



Enviar y capturar

Mediante dos procesos, solo existen dos formas para compartir información. Una es compartir la memoria que se usa (Shared Memory), y la otra es mediante mensajes. Estos mensajes pueden ser signals de un proceso a otro.

Enviar signals

Para enviar signals, usamos el comando kill, de la siguiente forma:

```
kill(int pid, int signum);
```

Donde `pid` es el PID del proceso al que mandar el signal, y `signum` el signal a enviar (Por ejemplo, **SIGUSR1**)

Recibir o capturar signals

Para capturar un signal y ejecutar un tratamiento en cuanto este llegue, hay que definir un sigaction:

```
sigaction(int signum, struct sigaction *trat, struct sigaction *trat_ant)
```

Donde `signum` es el número de la signal a modificar, `trat` es el tratamiento que queremos que tome, y el ultimo parametro `trat_ant` nos permite guardar el tratamiento antiguo (**NULL** si no lo deseamos)

Mas informacion: Struct sigaction.

Orden y posición

Si deseamos que el proceso **A** mande un signal al proceso **B**, el codigo tendra que estar estructurado de la siguiente forma:

|

|

Proceso A (Envia)

```
...  
    kill(pidB, SIGUSR1);  
...
```

Proceso B (Captura)

```
int main(){  
    struct sigaction trat, trat_ant;  
    //Codigo de configuración para trat  
    sigaction(SIGUSR1, &trat, &trat_ant);  
    ...  
}
```