

About this template

First off: This project is fully configured **except** the inputs. That means to set up this project you have to do the following steps:

1. Clone this template
2. create a file `touch input/cookie.txt` and add your session cookie (without the `cookie=` part) you can get the cookie with the help of your developer tools of your website. 3 will fail if you skip this step.
3. Change into the input directory and download all inputs: `cd input && sh ./getalldays.sh`
4. To start visit the advent of code website. Then go into `main/src/days/day...` where `...` represents the day you wanna solve.

How this is structured

`main/src/lib.rs` defines a trait called `Solvable`. This trait has a default method called `solve`, which calls the abstract methods `part_a` and `part_b`.

The methods `expected_a` and `expected_b` should return the expected answers for the test cases.

`parse` is supposed to parse the input file to reasonable data structures.

You don't need to do anything else except implement your solutions instead of the default return `"a"` or `"b"`.

Please note that you can safely change the generics `<String, String>` to any type you want. The only the constrains they have is that the right hand side must implement `std::display::Debug`.

How to get your solution

Simply run `cargo run`. Cargo will run all solutions asynchronously and print them out if everything went the ok.

How to test your solutions

To test the solutions you can run `Cargo test`. To see the printed output run `Cargo test -- --nocapture`.

Write individual tests alongside the `dayNum.rs` file.