

Advent of Code [About] [Events] [Shop] [Settings] [Log Out] yangh 18*
 sub y(2019) [Calendar] [AoC++] [Sponsors] [Leaderboard] [Stats]

--- Day 7: Amplification Circuit ---

Based on the navigational maps, you're going to need to send more power to your ship's thrusters to reach Santa in time. To do this, you'll need to configure a series of **amplifiers** already installed on the ship.

There are five amplifiers connected in series; each one receives an input signal and produces an output signal. They are connected such that the first amplifier's output leads to the second amplifier's input, the second amplifier's output leads to the third amplifier's input, and so on. The first amplifier's input value is `0`, and the last amplifier's output leads to your ship's thrusters.

```

0-----0 0-----0 0-----0 0-----0 0-----0
0 ->| Amp A |->| Amp B |->| Amp C |->| Amp D |->| Amp E |-> (to thrusters)
0-----0 0-----0 0-----0 0-----0 0-----0

```

The Elves have sent you some **Amplifier Controller Software** (your puzzle input), a program that should run on your **existing Intcode computer**. Each amplifier will need to run a copy of the program.

When a copy of the program starts running on an amplifier, it will first use an input instruction to ask the amplifier for its current **phase setting** (an integer from `0` to `4`). Each phase setting is used **exactly once**, but the Elves can't remember which amplifier needs which phase setting.

The program will then call another input instruction to get the amplifier's input signal, compute the correct output signal, and supply it back to the amplifier with an output instruction. (If the amplifier has not yet received an input signal, it waits until one arrives.)

Your job is to find the **largest output signal** that can be sent to the **thrusters** by trying every possible combination of phase settings on the amplifiers. Make sure that memory is not shared or reused between copies of the program.

For example, suppose you want to try the phase setting sequence `3,1,2,4,0`, which would mean setting amplifier **A** to phase setting `3`, amplifier **B** to setting `1`, **C** to `2`, **D** to `4`, and **E** to `0`. Then, you could determine the output signal that gets sent from amplifier **E** to the thrusters with the following steps:

- Start the copy of the amplifier controller software that will run on amplifier **A**. At its first input instruction, provide it the amplifier's phase setting, `3`. At its second input instruction, provide it the input signal, `0`. After some calculations, it will use an output instruction to indicate the amplifier's output signal.
- Start the software for amplifier **B**. Provide it the phase setting `(1)` and then whatever output signal was produced from amplifier **A**. It will then produce a new output signal destined for amplifier **C**.
- Start the software for amplifier **C**, provide the phase setting `(2)` and the value from amplifier **B**, then collect its output signal.
- Run amplifier **D**'s software, provide the phase setting `(4)` and input value, and collect its output signal.
- Run amplifier **E**'s software, provide the phase setting `(0)` and input value, and collect its output signal.

The final output signal from amplifier **E** would be sent to the thrusters. However, this phase setting sequence may not have been the best one; another sequence might have sent a higher signal to the thrusters.

Here are some example programs:

- Max thruster signal `43210` (from phase setting sequence `4,3,2,1,0`):
`3,15,3,16,1002,16,10,16,1,16,15,15,4,15,99,0,0`
- Max thruster signal `54321` (from phase setting sequence `0,1,2,3,4`):

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

TwilioQuest - Play Advent of Code and earn rad loot in TwilioQuest, a developer RPG for Mac, Windows, and Linux. Learn JavaScript, Python, git, APIs for SMS, VoIP, or WhatsApp, and much more.

Your puzzle answer was 7818398.

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.