

[Advent of Code](#)
[\[About\]](#)
[\[Events\]](#)
[\[Shop\]](#)
[\[Settings\]](#)
[\[Log Out\]](#)
 teolicht 3*
 Int y=2024;
 [\[Calendar\]](#)
[\[AoC++\]](#)
[\[Sponsors\]](#)
[\[Leaderboard\]](#)
[\[Stats\]](#)

--- Day 2: Red-Nosed Reports ---

Fortunately, the first location The Historians want to search isn't a long walk from the Chief Historian's office.

While the Red-Nosed Reindeer nuclear fusion/fission plant appears to contain no sign of the Chief Historian, the engineers there run up to you as soon as they see you. Apparently, they still talk about the time Rudolph was saved through molecular synthesis from a single electron.

They're quick to add that - since you're already here - they'd really appreciate your help analyzing some unusual data from the Red-Nosed reactor. You turn to check if The Historians are waiting for you, but they seem to have already divided into groups that are currently searching every corner of the facility. You offer to help with the unusual data.

The unusual data (your puzzle input) consists of many reports, one report per line. Each report is a list of numbers called levels that are separated by spaces. For example:

```

7 6 4 2 1
1 2 7 8 9
9 7 6 2 1
1 3 2 4 5
8 6 4 4 1
1 3 6 7 9

```

This example data contains six reports each containing five levels.

The engineers are trying to figure out which reports are safe. The Red-Nosed reactor safety systems can only tolerate levels that are either gradually increasing or gradually decreasing. So, a report only counts as safe if both of the following are true:

- The levels are either all increasing or all decreasing.
- Any two adjacent levels differ by at least one and at most three.

In the example above, the reports can be found safe or unsafe by checking those rules:

- 7 6 4 2 1: Safe because the levels are all decreasing by 1 or 2.
- 1 2 7 8 9: Unsafe because 2 7 is an increase of 5.
- 9 7 6 2 1: Unsafe because 6 2 is a decrease of 4.
- 1 3 2 4 5: Unsafe because 1 3 is increasing but 3 2 is decreasing.
- 8 6 4 4 1: Unsafe because 4 4 is neither an increase or a decrease.
- 1 3 6 7 9: Safe because the levels are all increasing by 1, 2, or 3.

So, in this example, 2 reports are safe.

Analyze the unusual data from the engineers. How many reports are safe?

Your puzzle answer was 606.

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

--- Part Two ---

The engineers are surprised by the low number of safe reports until they realize they forgot to tell you about the Problem Dampener.

The Problem Dampener is a reactor-mounted module that lets the reactor safety systems tolerate a single bad level in what would otherwise be a safe report. It's like the bad level never happened!

Now, the same rules apply as before, except if removing a single level from an unsafe report would make it safe, the report instead counts as safe.

More of the above example's reports are now safe:

- 7 6 4 2 1: Safe without removing any level.
- 1 2 7 8 9: Unsafe regardless of which level is removed.
- 9 7 6 2 1: Unsafe regardless of which level is removed.
- 1 3 2 4 5: Safe by removing the second level, 3.
- 8 6 4 4 1: Safe by removing the third level, 4.
- 1 3 6 7 9: Safe without removing any level.

Thanks to the Problem Dampener, 4 reports are actually safe!

Update your analysis by handling situations where the Problem Dampener can

Our sponsors help
make Advent of
Code possible:

Optiver - Ready
to solve puzzles
for a living?
We're hiring C++,
C# and Python
experts to code
sub-nanosecond
trading systems.
Get ready for
problem solving,
continuous
learning and the
freedom to bring
your software
solutions to life

remove a single level from unsafe reports. How many reports are now safe?

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

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

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