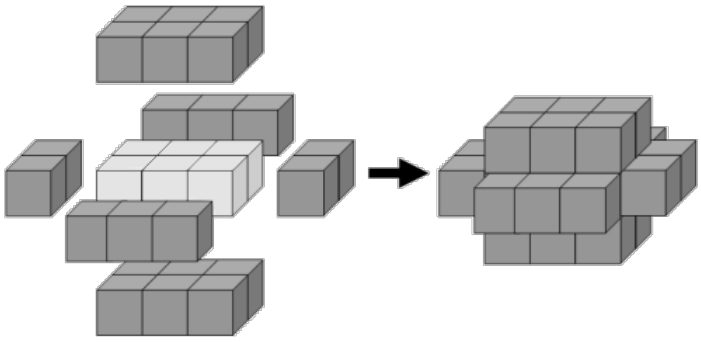
Project Euler #126: Cuboid layers

This problem is a programming version of Problem 126 from projecteuler.net

The minimum number of cubes to cover every visible face on a cuboid measuring $3 \times 2 \times 1$ is twenty-two.



If we then add a second layer to this solid it would require forty-six cubes to cover every visible face, the third layer would require seventy-eight cubes, and the fourth layer would require one-hundred and eighteen cubes to cover every visible face.

However, the first layer on a cuboid measuring $5 \times 1 \times 1$ also requires twenty-two cubes; similarly the first layer on cuboids measuring $5 \times 3 \times 1$, $7 \times 2 \times 1$, and $11 \times 1 \times 1$ all contain forty-six cubes.

We shall define $C(n)$ to represent the number of cuboids that contain n cubes in one of its layers. So $C(22) = 2$, $C(46) = 4$, $C(78) = 5$, $C(118) = 8$ and $C(154) = 10$.

Given n, compute $C(n)$.

Input Format

The first line of input contains T, the number of test cases. Each test case consists of a single line containing a single integer, n.

Constraints

$1 \leq T \leq 50$
$1 \leq n$

For the first few test files worth 25% of the total points:
$n \leq 10000$

For the next few test files worth 25% of the total points:
$n \leq 100000$

For the last few test files worth 50% of the total points:
$n \leq 1000000$

Output Format

For each test case, output a single line containing a single integer, the value $C(n)$.

Sample Input

```
5
22
46
```

```
78
118
154
```

Sample Output

```
2
4
5
8
10
```

Explanation

The sample I/O are mentioned in the problem statement.