

### 1. Explicació del "programa"

Breument, el nostre programa consta de 4 fitxers .c. Com que l'objectiu és crear un temporitzador, 3 d'ells es diuen segons, minuts i hores. El quart fitxer, anomenat principal, és on es controlen aquets 3 fitxer mencionats.

A part, el programa compte amb l'ajuda de 4 *scripts*: "limpiar", que mata l'execució d'aquests 4 fitxers, iniciar, que inicialitza els comptadors a zero, "iniciar\_temporitzador" que inicia els 3 temporitzadors, "mostrar\_temporitzador" que mostra la informació d'aquest temporitzador en qualsevol instant i finalment "finalitzar\_contador", que atura "naturalment" l'execució del temporitzador.

El flux del programa breument és: principal rep la primera senyal SIGCONT provinent d'executar iniciar.sh. Això farà que principal envii també aquesta senyal a segons, minuts i hores de manera que cadascun d'ells inicialitza els seus comptadors a zero. Originalment, estan inicialitzats a -1.

Principal rebrà llavors una segona senyal SIGCONT provinent d'executar iniciar\_contador.sh. Això provocarà que principal envii una altre senyal SIGCONT a segons, el qual començarà a contar el seu rellotge de "segons". Per fer-ho, ho fem amb la instrucció "alarm(1)", on l'1 indica el temps que ha de passar (en segons) perquè el sistema se'n vagi a la funció que tracta aquesta senyal SIGALRM. Cada vegada que s'incrementa 1 segon al procés segons, aquest envia una senyal SIGCONT a minuts. Minuts, doncs, incrementa el seu comptador de "minuts" cada vegada que rep 60 vegades una senyal SIGCONT. D'altra banda, també envia cada vegada una senyal SIGUSR1 cap a principal, indicant així que el temporitzador de principal incrementi els segons respectivament.

Quan minuts rep conseqüentment 60 senyals SIGCONT del procés "segons", ell és l'encarregat d'enviar 2 senyals: 1 senyal SIGCONT a hores i una senyal SIGUSR1 a principal. En primer lloc, la senyal SIGCONT cap a hores indica a aquest mateix procés que ha passat 1 minut. En conseqüència, hores incrementarà el seu comptador de "hores" cada vegada que rebí 60 vegades la senyal SIGCONT de minuts. En segon lloc, la senyal SIGUSR1 cap a principal és l'ordre que provoca que el temporitzador de principal incrementi els minuts.

El procés es repeteix amb hores: cada vegada que rep una senyal SIGCONT provinent de minuts, incrementa el seu rellotge. Quan aquest adquireix el valor de 59, voldrà dir que ha passat 1 hora, de manera que enviarà una senyal SIGCONT a principal. Aquesta última senyal SIGCONT serà l'ordre que farà que principal incrementi el seu rellotge d'hores.

Per últim, principal conté les funcions per controlar altres tipus de senyals: SIGTERM, per acabar l'execució d'ell mateix i de segons, minuts i senyals i SIGALRM, per mostrar el seu temporitzador.

### 2. Problemes que han sorgit durant la implementació

- a. La funció pause: "pause()" deixa el procés en estat de "pausa", és a dir, a la espera. En primer lloc, nosaltres només la declaràvem dintre del "while(1)". Això provocava que les variables que llegeixen el "pid" dels altres processos no s'executessin, fent impossible així que s'enviessin les senyals. Ho vam arreglar declarant una altre "pause()" després de les declaracions de "signal(senyal, handler)".

- b. Makefile: a nosaltres ens van ensenyar a fer “Makefile’s” a la pràctica 2, però aquest en qüestió era una mica més complicat perquè el fitxer “rw\_pid” que feia les funcions de “read” i “write” s’havia de compilar juntament amb altres fitxers .c però mai s’arribava a compilar, de manera que ho feia tot una mica més complicat en quan a dependències.
- c. Senyal SIGTERM: quan es disposàvem a coordinar la fi dels processos quan es rebés la senyal SIGTERM i per tant imprimir per consola el temporitzador i la hora del sistema, resulta que s’imprimia dues vegades. Després de buscar i rebuscar, el problema era l’ordre dels paràmetres de la funció “kill”: estaven intercanviats.

### 3. Punts dèbils del programa

- a. Col·lapse de senyals: En certes situacions, el programa sofreix un col·lapse de senyals. Per exemple, quan segons envia la senyal SIGTERM de que ha transcorregut el segon 59, l’envia a minuts i ha principal. Minuts, que ja ha rebut 59 senyals provinents de segon, ha d’incrementar el rellotge de minuts, de manera que també envia una senyal SIGUSR1 a principal. El procés principal, doncs, rep a la vegada dos senyals. Segons el que hem llegit, si això succeeix, el propi procés dóna prioritat a una de les dos senyals, de manera que primer gestiona una i després l’altre. Aquesta prioritat provoca un fet bastant important als temporitzadors: gairebé un segon de retard. Es pot veure quan executes “mostrar\_temporizador” quan passa just un minut: la diferència entre la hora del sistema quan el temporitzador ha començat i la hora del sistema que s’ha imprès quan hem executat “mostrar\_temporizador” no coincideix amb el temps transcorregut que indica el temporitzador. Aquest error, si fa o no fa, és mes o menys d’un segon. En conseqüència, podem dir que cada minut el temporitzador s’endarrereix un segon del temps “real”.