List Comprehensions

Problem	Submissions

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, $0 \le i \le x; 0 \le j \le y; 0 \le k \le z$. 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]]
```
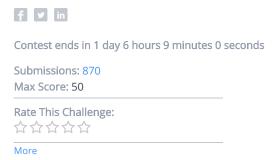
Explanation 0

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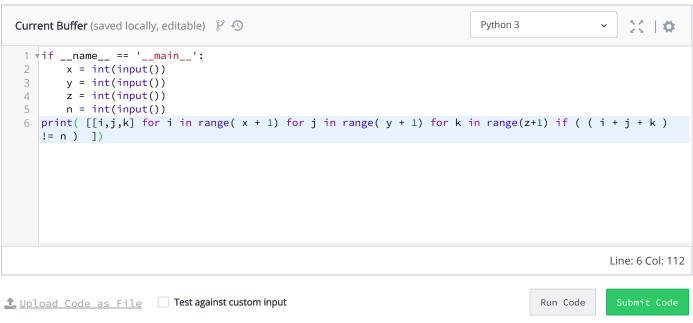Remove all arrays that sum to $n = 2$ to leave only the valid permutations.

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, 2, 2]]
```

Contest ends in 1 day 6 hours 9 minutes 0 seconds

Submissions: 870
Max Score: 50

Rate This Challenge:
☆ ☆ ☆ ☆ ☆

More

Current Buffer (saved locally, editable) ⑂ ↻

Python 3 ⌄   ⛶ | ⚙

```python
if __name__ == '__main__':
    x = int(input())
    y = int(input())
    z = int(input())
    n = int(input())
    print( [[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 )  ])
```

Line: 6 Col: 112

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