Problem E - Magical Gardening

Rin wakes up to find herself in the middle of a mysterious garden. Unlike last time, she is no longer stuck in her virtual world, but is instead stuck in a physical one.

Again, our protagonist is bent on maximizing the beauty of the garden. But unlike last time, she finds there is already plenty of flowers filling the infinite plane on which the garden is located. Specifically, those flower spots occupy exactly all the points on the infinite plane with integer coordinates, i.e $\text{all } (x, y) \text{ such that } x \in \mathbb{Z}, y \in \mathbb{Z}.$, and nothing more. Furthermore, Rin discovers that each time she chants the word "motus" (which is Latin for "movement"), the garden's boundary will shift itself along the x-axis by any amount she desires (this can be a real number).

Now, as she looks around, she realizes that the garden's boundary is a perfect circle centered at the origin $(0, 0)$ with radius $r \in \mathbb{N}^+$. So she decides to move the garden (or not move it, if she wishes) by some amount, so that the number of flower spots within the garden (including those on the boundary) is maximized.

Rin needs guidance from you, the great magician. Help her determine the maximum number of flower spots within the garden after shifting it along the x-axis by some real number.

Input

The first line of the input consist of a single integer t $(1 \leq t \leq 20)$, the number of gardens you are going to deal with.

The next t lines will contain one integer r_i, $1 \leq r_i \leq 5000$, the radius of the garden.

Output

Output a single integer for each test case, the maximum number of flower spots contained in the shifted (or unshifted) garden.

Sample Input

```
3
1
```

```
2
4
```

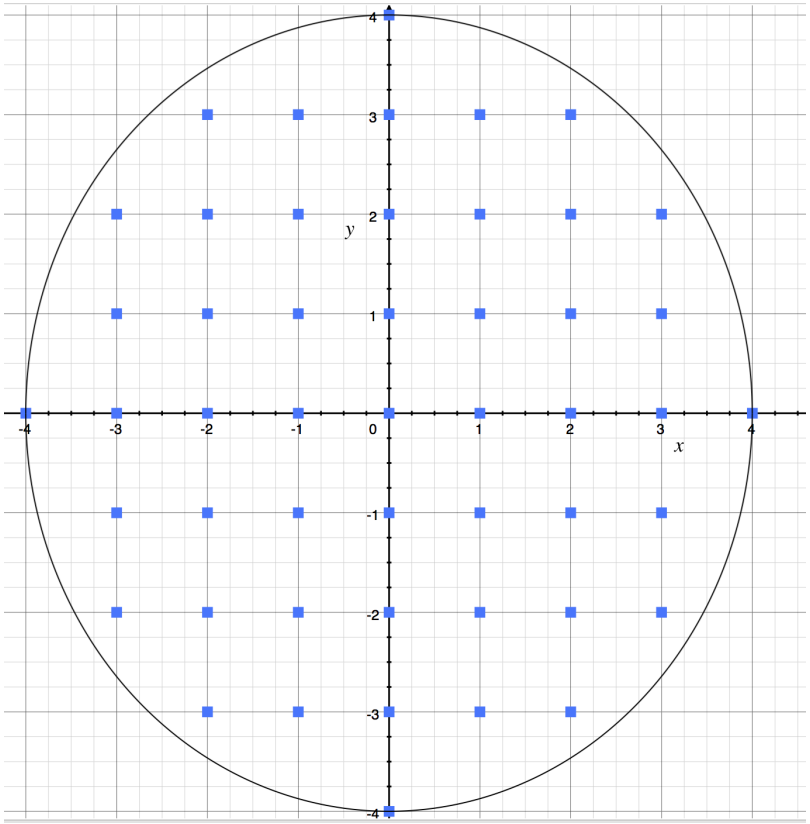
Sample Output

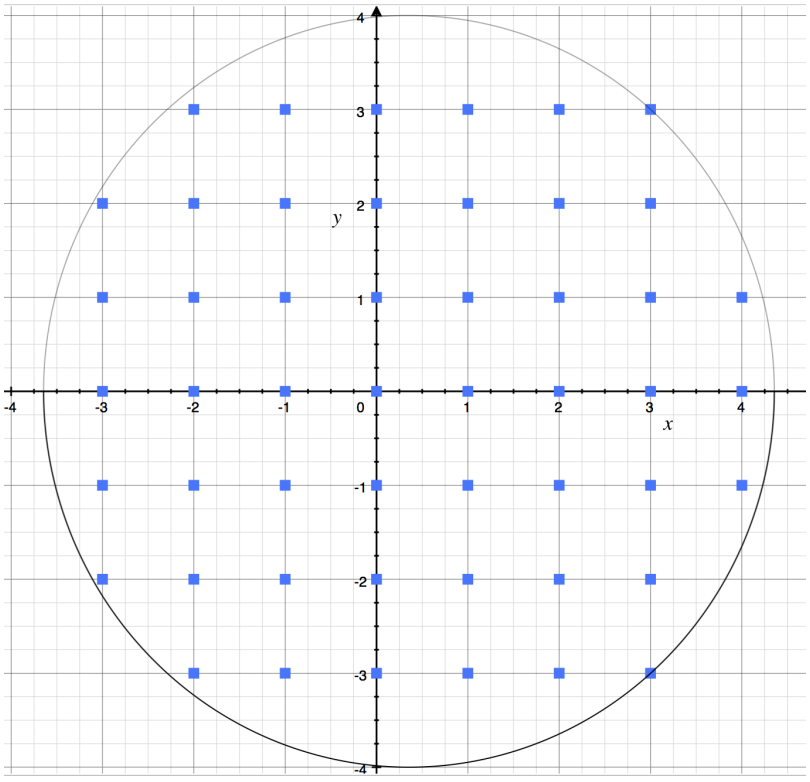
```
5
13
50
```

Sample Explanation

In the first two cases, r = 1 and r = 2, we can easily verify that moving the garden would throw out too many points to be worth the effort, so the best solution is to stay put.

In the third case, moving it by about 0.355 gives the maximum number, as graphed below.


4
3
2
y
1
-4
-3
-2
-1
0
1
2
3
4
x
-1
-2
-3
-4


4
3
2
y
1
-4
-3
-2
-1
0
1
2
3
4
x
-1
-2
-3
-4