Missing Semester

1. Most makefiles provide a target called clean. This isn’t intended to produce a file called clean, but instead to clean up any files that can be re-built by make. Think of it as a way to “undo” all of the build steps. Implement a clean target for the paper.pdf Makefile above. You will have to make the target [phony](https://www.gnu.org/software/make/manual/html_node/Phony-Targets.html). You may find the [git ls-files](https://git-scm.com/docs/git-ls-files) subcommand useful. A number of other very common make targets are listed [here](https://www.gnu.org/software/make/manual/html_node/Standard-Targets.html" \l "Standard-Targets).
2. Take a look at the various ways to specify version requirements for dependencies in [Rust’s build system](https://doc.rust-lang.org/cargo/reference/specifying-dependencies.html). Most package repositories support similar syntax. For each one (caret, tilde, wildcard, comparison, and multiple), try to come up with a use-case in which that particular kind of requirement makes sense.
3. Git can act as a simple CI system all by itself. In .git/hooks inside any git repository, you will find (currently inactive) files that are run as scripts when a particular action happens. Write a [pre-commit](https://git-scm.com/docs/githooks" \l "_pre_commit) hook that runs make paper.pdf and refuses the commit if the make command fails. This should prevent any commit from having an unbuildable version of the paper.
4. Set up a simple auto-published page using [GitHub Pages](https://pages.github.com/). Add a [GitHub Action](https://github.com/features/actions) to the repository to run shellcheck on any shell files in that repository (here is [one way to do it](https://github.com/marketplace/actions/shellcheck)). Check that it works!
5. [Build your own](https://help.github.com/en/actions/automating-your-workflow-with-github-actions/building-actions) GitHub action to run [proselint](http://proselint.com/) or [write-good](https://github.com/btford/write-good) on all the .md files in the repository. Enable it in your repository, and check that it works by filing a pull request with a typo in it.

Semua hasil output ada di folder lecture 8