

I.A. Fichero `~/.gitignore` y primer *commit* de la práctica.

De cara a gestionar las versiones del software con Git, sólo estamos interesados en los ficheros y directorios que realmente forman el código. No tiene sentido gestionar las versiones de los directorios de compilación, porque son directorios que se pueden regenerar automáticamente a partir del código en `src`. Tampoco tiene sentido gestionar la base de datos de evaluación, porque siempre puede recuperarse la original.

Lo que queremos que gestione Git es:

- Los directorios `src`, `scripts` y `docopt_c`, y todo su contenido.
- El fichero `meson.build` y, en general, cualquier fichero que coloquemos en el directorio `PAV/P2`. Por ejemplo: la memoria de la práctica.
- Los propios ficheros de gestión de Git (a continuación crearemos el fichero `.gitignore`, que debería acompañar siempre al paquete de software).

Y, al menos por ahora, lo que queremos que se excluya de la gestión de versiones es cualquier subdirectorio de `PAV/P2` distinto de `src`, `scripts` o `docopt_c`.

Esto puede conseguirse creando un fichero de texto en el directorio del repositorio con el nombre `.gitignore` (repárese en el punto inicial, que provoca que el fichero sea oculto). En este fichero hemos de indicar, con una sintaxis explicada en [.gitignore](#), qué ficheros y directorios queremos excluir de la gestión por parte de Git. También permite estables excepciones a los anteriores, lo cual, a menudo, es casi más interesante.

Para excluir de la gestión Git de todos los directorios salvo los tres mencionados, vamos a construir un fichero `.gitignore` con el contenido siguiente:

```
1  /*/
2  !/docopt_c/
3  !/scripts/
4  !/src/
```

La primera línea de `.gitignore` (`/*/`) indica que no debe realizarse el seguimiento de ningún subdirectorio del proyecto (esto lo indicamos con las barra al inicio y al final); la segunda línea (`!/docopt_c/`) indica que sí debemos seguir el directorio `docopt_c` (lo indicamos con la exclamación inicial; de hecho, esa exclamación indica que un directorio llamado `docopt_c` es una excepción a la regla anterior que excluye todos los directorios). Hacemos lo mismo con los directorios `src` y `scripts`.

A partir de este momento, sólo la parte del proyecto realmente necesaria queda bajo el control de versiones de Git. Cada vez que queramos almacenar una nueva versión, todo lo que tenemos que hacer es repetir los comandos `git add .` y `git commit` con un mensaje explicativo de los cambios incorporados en ella.

Puede eliminarse del control de Git cualquier archivo o directorio incorporado al proyecto antes de escribir el fichero `.gitignore`, por ejemplo el directorio `bin`, con la orden `git rm -rf bin`.

Tareas:

- Escriba el fichero `.gitignore` de manera que Git sólo gestione los directorios `P2/src`, `P2/docopt_c` y `P2/scripts`, y los ficheros del directorio `P2`.
- Realice la primera confirmación (*commit*) con los ficheros originales de la práctica.