# Problem 3

Write function called **fill\_the\_box** which receives a different number of arguments representing:

* the **height** of a box
* the **length** of a box
* the **width** of a box
* n-times different count of cubes with exact **size 1 x 1 x 1**
* a string **"Finish"**

You should fill the box with the given cubes **until the current argument is equal to "Finish"**.

***Note: Submit only the function in the judge system***

### Input

* There will be **no input**, just parameters passed to your function

### Output

### The function should return a string in the following format:

### If, at the end, there is free space left in the box, print:

**"There is free space in the box. You could put {free space in cubes} more cubes."**

### If there is no free space in the box, print:

**"No more free space! You have {cubes left} more cubes."**

### Examples

|  |  |  |
| --- | --- | --- |
| **Test Code** | **Output** | **Comment** |
| print(fill\_the\_box(2, 8, 2, 2, 1, 7, 3, 1, 5, "Finish")) | There is free space in the box. You could put 13 more cubes. | The size of the box: 2 \* 8 \* 2 = 32  We put the cubes consistently. At the end there is more free space left. |
| print(fill\_the\_box(5, 5, 2, 40, 11, 7, 3, 1, 5, "Finish")) | No more free space! You have 17 more cubes. | The size of the box: 5 \* 5 \* 2 = 50  We put the cubes consistently. First, we put 40 cubes and there is free space left. Then we try to put 11 cubes, but there is only space for 10.  Cubes left: 1 + 7 + 3 + 1 + 5 = 17 |
| print(fill\_the\_box(10, 10, 10, 40, "Finish", 2, 15, 30)) | There is free space in the box. You could put 960 more cubes. |  |