--- Day 12: The N-Body Problem ---

The space near Jupiter is not a very safe place; you need to be careful of a big distracting red spot, extreme radiation, and a whole lot of moons swirling around. You decide to start by tracking the four largest moons: **Io**, **Europa**, **Ganymede**, and **Callisto**.

After a brief scan, you calculate the **position of each moon** (your puzzle input). You just need to **simulate their motion** so you can avoid them.

Each moon has a 3-dimensional position (`x`, `y`, and `z`) and a 3-dimensional velocity. The position of each moon is given in your scan; the `x`, `y`, and `z` velocity of each moon starts at `0`.

Simulate the motion of the moons in **time steps**. Within each time step, first update the velocity of every moon by applying **gravity**. Then, once all moons' velocities have been updated, update the position of every moon by applying **velocity**. Time progresses by one step once all of the positions are updated.

To apply **gravity**, consider every **pair** of moons. On each axis (`x`, `y`, and `z`), the velocity of each moon changes by **exactly +1 or -1** to pull the moons together. For example, if Ganymede has an `x` position of `3`, and Callisto has a `x` position of `5`, then Ganymede's `x` velocity **changes by +1** (because `5 > 3`) and Callisto's `x` velocity **changes by -1** (because `3 < 5`). However, if the positions on a given axis are the same, the velocity on that axis **does not change** for that pair of moons.

Once all gravity has been applied, apply **velocity**: simply add the velocity of each moon to its own position. For example, if Europa has a position of `x=1, y=2, z=3` and a velocity of `x=-2, y=0,z=3`, then its new position would be `x=-1, y=2, z=6`. This process does not modify the velocity of any moon.

For example, suppose your scan reveals the following positions:

```
<x=-1, y=0, z=2>
<x=2, y=-10, z=-7>
<x=4, y=-8, z=8>
<x=3, y=5, z=-1>
```

Simulating the motion of these moons would produce the following:

```
After 0 steps:
pos=<x=-1, y=  0, z= 2>, vel=<x= 0, y= 0, z= 0>
pos=<x= 2, y=-10, z=-7>, vel=<x= 0, y= 0, z= 0>
pos=<x= 4, y= -8, z= 8>, vel=<x= 0, y= 0, z= 0>
pos=<x= 3, y=  5, z=-1>, vel=<x= 0, y= 0, z= 0>

After 1 step:
pos=<x= 2, y=-1, z= 1>, vel=<x= 3, y=-1, z=-1>
pos=<x= 3, y=-7, z=-4>, vel=<x= 1, y= 3, z= 3>
pos=<x= 1, y=-7, z= 5>, vel=<x=-3, y= 1, z=-3>
pos=<x= 2, y= 2, z= 0>, vel=<x=-1, y=-3, z= 1>

After 2 steps:
pos=<x= 5, y=-3, z=-1>, vel=<x= 3, y=-2, z=-2>
pos=<x= 1, y=-2, z= 2>, vel=<x=-2, y= 5, z= 6>
pos=<x= 1, y=-4, z=-1>, vel=<x= 0, y= 3, z=-6>
pos=<x= 1, y=-4, z= 2>, vel=<x=-1, y=-6, z= 2>

After 3 steps:
pos=<x= 5, y=-6, z=-1>, vel=<x= 0, y=-3, z= 0>
```

Our sponsors help make Advent of Code possible:

ING - Tech is simply at the core of what we do.

```
pos=<x= 0, y= 0, z= 6>, vel=<x=-1, y= 2, z= 4>
pos=<x= 2, y= 1, z=-5>, vel=<x= 1, y= 5, z=-4>
pos=<x= 1, y=-8, z= 2>, vel=<x= 0, y=-4, z= 0>

After 4 steps:
pos=<x= 2, y=-8, z= 0>, vel=<x=-3, y=-2, z= 1>
pos=<x= 2, y= 1, z= 7>, vel=<x= 2, y= 1, z= 1>
pos=<x= 2, y= 3, z=-6>, vel=<x= 0, y= 2, z=-1>
pos=<x= 2, y=-9, z= 1>, vel=<x= 1, y=-1, z=-1>

After 5 steps:
pos=<x=-1, y=-9, z= 2>, vel=<x=-3, y=-1, z= 2>
pos=<x= 4, y= 1, z= 5>, vel=<x= 2, y= 0, z=-2>
pos=<x= 2, y= 2, z=-4>, vel=<x= 0, y=-1, z= 2>
pos=<x= 3, y=-7, z=-1>, vel=<x= 1, y= 2, z=-2>

After 6 steps:
pos=<x=-1, y=-7, z= 3>, vel=<x= 0, y= 2, z= 1>
pos=<x= 3, y= 0, z= 0>, vel=<x=-1, y=-1, z=-5>
pos=<x= 3, y=-2, z= 1>, vel=<x= 1, y=-4, z= 5>
pos=<x= 3, y=-4, z=-2>, vel=<x= 0, y= 3, z=-1>

After 7 steps:
pos=<x= 2, y=-2, z= 1>, vel=<x= 3, y= 5, z=-2>
pos=<x= 1, y=-4, z=-4>, vel=<x=-2, y=-4, z=-4>
pos=<x= 3, y=-7, z= 5>, vel=<x= 0, y=-5, z= 4>
pos=<x= 2, y= 0, z= 0>, vel=<x=-1, y= 4, z= 2>

After 8 steps:
pos=<x= 5, y= 2, z=-2>, vel=<x= 3, y= 4, z=-3>
pos=<x= 2, y=-7, z=-5>, vel=<x= 1, y=-3, z=-1>
pos=<x= 0, y=-9, z= 6>, vel=<x=-3, y=-2, z= 1>
pos=<x= 1, y= 1, z= 3>, vel=<x=-1, y= 1, z= 3>

After 9 steps:
pos=<x= 5, y= 3, z=-4>, vel=<x= 0, y= 1, z=-2>
pos=<x= 2, y=-9, z=-3>, vel=<x= 0, y=-2, z= 2>
pos=<x= 0, y=-8, z= 4>, vel=<x= 0, y= 1, z=-2>
pos=<x= 1, y= 1, z= 5>, vel=<x= 0, y= 0, z= 2>

After 10 steps:
pos=<x= 2, y= 1, z=-3>, vel=<x=-3, y=-2, z= 1>
pos=<x= 1, y=-8, z= 0>, vel=<x=-1, y= 1, z= 3>
pos=<x= 3, y=-6, z= 1>, vel=<x= 3, y= 2, z=-3>
pos=<x= 2, y= 0, z= 4>, vel=<x= 1, y=-1, z=-1>
```

Then, it might help to calculate the **total energy in the system**. The total energy for a single moon is its **potential energy** multiplied by its **kinetic energy**. A moon's **potential energy** is the sum of the absolute values of its `x`, `y`, and `z` position coordinates. A moon's **kinetic energy** is the sum of the absolute values of its velocity coordinates. Below, each line shows the calculations for a moon's potential energy (`pot`), kinetic energy (`kin`), and total energy:

```
Energy after 10 steps:
pot: 2 + 1 + 3 =  6;   kin: 3 + 2 + 1 = 6;   total:  6 * 6 = 36
pot: 1 + 8 + 0 =  9;   kin: 1 + 1 + 3 = 5;   total:  9 * 5 = 45
pot: 3 + 6 + 1 = 10;   kin: 3 + 2 + 3 = 8;   total: 10 * 8 = 80
pot: 2 + 0 + 4 =  6;   kin: 1 + 1 + 1 = 3;   total:  6 * 3 = 18
Sum of total energy: 36 + 45 + 80 + 18 = 179
```

In the above example, adding together the total energy for all moons after 10 steps produces the total energy in the system, `179`.

Here's a second example:

```
<x=-8, y=-10, z=0>
<x=5, y=5, z=10>
<x=2, y=-7, z=3>
<x=9, y=-8, z=-3>
```

Every ten steps of simulation for 100 steps produces:

```
After 0 steps:
pos=<x= -8, y=-10, z=  0>, vel=<x=  0, y=  0, z=  0>
pos=<x=  5, y=  5, z= 10>, vel=<x=  0, y=  0, z=  0>
pos=<x=  2, y= -7, z=  3>, vel=<x=  0, y=  0, z=  0>
pos=<x=  9, y= -8, z= -3>, vel=<x=  0, y=  0, z=  0>

After 10 steps:
pos=<x= -9, y=-10, z=  1>, vel=<x= -2, y= -2, z= -1>
pos=<x=  4, y= 10, z=  9>, vel=<x= -3, y=  7, z= -2>
pos=<x=  8, y=-10, z= -3>, vel=<x=  5, y= -1, z= -2>
pos=<x=  5, y=-10, z=  3>, vel=<x=  0, y= -4, z=  5>

After 20 steps:
pos=<x=-10, y=  3, z= -4>, vel=<x= -5, y=  2, z=  0>
pos=<x=  5, y=-25, z=  6>, vel=<x=  1, y=  1, z= -4>
pos=<x= 13, y=  1, z=  1>, vel=<x=  5, y= -2, z=  2>
pos=<x=  0, y=  1, z=  7>, vel=<x= -1, y= -1, z=  2>

After 30 steps:
pos=<x= 15, y= -6, z= -9>, vel=<x= -5, y=  4, z=  0>
pos=<x= -4, y=-11, z=  3>, vel=<x= -3, y=-10, z=  0>
pos=<x=  0, y= -1, z= 11>, vel=<x=  7, y=  4, z=  3>
pos=<x= -3, y= -2, z=  5>, vel=<x=  1, y=  2, z= -3>

After 40 steps:
pos=<x= 14, y=-12, z= -4>, vel=<x= 11, y=  3, z=  0>
pos=<x= -1, y= 18, z=  8>, vel=<x= -5, y=  2, z=  3>
pos=<x= -5, y=-14, z=  8>, vel=<x=  1, y= -2, z=  0>
pos=<x=  0, y=-12, z= -2>, vel=<x= -7, y= -3, z= -3>

After 50 steps:
pos=<x=-23, y=  4, z=  1>, vel=<x= -7, y= -1, z=  2>
pos=<x= 20, y=-31, z= 13>, vel=<x=  5, y=  3, z=  4>
pos=<x= -4, y=  6, z=  1>, vel=<x= -1, y=  1, z= -3>
pos=<x= 15, y=  1, z= -5>, vel=<x=  3, y= -3, z= -3>

After 60 steps:
pos=<x= 36, y=-10, z=  6>, vel=<x=  5, y=  0, z=  3>
pos=<x=-18, y= 10, z=  9>, vel=<x= -3, y= -7, z=  5>
pos=<x=  8, y=-12, z= -3>, vel=<x= -2, y=  1, z= -7>
pos=<x=-18, y= -8, z= -2>, vel=<x=  0, y=  6, z= -1>

After 70 steps:
pos=<x=-33, y= -6, z=  5>, vel=<x= -5, y= -4, z=  7>
pos=<x= 13, y= -9, z=  2>, vel=<x= -2, y= 11, z=  3>
pos=<x= 11, y= -8, z=  2>, vel=<x=  8, y= -6, z= -7>
pos=<x= 17, y=  3, z=  1>, vel=<x= -1, y= -1, z= -3>

After 80 steps:
pos=<x= 30, y= -8, z=  3>, vel=<x=  3, y=  3, z=  0>
pos=<x= -2, y= -4, z=  0>, vel=<x=  4, y=-13, z=  2>
pos=<x=-18, y= -7, z= 15>, vel=<x= -8, y=  2, z= -2>
pos=<x= -2, y= -1, z= -8>, vel=<x=  1, y=  8, z=  0>

After 90 steps:
```

```
pos=<x=-25, y= -1, z=  4>, vel=<x=  1, y= -3, z=  4>
pos=<x=  2, y= -9, z=  0>, vel=<x= -3, y= 13, z= -1>
pos=<x= 32, y= -8, z= 14>, vel=<x=  5, y= -4, z=  6>
pos=<x= -1, y= -2, z= -8>, vel=<x= -3, y= -6, z= -9>

After 100 steps:
pos=<x=  8, y=-12, z= -9>, vel=<x= -7, y=  3, z=  0>
pos=<x= 13, y= 16, z= -3>, vel=<x=  3, y=-11, z= -5>
pos=<x=-29, y=-11, z= -1>, vel=<x= -3, y=  7, z=  4>
pos=<x= 16, y=-13, z= 23>, vel=<x=  7, y=  1, z=  1>

Energy after 100 steps:
pot:  8 + 12 +  9 = 29;   kin: 7 +  3 + 0 = 10;   total: 29 * 10 = 290
pot: 13 + 16 +  3 = 32;   kin: 3 + 11 + 5 = 19;   total: 32 * 19 = 608
pot: 29 + 11 +  1 = 41;   kin: 3 +  7 + 4 = 14;   total: 41 * 14 = 574
pot: 16 + 13 + 23 = 52;   kin: 7 +  1 + 1 =  9;   total: 52 *  9 = 468
Sum of total energy: 290 + 608 + 574 + 468 = 1940
```

What is the total energy in the system after simulating the moons given in your scan for `1000` steps?

Your puzzle answer was `12082`.

--- Part Two ---

All this drifting around in space makes you wonder about the nature of the universe. Does history really repeat itself? You're curious whether the moons will ever return to a previous state.

Determine **the number of steps** that must occur before all of the moons' **positions and velocities** exactly match a previous point in time.

For example, the first example above takes `2772` steps before they exactly match a previous point in time; it eventually returns to the initial state:

```
After 0 steps:
pos=<x= -1, y=  0, z=  2>, vel=<x=  0, y=  0, z=  0>
pos=<x=  2, y=-10, z= -7>, vel=<x=  0, y=  0, z=  0>
pos=<x=  4, y= -8, z=  8>, vel=<x=  0, y=  0, z=  0>
pos=<x=  3, y=  5, z= -1>, vel=<x=  0, y=  0, z=  0>

After 2770 steps:
pos=<x=  2, y= -1, z=  1>, vel=<x= -3, y=  2, z=  2>
pos=<x=  3, y= -7, z= -4>, vel=<x=  2, y= -5, z= -6>
pos=<x=  1, y= -7, z=  5>, vel=<x=  0, y= -3, z=  6>
pos=<x=  2, y=  2, z=  0>, vel=<x=  1, y=  6, z= -2>

After 2771 steps:
pos=<x= -1, y=  0, z=  2>, vel=<x= -3, y=  1, z=  1>
pos=<x=  2, y=-10, z= -7>, vel=<x= -1, y= -3, z= -3>
pos=<x=  4, y= -8, z=  8>, vel=<x=  3, y= -1, z=  3>
pos=<x=  3, y=  5, z= -1>, vel=<x=  1, y=  3, z= -1>

After 2772 steps:
pos=<x= -1, y=  0, z=  2>, vel=<x=  0, y=  0, z=  0>
pos=<x=  2, y=-10, z= -7>, vel=<x=  0, y=  0, z=  0>
pos=<x=  4, y= -8, z=  8>, vel=<x=  0, y=  0, z=  0>
pos=<x=  3, y=  5, z= -1>, vel=<x=  0, y=  0, z=  0>
```

Of course, the universe might last for a **very long time** before repeating. Here's a copy of the second example from above:

```
<x=-8, y=-10, z=0>
<x=5, y=5, z=10>
<x=2, y=-7, z=3>
<x=9, y=-8, z=-3>
```

This set of initial positions takes `4686774924` steps before it repeats a previous state! Clearly, you might need to **find a more efficient way to simulate the universe**.

How many steps does it take to reach the first state that exactly matches a previous state?

Your puzzle answer was `295693702908636`.

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.