# 16. Tétel

A Linux alkalmazások determinisztikus futásidejének érdekében milyen óvintézkedéseket kell megtennünk az alkalmazásunkban? (Valósidejű Linux alkalmazás fejlesztése)

## Valósidejű Linux

* RT-Preemt Kernel patch: a kernel nem tér el nagyban
* az RT-ből adódó különbségeket a kernel kezeli 🡪 **alkalmazásfejlesztésben nincs különbség**
  + a rendszerhívások ugyanazok, csak a lekezelésük más
* **amire figyelni kell: futásidó determinisztikus legyen**
  + valósidejű ütemezést és prioritást kell használni
  + az alkalmazás memória területének végig a fizikai memóriában kell lennie
  + a stack területéből adódó laphibákat meg kell előzni

## Ütemezés és prioritás

* **ütemezés típusai** (a legmagasabb prioritású futásra kész folyamatok közül melyiket válassza ki a scheduler)
  + FIFO
  + Round Robin
* **folyamatokra**
  + #include <sched.h>
  + stratégia kiválasztása + prioritás közös függvényben: int sched\_setscheduler(pid\_t pid, int policy, const struct shed\_param \*param);
    - pid – process id
    - policy – ütemezési stratégia
    - a ssched\_param struct-ban int sched\_priority – prioritás érték
* **szálakra**
  + #include <pthread.h>
  + pthread\_setschedparam(pthread\_t thread, int policy, const struct sched\_param \*param);

## Memória terület fizikai memóriában tárolása

* a folyamat memórialapjai nem kerülhetnek ki a fizikai memóriából 🡨 nem engedhető meg, hogy a folyamat közben töltödjenek be a lapok
* nem megengedhető, hogy ne férjenek be a memóriába a lapok, így az újabbak betöltésénél fel kelljen szabadítani helyet
* folyamat lapjainak zárolása

#include <sys.mman.h>

* + int lockall(int flags);
    - MCL\_CURRENT – a jelenlegi lapokat lockolja
    - MCL\_FUTURE – a jövőbenieket lockolja
  + ajánlott: mlockall(MCL\_CURRENT|MCL\_FUTURE);

## Stack okozta laphibák

* stack növekedése 🡪 új memórialapok allokálása vállik szükségessé 🡪 késleltetés, nem-determinisztikus
* meglőzés: előre allokálás 🡪 ki kell számolni a szükséges területet
* memset(unsigned char tmp[MAX\_STACK], 0, MAX\_STACK);
  + a program elején