

CVS

1. Crear un repositorio CVS nuevo en ~/cvsroot (directorio cvsroot en el directorio de inicio del usuario).

```
mkdir ~/cvsroot
export CVSROOT=~/cvsroot
cvs init
```

2. Crear un directorio vacío dentro del repositorio para contener el primer módulo (que se llamará PRUEBA).

```
mkdir ~/cvsroot/prueba
```

3. Crear un directorio donde contener las copias de trabajo.

```
mkdir ~/copiasTrabajo
```

4. Obtener una copia del módulo PRUEBA, actualmente vacío, y alojarla en el directorio recién creado.

```
cd ~/copiasTrabajo
cvs checkout prueba
```

5. Copiar los archivos fuente de los dos prácticos anteriores en la copia, y declararlos parte del módulo prueba.

```
cvs add holamundo.c prueba.c {promedio,max}{.c,.h} Makefile
```

6. Hacer el commit de los cambios.

```
cvs commit -m "Se ingresa a CVS un módulo de prueba" .
```

7. Agregar alguna sentencia en la función main del archivo prueba.c cerca del *final* de la misma (antes del return), sin hacer commit aún.

8. Crear un directorio para guardar las copias remotas, y extraer el módulo prueba de algún compañero en otra máquina utilizando ssh (en caso de no tener dos máquinas, sacar otra copia del módulo propio).

```
cd ~ ; mkdir copiasRemotas ; cd copiasRemotas
export CVSROOT=<IP>:/home/<usuario>/cvsroot
export CVS_RSH=ssh
cvs checkout prueba
```

9. Agregar alguna sentencia en la función main del archivo prueba.c al *principio* de la misma, y hacer el commit de los cambios.

```
cvs ci -m "Se agregó código al principio de la función main"
```

10. Esperar a que el compañero haga el ejercicio anterior y volver al módulo propio, para hacer el commit de los cambios.

```
cd ~/copiasTrabajo/prueba
export CVSROOT=~/cvsroot
cvs commit -m "Se agregó código al final de la función main"
```

11. Si el commit del ejercicio anterior falló, averiguar qué pasó, hacer *update* de la copia.

```
cvs log prueba.c
cvs update .
```

12. Revisar el estado del archivo prueba.c, y reintentar el commit.