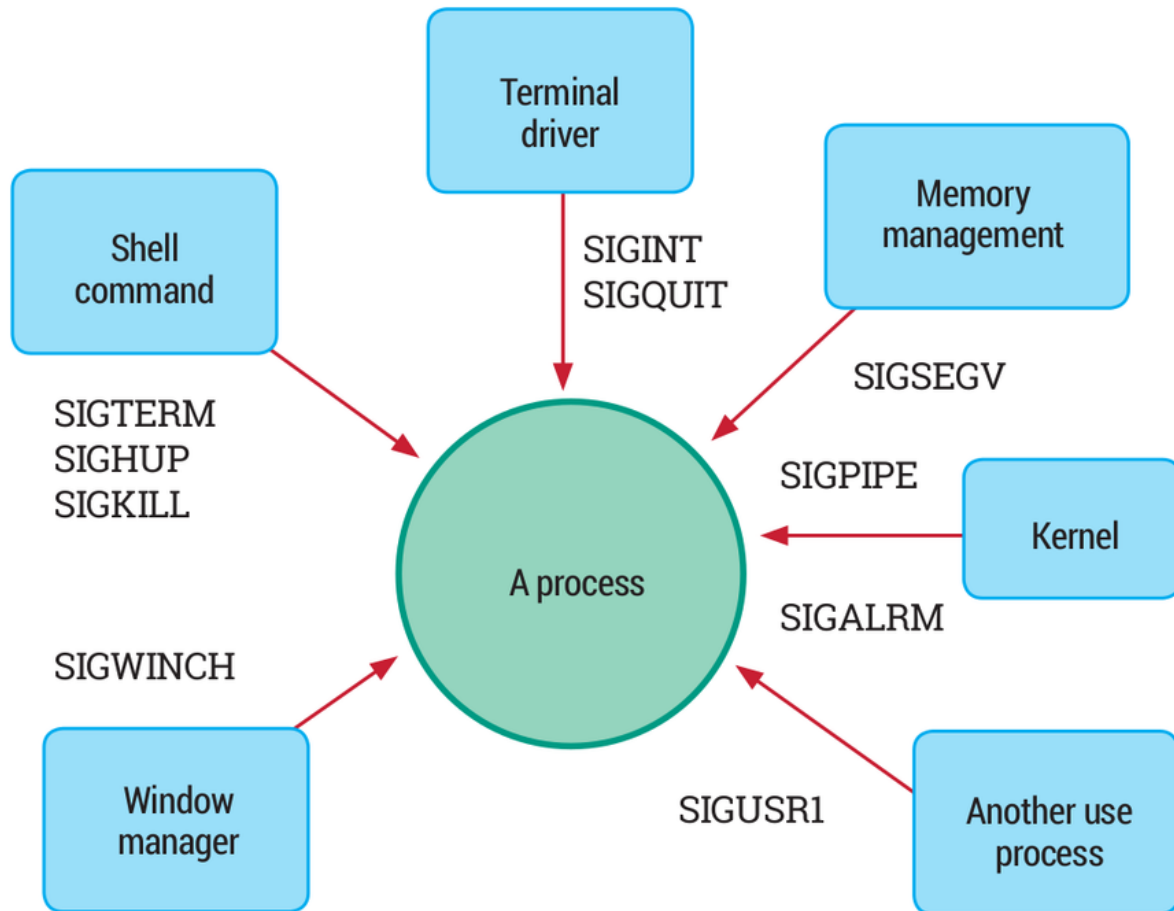


10. Gyakorlat

Processzek közötti kommunikáció: szignálok



Szignál diszpozíció:

A szignál diszpozíció annak megadása, hogy mi történjen, ha egy processz szignált kap. Minden szignálnak van alapértelmezett diszpozíciója, amely meghatározza, hogy a processz milyen módon reagál az adott szignálra. A szignálok alapértelmezett diszpozíciója az alábbiak valamelyike lehet:

- **Term**: a processz terminálódik a beérkező szignál hatására.
- **Ign**: a processz figyelmem kívül hagyja (ignorálja) a beérkező szignált.
- **Core**: a processz terminálódik a szignál hatására, és core fájl készül.
- **Stop**: a szignál hatására a processz futása megáll.
- **Cont**: a szignál hatására a processz futása folytatódik.

Az egyes szignálok diszpozíciója megváltoztatható (vannak kivételek, pl.: *SIGKILL*, *SIGSTOP*). A diszpozíció beállítható olyan módon, hogy a szignál ignorálva legyen, az alapértelmezett diszpozíció legyen érvényben, vagy saját lekezelő függvényt is rendelhetünk hozzá.

Az egyes szignálok diszpozícióját a **signal** függvény segítségével változtathatjuk meg.

A függvény paraméterei: a diszponálható szignál azonosítója, az adott szignál lekezelésének módja (signal handler), ami lehet: *SIG_IGN* (ignorálás), *SIG_DFL* (alapértelmezett), vagy egy saját lekezelő függvény.

A visszatérési érték normál esetben a korábbi signal handler értéke, hiba esetén *SIG_ERR*.

Pl.: `signal(SIGUSR1, myFunction);`

Részletekért, lásd: **man signal**

Szignál küldése:

Erre a célra több függvény is a rendelkezésünkre áll. Vegyük a következőket: **kill**, **raise**, **alarm**.

A **kill** függvény segítségével tetszőleges szignált küldhetünk tetszőleges processznek.

A függvény paraméterei: annak a processznek a PID-je, amelyiknek a szignált küldeni akarjuk, valamint a küldendő szignál azonosítója.

Az első paraméter valójában nem csupán egyszerű PID lehet. Amennyiben 0-t adunk meg, a szignált a hívó processz csoportjába tartozó összes processz megkapja. Ha 1-et adunk meg, akkor minden olyan processz megkapja a szignált, amelynek a hívó processz jogosult szignált küldeni.

Ha a második paraméternek 0-t adunk meg, akkor nem küldődik szignál, azonban a hiba ellenőrzés végbemegy, ilyen módon megállapítható, hogy egy adott PID-del létezik-e processz.

A visszatérési értéke hiba esetén -1, egyébként 0.

Pl.: `int ret = kill(getppid(), SIGUSR1);`

Részletekért, lásd: **man 2 kill**

A **raise** függvény a megadott szignált elküldi a hívó processznek. Gyakorlatilag megfelel a `kill(getpid(), megadott szignál);` hívásnak.

A függvény paramétereként csupán a küldendő szignál azonosítóját kell megadni.

A visszatérési értéke normál esetben 0, egyébként valamilyen nullától különböző érték.

Pl.: `int ret = raise(SIGUSR2);`

Részletekért, lásd: **man raise**

Az **alarm** függvény megadott idő letelte után `SIGALRM` szignált küld a hívó processznek. Ha újból meghívjuk a függvényt, még mielőtt letelt volna az előző által beállított időtartam, az időtartam felülíródik az újonnan megadottal.

A függvény paramétereként azt kell megadni, hogy hány másodperc múlva kell szignált küldeni.

A visszatérési értéke az előző `alarm` hívásból visszamaradt időtartam, vagy 0, ha nem volt ilyen.

Pl.: `int remainingTime = alarm(5);`

Részletekért, lásd: **man alarm**

Várakozás szignálra:

A **pause** függvény addig várakoztatja a hívó processzt, amíg az nem kap valamilyen szignált.

Paraméterei nincsenek, a visszatérési értéke minden esetben -1.

Pl.: `int ret = pause();`

Részletekért, lásd: **man pause**

Feladat 1:

A már jól ismert tippelgetős programunkat írjuk át olyan módon, hogy ezúttal nem a felhasználó adja a tippet, hanem a szülő processz visszajelzései alapján a gyerek processz bináris keresés segítségével megpróbálja magától kitalálni a "gondolt számot". A gyerek a tippjeit pipe-on keresztül kell, hogy küldje a szülőnek. A szülő a visszajelzéseit szignálok formájában kell, hogy "közölje" a gyerekekkel.

Feladat 2:

A már jól ismert tippelgetős programunkat írjuk át olyan módon, hogy ezúttal nem a felhasználó adja a tippet, hanem a szülő processz visszajelzései alapján a gyerek processz bináris keresés segítségével megpróbálja magától kitalálni a "gondolt számot". A gyerek a tippjeit FIFO fájlon keresztül kell, hogy küldje a szülőnek. A szülő a visszajelzéseit szignálok formájában kell, hogy "közölje" a gyerekekkel.