

[Advent of Code](#) [\[About\]](#) [\[Events\]](#) [\[Shop\]](#) [\[Settings\]](#) [\[Log Out\]](#) (anonymous user #1056941) 34
[\\$year=2020;](#) [\[Calendar\]](#) [\[AoC++\]](#) [\[Sponsors\]](#) [\[Leaderboard\]](#) [\[Stats\]](#)

--- Day 17: Conway Cubes ---

As your flight slowly drifts through the sky, the Elves at the Mythical Information Bureau at the North Pole contact you. They'd like some help debugging a malfunctioning experimental energy source aboard one of their super-secret imaging satellites.

The experimental energy source is based on cutting-edge technology: a set of Conway Cubes contained in a pocket dimension! When you hear it's having problems, you can't help but agree to take a look.

The pocket dimension contains an infinite 3-dimensional grid. At every integer 3-dimensional coordinate (x,y,z) , there exists a single cube which is either **active** or **inactive**.

In the initial state of the pocket dimension, almost all cubes start **inactive**. The only exception to this is a small flat region of cubes (your puzzle input); the cubes in this region start in the specified **active** (#) or **inactive** (.) state.

The energy source then proceeds to boot up by executing six cycles.

Each cube only ever considers its **neighbors**: any of the 26 other cubes where any of their coordinates differ by at most 1. For example, given the cube at $x=1,y=2,z=3$, its neighbors include the cube at $x=2,y=2,z=2$, the cube at $x=0,y=2,z=3$, and so on.

During a cycle, **all** cubes **simultaneously** change their state according to the following rules:

- If a cube is **active** and exactly **2** or **3** of its neighbors are also active, the cube remains **active**. Otherwise, the cube becomes **inactive**.
- If a cube is **inactive** but exactly **3** of its neighbors are active, the cube becomes **active**. Otherwise, the cube remains **inactive**.

The engineers responsible for this experimental energy source would like you to simulate the pocket dimension and determine what the configuration of cubes should be at the end of the six-cycle boot process.

For example, consider the following initial state:

```
.#.
..#
###
```

Even though the pocket dimension is 3-dimensional, this initial state represents a small 2-dimensional slice of it. (In particular, this initial state defines a 3x3x1 region of the 3-dimensional space.)

Simulating a few cycles from this initial state produces the following configurations, where the result of each cycle is shown layer-by-layer at each given z coordinate (and the frame of view follows the active cells in each cycle):

Before any cycles:

```
z=0
.#.
..#
###
```

After 1 cycle:

```
z=-1
#..
..#
```

Our **sponsors** help make Advent of Code possible:

ING - ING partners up with Advent of Code. Will you join?

```
.#.

z=0
#.#
.##
.#.

z=1
#..
..#
.#.

After 2 cycles:

z=-2
.....
.....
..#..
.....
.....

z=-1
..#..
.#...#
....#
.#...
.....

z=0
##...
##...
#....
....#
.###.

z=1
..#..
.#...#
....#
.#...
.....

z=2
.....
.....
..#..
.....
.....

After 3 cycles:

z=-2
.....
.....
..##...
..###..
.....
.....
.....

z=-1
..#....
...#...
#.....
....##
.#...#.
```

```

..#.#..
...#...

z=0
...#...
.....
#.....
.....
.....##
.##.#..
...#...

z=1
..#....
...#...
#.....
.....##
.#...#.
..#.#..
...#...

z=2
.....
.....
..##...
..###..
.....
.....
.....

```

After the full six-cycle boot process completes, `112` cubes are left in the active state.

Starting with your given initial configuration, simulate six cycles. How many cubes are left in the active state after the sixth cycle?

Your puzzle answer was `426`.

--- Part Two ---

For some reason, your simulated results don't match what the experimental energy source engineers expected. Apparently, the pocket dimension actually has four spatial dimensions, not three.

The pocket dimension contains an infinite 4-dimensional grid. At every integer 4-dimensional coordinate `(x,y,z,w)`, there exists a single cube (really, a hypercube) which is still either active or inactive.

Each cube only ever considers its **neighbors**: any of the 80 other cubes where any of their coordinates differ by at most `1`. For example, given the cube at `x=1,y=2,z=3,w=4`, its neighbors include the cube at `x=2,y=2,z=3,w=3`, the cube at `x=0,y=2,z=3,w=4`, and so on.

The initial state of the pocket dimension still consists of a small flat region of cubes. Furthermore, the same rules for cycle updating still apply: during each cycle, consider the **number of active neighbors** of each cube.

For example, consider the same initial state as in the example above. Even though the pocket dimension is 4-dimensional, this initial state represents a small 2-dimensional slice of it. (In particular, this initial state defines a `3x3x1x1` region of the 4-dimensional space.)

Simulating a few cycles from this initial state produces the following configurations, where the result of each cycle is shown layer-by-layer at each given `z` and `w` coordinate:

Before any cycles:

```
z=0, w=0
```

```
.#.
```

```
..#
```

```
###
```

After 1 cycle:

```
z=-1, w=-1
```

```
#..
```

```
..#
```

```
.#.
```

```
z=0, w=-1
```

```
#..
```

```
..#
```

```
.#.
```

```
z=1, w=-1
```

```
#..
```

```
..#
```

```
.#.
```

```
z=-1, w=0
```

```
#..
```

```
..#
```

```
.#.
```

```
z=0, w=0
```

```
#.#
```

```
.##
```

```
.#.
```

```
z=1, w=0
```

```
#..
```

```
..#
```

```
.#.
```

```
z=-1, w=1
```

```
#..
```

```
..#
```

```
.#.
```

```
z=0, w=1
```

```
#..
```

```
..#
```

```
.#.
```

```
z=1, w=1
```

```
#..
```

```
..#
```

```
.#.
```

After 2 cycles:

```
z=-2, w=-2
```

```
.....
```

```
.....
```

```
..#..
```

```
.....
```

```
.....
```

```
z=-1, w=-2
```

```
.....
```

```
.....
```

```
.....
```

```
.....
```

```
.....

z=0, w=-2
###..
##.##
#...#
.#..#
.###.

z=1, w=-2
.....
.....
.....
.....
.....

z=2, w=-2
.....
.....
..#..
.....
.....

z=-2, w=-1
.....
.....
.....
.....
.....

z=-1, w=-1
.....
.....
.....
.....
.....

z=0, w=-1
.....
.....
.....
.....
.....

z=1, w=-1
.....
.....
.....
.....
.....

z=2, w=-1
.....
.....
.....
.....
.....

z=-2, w=0
###..
##.##
#...#
.#..#
.###.

z=-1, w=0
.....
.....
```

```
.....  
.....  
.....
```

```
z=0, w=0
```

```
.....  
.....  
.....  
.....  
.....
```

```
z=1, w=0
```

```
.....  
.....  
.....  
.....  
.....
```

```
z=2, w=0
```

```
###..  
##.##  
#...#  
.#..#  
.###.
```

```
z=-2, w=1
```

```
.....  
.....  
.....  
.....  
.....
```

```
z=-1, w=1
```

```
.....  
.....  
.....  
.....  
.....
```

```
z=0, w=1
```

```
.....  
.....  
.....  
.....  
.....
```

```
z=1, w=1
```

```
.....  
.....  
.....  
.....  
.....
```

```
z=2, w=1
```

```
.....  
.....  
.....  
.....  
.....
```

```
z=-2, w=2
```

```
.....  
.....  
..#..  
.....  
.....
```

```
z=-1, w=2
```

```
.....
.....
.....
.....
.....

z=0, w=2
###..
##.##
#...#
.#..#
.###.

z=1, w=2
.....
.....
.....
.....
.....

z=2, w=2
.....
.....
..#..
.....
.....
```

After the full six-cycle boot process completes, `848` cubes are left in the `active` state.

Starting with your given initial configuration, simulate six cycles in a 4-dimensional space. How many cubes are left in the active state after the sixth cycle?

Your puzzle answer was `1892`.

Both parts of this puzzle are complete! They provide two gold stars: **

At this point, you should [return to your Advent calendar](#) and try another puzzle.

If you still want to see it, you can [get your puzzle input](#).

You can also [\[Share\]](#) this puzzle.