

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

--- Day 2: Password Philosophy ---

Your flight departs in a few days from the coastal airport; the easiest way down to the coast from here is via [toboggan](#).

The shopkeeper at the North Pole Toboggan Rental Shop is having a bad day. "Something's wrong with our computers; we can't log in!" You ask if you can take a look.

Their password database seems to be a little corrupted: some of the passwords wouldn't have been allowed by the Official Toboggan Corporate Policy that was in effect when they were chosen.

To try to debug the problem, they have created a list (your puzzle input) of passwords (according to the corrupted database) and the corporate policy when that password was set.

For example, suppose you have the following list:

```
1-3 a: abcde
1-3 b: cdefg
2-9 c: cccccccc
```

Each line gives the password policy and then the password. The password policy indicates the lowest and highest number of times a given letter must appear for the password to be valid. For example, `1-3 a` means that the password must contain `a` at least `1` time and at most `3` times.

In the above example, `2` passwords are valid. The middle password, `cdefg`, is not; it contains no instances of `b`, but needs at least `1`. The first and third passwords are valid: they contain one `a` or nine `c`, both within the limits of their respective policies.

How many passwords are valid according to their policies?

To begin, [get your puzzle input](#).

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

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

Our [sponsors](#) help make Advent of Code possible:

[AuricVault®](#) - Thieves can't steal--what isn't there! Protect your sensitive data. Simplify PCI, PII, CCPA & GDPR compliance. FREE test credentials.