

OPERÁCIÓS RENDSZEREK 1.

---

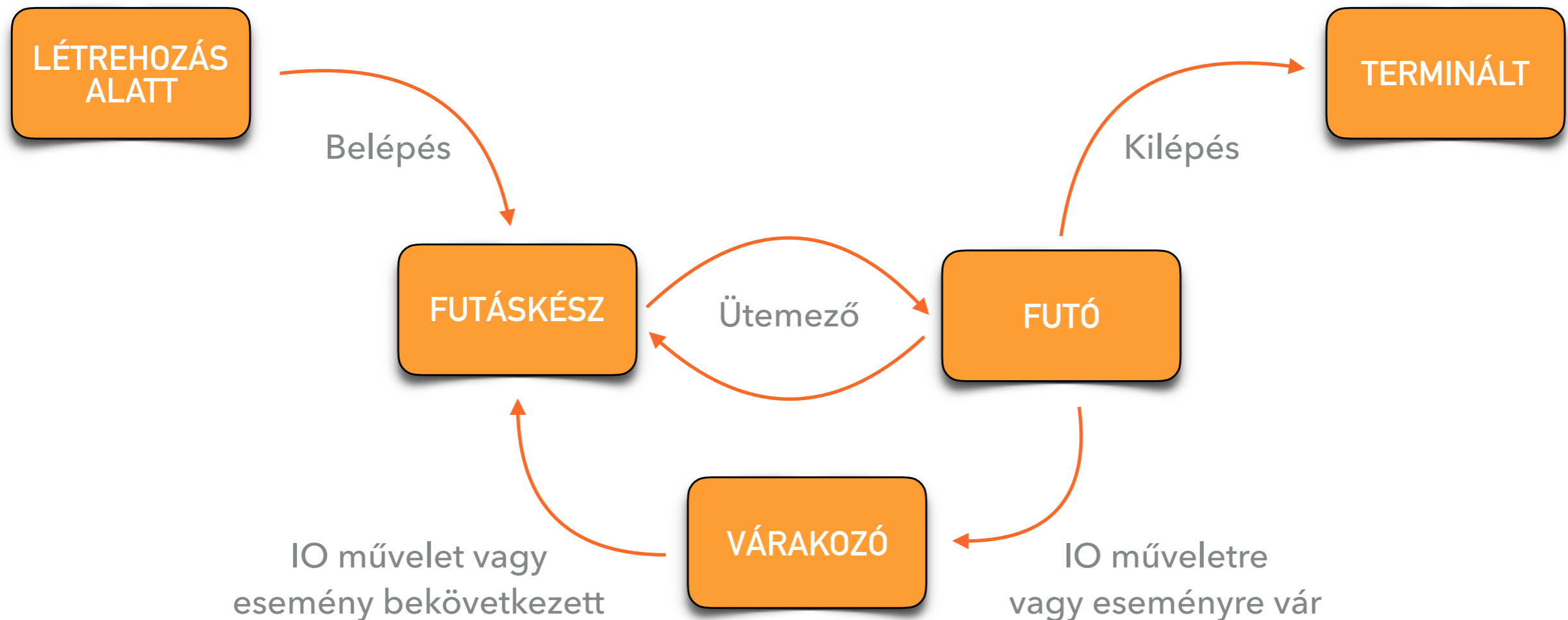
**PROCESSZKEZELÉS**

# A PROCESSZ

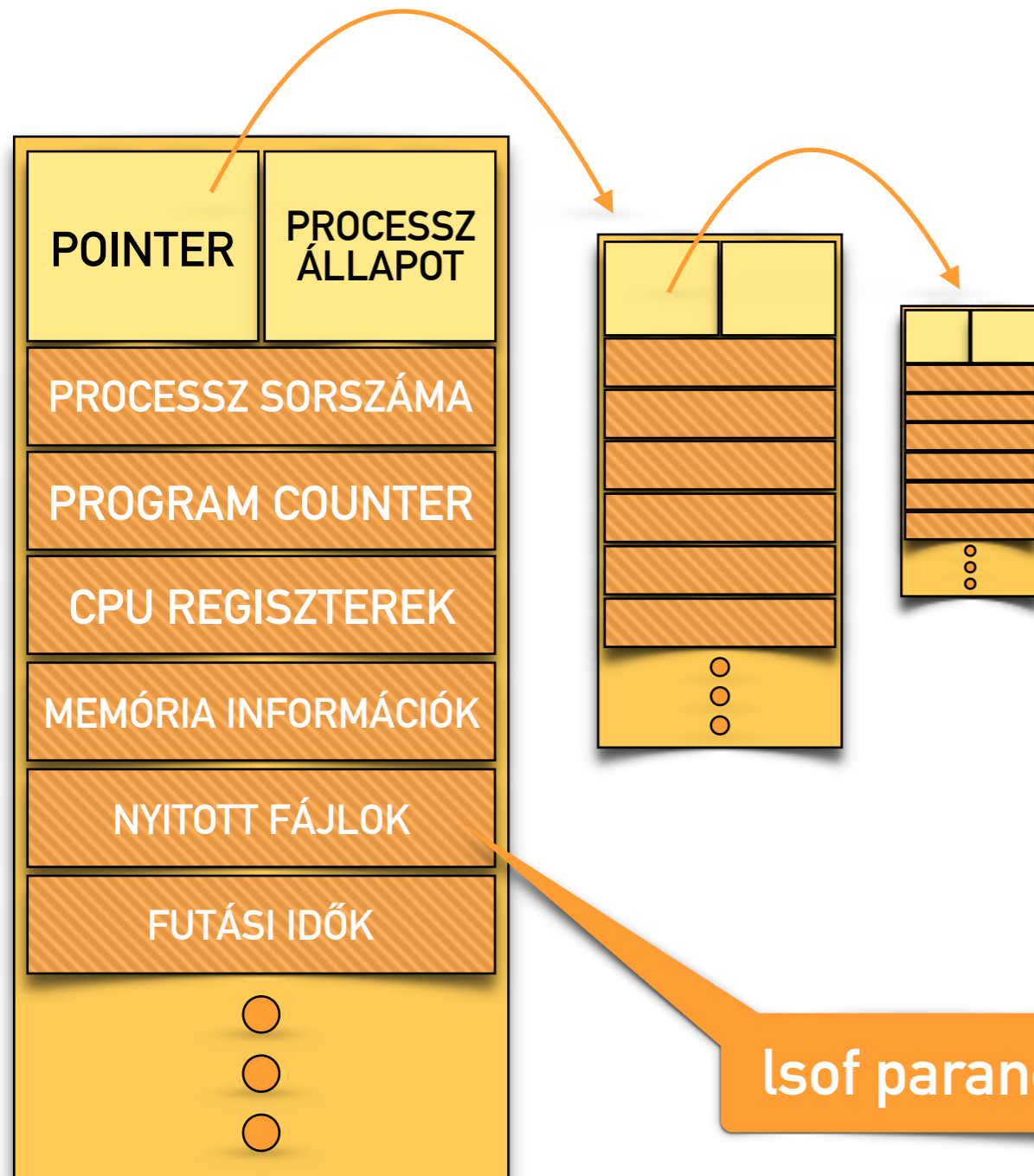
**Program:**  
a végrehajtandó utasítások sorozata

**Processz:**  
a végrehajtás alatt levő program

## ÁLLAPOTOK



# A PROCESS CONTROL BLOCK



**lsuf parancs**

Pointer: egy következő PCB-re mutat.

Processz állapot: új, futáskész, futó, várakozó, terminált.

Processz sorszám: a processz azonosítására szolgáló szám.

Program counter: a soron következő utasításra mutató pointer.

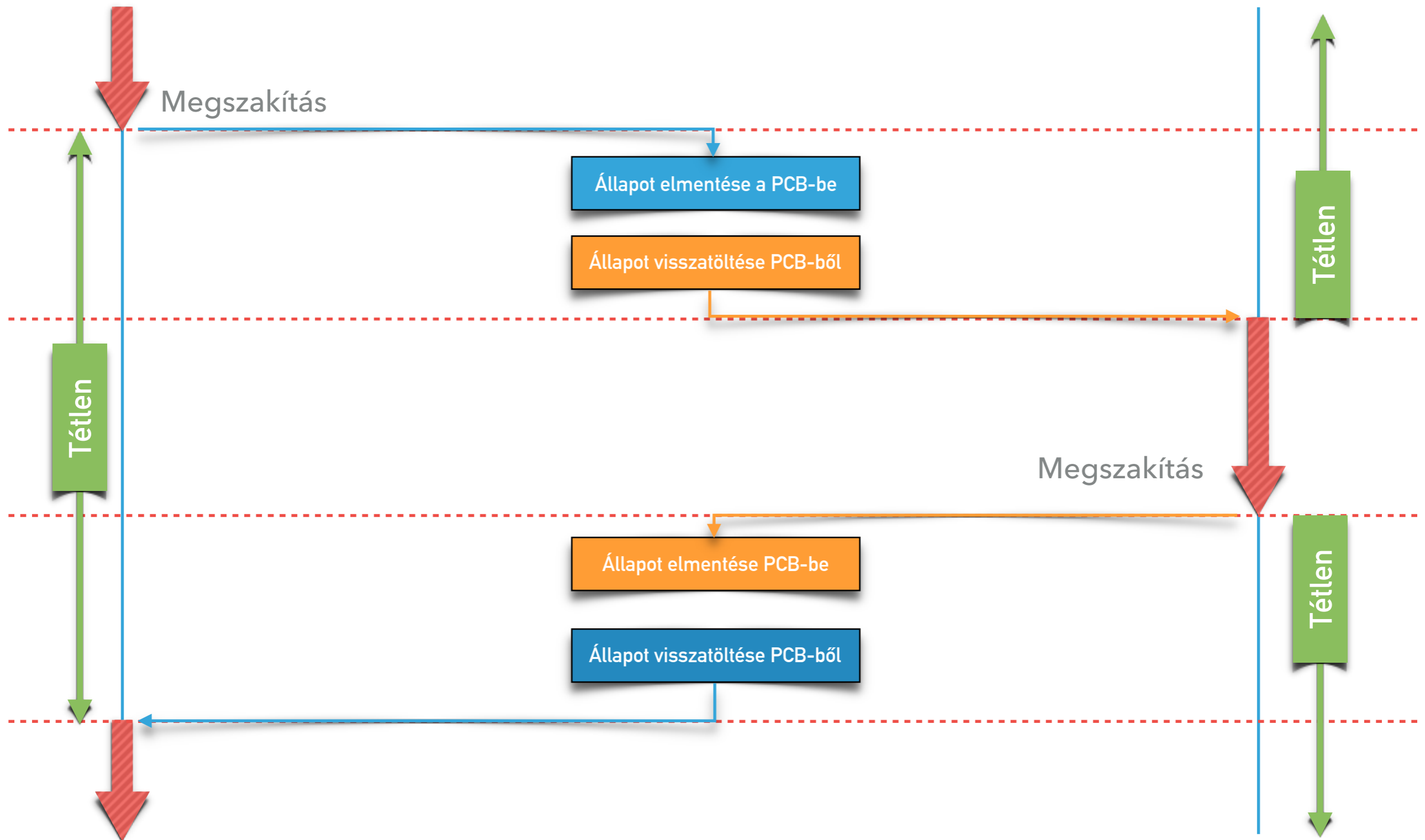
CPU regiszterek: a program futása során beállított CPU regiszterek (akkumulátorok, stack pointerok stb.)

Memória információk: a processz memória felhasználásával kapcsolatos információk.

Nyitott fájlok listája (lsuf parancs).

Futási idők: a processz által elhasznált rendszer idők (time parancs).

# CONTEXT SWITCH



## FIRST COME FIRST SERVE (FCFS)

A legrégebben várakozó processz kerül előre. A processzek egy várakozási listában helyezkednek el, az ütemező a sor elején levőt szolgálja ki.

Hátránya: nem időosztásos eljárás, így egy hosszú futási idejű processz feltartja az őt követőket.

| Process | Érkezési idő | Futási idő |
|---------|--------------|------------|
| P0      | 0            | 5          |
| P1      | 1            | 3          |
| P2      | 2            | 8          |
| P3      | 3            | 6          |

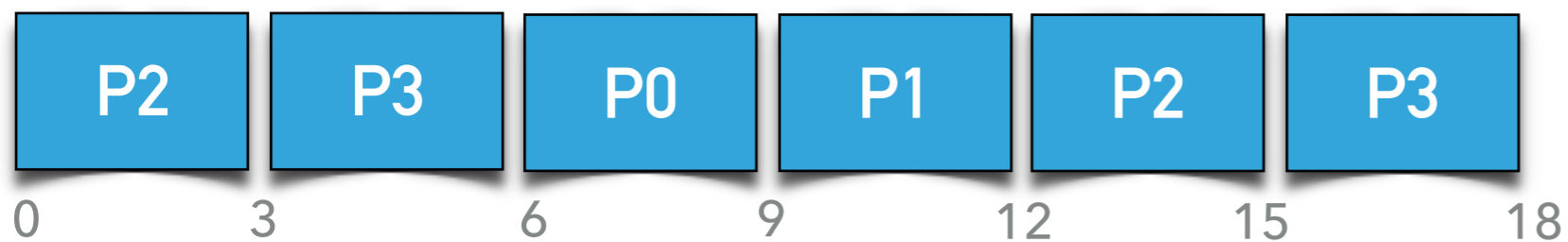


## ROUND ROBIN

Időosztásos eljárás. Minden processz azonos időszeletet kap, ennek neve: quantum.

Hátránya: a prioritizálás hiánya. Túl nagy quantum esetén átmegy FCFS-be, túl rövid esetén nagyon sok időt vesz el a processzek váltásának feladata.

| Process | Érkezési idő | Futási idő |
|---------|--------------|------------|
| P0      | 0            | 5          |
| P1      | 1            | 3          |
| P2      | 2            | 8          |
| P3      | 3            | 6          |



## PRIORITÁSOS ÜTEMEZÉS

Időosztásos eljárás. Minden processzhez egy prioritást is rendel az operációs rendszer, a következő futtatandó processz kiválasztásának ez az alapja.

Azonos prioritású processzek round robin ütemezésben futnak.

| Process | Érkezési idő | Futási idő | Prioritás |
|---------|--------------|------------|-----------|
| P0      | 0            | 5          | 1         |
| P1      | 1            | 3          | 2         |
| P2      | 2            | 8          | 1         |
| P3      | 3            | 6          | 3         |



# PROGRAM INDÍTÁSA

- ▶ A \$PATH környezeti változó
- ▶ Futásidő mérése a time paranccsal.
- ▶ sudo
- ▶ su



# PROGRAM INDÍTÁSA A HÁTTÉRZEN

- ▶ A & jelentése
- ▶ nohup
- ▶ [ctrl][z] / fg

```
feri@columbo:~$ hosszanfuto &  
[1] 32280
```

```
feri@columbo:~$ nohup hosszanfuto &  
[1] 32297
```

# PROCESSZEK ÁLLAPOTA

```
feri@columbo:~$ ps auxu
USER          PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root           1  0.0  0.2 33476   2240 ?        Ss   febr17   0:09 /sbin/init
root           2  0.0  0.0     0     0 ?        S    febr17   0:00 [kthreadd]
root           3  0.0  0.0     0     0 ?        S    febr17   0:04 [ksoftirqd/0]
root           5  0.0  0.0     0     0 ?        S<   febr17   0:00 [kworker/0:0H]
```

- ▶ USER: a folyamat tulajdonosa
- ▶ PID: a folyamat azonosítója
- ▶ %CPU: a processzor használat mértéke
- ▶ %MEM: a memória használat mértéke
- ▶ TTY: a vezérlő terminál azonosítója
- ▶ STAT: a folyamat állapota
- ▶ START: Az indítás ideje
- ▶ TIME: a processz által eddig elhasznált processzor idő
- ▶ CMD: a processz neve

## PROCESSZEK SZERVEZÉSE

```
feri@columbo:~$ pstree
init─┬─acpid
      ├─atd
      ├─cron──cron──sh──check_sys EVERY──sleep
      ├─dbus-daemon
      ├─exim4
      ├─6*[getty]
      ├─inetd
      ├─rsyslogd──3*[{rsyslogd}]
      ├─sshd──sshd──sshd──bash──pstree
      ├─systemd-logind
      ├─systemd-udevd
      ├─upstart-file-br
      ├─upstart-socket-
      ├─upstart-udev-br
      ├─vmttoolsd
      └─whoopsie──2*[{whoopsie}]
```

- ▶ pstree
- ▶ top
- ▶ htop
- ▶ pgrep

```
feri@columbo:~$ pgrep sshd
785
20837
20936
```

# PRIORITÁS KEZELÉSE

## nice

```
feri@columbo:~$ nice 10 számol
```

- ▶ Prioritás -10 és +10 intervallumban
- ▶ Alapértelmezett érték: 0
- ▶ A nagyobb érték alacsonyabb prioritást jelent
- ▶ Negatív értéket csak rendszergazda adhat meg.

## renice

```
root@columbo:~# renice -10 785  
785 (process ID) old priority 0, new priority -10
```

- ▶ Paraméter: processz id
- ▶ Csak a processz tulajdonosa vagy a rendszergazda változtathat az értékén.

# SZIGNÁLOK

A szignál egy állapotjelző, mely a címzett processzben kiváltja a kezelést végző függvény meghívását.

| Szignál             | Hatás                | Leírás  |
|---------------------|----------------------|---|
| SIGHUP (HangUp)     |                      | A processz konfigurációs fájljának újraolvasása                       |
| SIGINT (Interrunpt) | Processz befejezése  | CTRL-C  |
| SIGTERM (Terminate) | Processz befejezése  | Program szabályos leállítása (mentéssel)                              |
| SIGKILL (Kill)      | Processz befejezése  | Program leállítása azonnal (a processz nem tudja befolyásolni)        |
| SIGSTOP (Stop)      | Processz megállítása | Blokkolt állapotba kerül, billentyűzetről: [ctrl][z]                  |
| SIGCONT (Continue)  | Processz indítása    | Futásra kész állapotba kerül, és vár                                  |
| SIGALRM (Alarm)     |                      | Az OS a megadott idő lejártá után küld jelzést a processznek (timer). |

# SZIGNÁLOK KÜLDÉSE

- ▶ kill
- ▶ killall
- ▶ slay

```
feri@columbo:~$ kill -HUP 8192
```

```
feri@columbo:~$ kill 11823
```

```
feri@columbo:~$ kill -9 16384
```

```
feri@columbo:~$ killall -9 mc
```

```
feri@columbo:~$ slay feri
```

## SZIGNÁLOK MEGVALÓSÍTÁSA

```
TMPFILE="/tmp/atmeneti-$$tmp"

trap exit_handler EXIT

exit_handler() {
    echo "Atmeneti fajok torlese"
    rm $TMPFILE
}

touch $TMPFILE
for i in `seq 1 1 1000`; do
    echo $$: $i
    sleep 1
done
```

```
feri@columbo:~$ kill 23222
```

# PROCESSZEK ÜTEMEZETT INDÍTÁSA

## AT

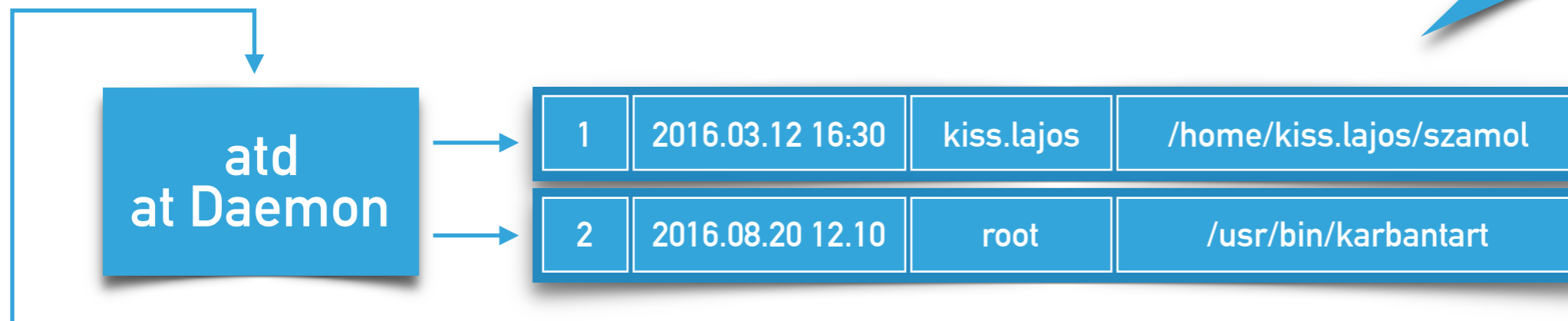
Háttér folyamat időzített indítása. Működését az atd biztosítja.

```
root@columbo:/var/spool/cron# at 12:20
warning: commands will be executed using /bin/sh
at> echo "Hello" >/tmp/hello.txt
<EOT>
job 196 at Wed Mar 9 12:20:00 2016
```

```
root@columbo:~# ls -l /var/spool/cron/atjobs/
összesen 8
-rwx----- 1 root daemon 2382 márc 9 07:11 a000c30172aac8
-rwx----- 1 root daemon 2401 márc 9 07:11 a000c40172aac8
```

A daemon olyan háttérben futó program, amely nem rendelkezik közvetlen kapcsolattal a felhasználó felé.

at  
atq  
atrm





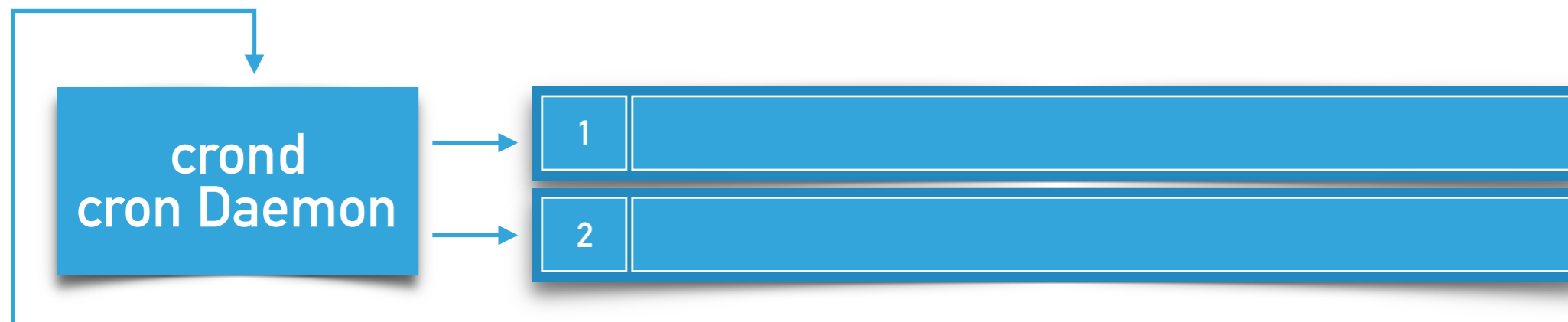
## CRON

Rendszeresen ismétlődő háttérfolyamatok indítására szolgál. Működését a crond biztosítja.

```
root@columbo:~# crontab -e
# m h dom mon dow   command
* * * * * date >>/tmp/hello.txt
```

```
root@columbo:~# ls -l /var/spool/cron/crontabs/
összesen 16
-rw----- 1 feri      crontab 1157 ápr   12  2014 feri
-rw----- 1 FGDYLU    crontab 1135 máj   13  2015 FGDYLU
-rw----- 1 QKIDVD   crontab 1091 máj   14  2015 QKIDVD
-rw----- 1 root      crontab 1122 márc   9  07:30 root
```

/etc/crontab



## A DAEMON/SERVICE

Indításuk, automatikus indulásuk Windows és Unix alatt