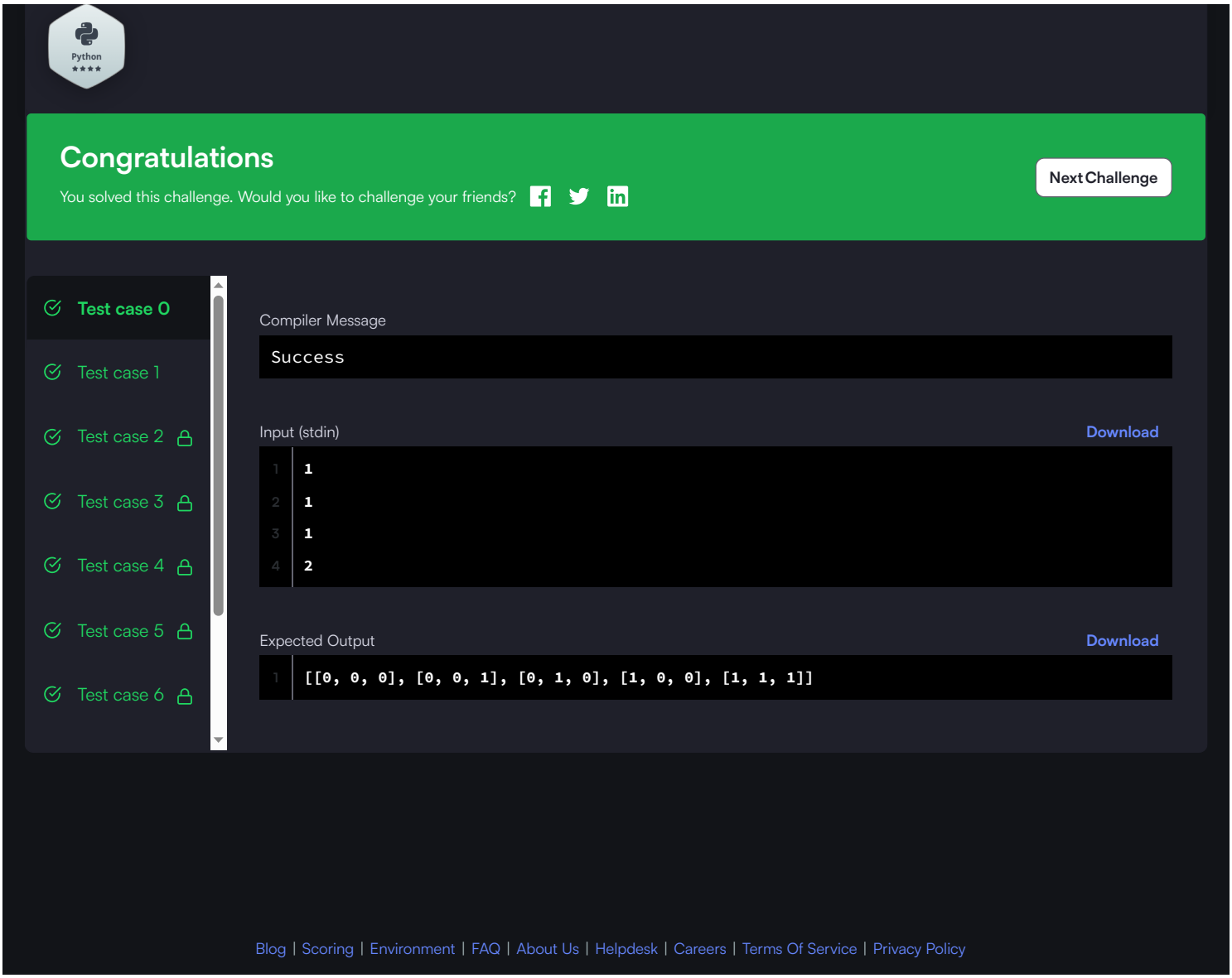
Line: 20 Col: 1

Upload Code as File Test against custom input Run Code Submit Code

You have earned 10.00 points!

You are now 155 points away from the gold level for your python badge.

14% 245/400


Python
Congratulations
You solved this challenge. Would you like to challenge your friends?
Next Challenge
Test case 0
Test case 1
Test case 2
Test case 3
Test case 4
Test case 5
Test case 6
Compiler Message
Success
Input (stdin)
Download
1
1
1
2
Expected Output
Download
[[0, 0, 0], [0, 0, 1], [0, 1, 0], [1, 0, 0], [1, 1, 1]]
Blog | Scoring | Environment | FAQ | About Us | Helpdesk | Careers | Terms Of Service | Privacy Policy