Problem Statement

You have a **length** x **width** x **height** box, and you want to fill it with cubes. The cubes have sides that are powers of 2 (1x1x1, 2x2x2, 4x4x4, 8x8x8, etc.). You are given a int[] **cubes**, the i-th element of which is the number of 2^i x 2^i x 2^i cubes you have (i is a 0-based index). Return the minimum number of cubes necessary to fill the box, or -1 if it is impossible to do so.

Definition

Class:	FillBox
Method:	minCubes
Parameters:	int, int, int, int[]
Returns:	int
Method signature:	int minCubes(int length, int width, int height, int[] cubes)
(be sure your method is public)	

Constraints

- **length**, **width** and **height** will each be between 1 and 10^6, inclusive.
- **cubes** will contain between 1 and 20 elements, inclusive.
- Each element of **cubes** will be between 0 and 10^6, inclusive.

Examples

0)

```
4
4
8
{10,10,10}
Returns: 2
```

In order to cover the 4x4x8 box we need two 4x4x4 cubes.

1)

```
4
4
8
{10,10,1}
Returns: 9
```

Same case as before but we have only one 4x4x4 cube so we will use eight 2x2x2 cubes

2)

```
10
10
11
{2000}
Returns: 1100
```

We have only 1x1x1 cubes. We will need 1100 of those cubes to cover the whole box.

3)

```
10
10
11
{1099}
Returns: -1
```

We don't have enough 1x1x1 cubes.

4)

```
37
42
59
{143821,14382,1438,143,14,1}
Returns: 5061
```