

Advent of Code [About] [Events] [Shop] [Settings] [Log Out] Jarek 25\*  
 {:year 2022} [Calendar] [AoC++] [Sponsors] [Leaderboard] [Stats]

--- Day 21: Monkey Math ---

The **monkeys** are back! You're worried they're going to try to steal your stuff again, but it seems like they're just holding their ground and making various monkey noises at you.

Eventually, one of the elephants realizes you don't speak monkey and comes over to interpret. As it turns out, they overheard you talking about trying to find the grove; they can show you a shortcut if you answer their riddle.

Each monkey is given a job: either to yell a specific number or to yell the result of a math operation. All of the number-yelling monkeys know their number from the start; however, the math operation monkeys need to wait for two other monkeys to yell a number, and those two other monkeys might also be waiting on other monkeys.

Your job is to work out the number the monkey named **root** will yell before the monkeys figure it out themselves.

For example:

```
root: pppw + sjmn
dbpl: 5
cczh: sllz + lgvd
zczc: 2
ptdq: humn - dvpt
dvpt: 3
lfqf: 4
humn: 5
ljgn: 2
sjmn: drzm * dbpl
sllz: 4
pppw: cczh / lfqf
lgvd: ljgn * ptdq
drzm: hmdt - zczc
hmdt: 32
```

Each line contains the name of a monkey, a colon, and then the job of that monkey:

- A lone number means the monkey's job is simply to yell that number.
- A job like **aaaa + bbbb** means the monkey waits for monkeys **aaaa** and **bbbb** to yell each of their numbers; the monkey then yells the sum of those two numbers.
- **aaaa - bbbb** means the monkey yells **aaaa**'s number minus **bbbb**'s number.
- Job **aaaa \* bbbb** will yell **aaaa**'s number multiplied by **bbbb**'s number.
- Job **aaaa / bbbb** will yell **aaaa**'s number divided by **bbbb**'s number.

So, in the above example, monkey **drzm** has to wait for monkeys **hmdt** and **zczc** to yell their numbers. Fortunately, both **hmdt** and **zczc** have jobs that involve simply yelling a single number, so they do this immediately: **32** and **2**. Monkey **drzm** can then yell its number by finding **32** minus **2**: **30**.

Then, monkey **sjmn** has one of its numbers (**30**, from monkey **drzm**), and already has its other number, **5**, from **dbpl**. This allows it to yell its own number by finding **30** multiplied by **5**: **150**.

This process continues until **root** yells a number: **152**.

However, your actual situation involves considerably more monkeys. What number will the monkey named **root** yell?

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

**Spotify** - Follow our engineering blog to see how our developers solve complex tech problems, at scale, every day.

Your puzzle answer was `170237589447588`.

The first half of this puzzle is complete! It provides one gold star: ★

--- Part Two ---

Due to some kind of monkey-elephant-human mistranslation, you seem to have misunderstood a few key details about the riddle.

First, you got the wrong job for the monkey named `root`; specifically, you got the wrong math operation. The correct operation for monkey `root` should be `=`, which means that it still listens for two numbers (from the same two monkeys as before), but now checks that the two numbers match.

Second, you got the wrong monkey for the job starting with `humn:`. It isn't a monkey - it's you. Actually, you got the job wrong, too: you need to figure out what number you need to yell so that `root`'s equality check passes. (The number that appears after `humn:` in your input is now irrelevant.)

In the above example, the number you need to yell to pass `root`'s equality test is `301`. (This causes `root` to get the same number, `150`, from both of its monkeys.)

What number do you yell to pass `root`'s equality test?

Answer:  [\[Submit\]](#)

Although it hasn't changed, you can still [get your puzzle input](#).

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