

Operációs rendszerek

11. gyakorlat Szemaforok



Problémák csövek használatakor

A cső állapotának lekérdezése nem garantálja az adatérvényességet.

Például, ha két folyamat közel egyidőben lekérdezi, hogy van-e adat a csőben, amiben egyetlen adat van, akkor mindkettő azt az információt kapja, hogy van kiolvasható adat. Ha ezután mindkettő megpróbál olvasni a csőből, az egyik kiolvassa az adatot, a másik pedig nem talál adatot benne.

Problémák osztott memória használatára során



- **Olvasáskor:** Érvényes-e az adat, amit a folyamat felhasználna az osztott memóriaterületről,
 - vagy még nem került oda érvényes adat,
 - vagy már megkezdte a feldolgozását egy másik folyamat?
- **Íráskor:** Szabad a memóriaterület, ahová írna a folyamat,
 - vagy még feldolgozatlan (érvényes) adatot fogunk felülírni,
 - vagy mialatt olvassa ki az adatot egy másik folyamat, módosítjuk azt?

Kritikus szakasz: a folyamat azon szakasza, amikor egy közösen használt erőforráson manipulál. A folyamatok szinkronizálásához meg kell oldani, hogy a folyamat ne lépjen be a kritikus szakaszba, ha

- nem érvényes az információ, amit a kritikus szakaszban olvasni kíván,
- érvényes, még feldolgozásra váró információ van azon a helyen, ahová írni szeretne a kritikus szakaszban,
- másik folyamat manipulál a kritikus szakaszban használandó erőforráson.

További megoldandó problémák



Beléptetés:

- Hely kapacitása adott.
- Új társaság csak akkor léphet be, amikor van megfelelő mennyiségű üres hely.

Termelő-raktár-fogyasztó:

- **Raktár** kapacitása adott.
- **Termelő** raktárra termel. Álljon le átmenetileg a termeléssel, ha a raktár tele van.
- **Fogyasztó** raktárból vételez. Várjon, ha a raktár üres.

Eszköz zárolása

IPC szemafor parancssori műveletek:

- **ipcs**: kilistázza az IPC-ket.
- **ipcrm** -s {szemaforkészlet azonosító}: eltávolítja a szemaforkészletet (pl. beragadt).
- **ipcmk** -S {elemszám}: új szemaforkészletet hoz létre.

IPC parancssori műveletek:

- **ipcrm** -a: eltávolítja az összes üzenetsort, osztott memóriát és szemaforkészletet.

gy6/termelo_fogyaszto_systemv.c

SystemV szemaforkészlet létrehozása:

- **semget**(*kulcs, szemaforok száma, jogosultságok*)
Létrehoz egy *szemaforok száma* darab szemaforból álló szemaforkészletet.

kulcs: key_t típusú (ftok-kal generált vagy IPC_PRIVATE)

jogosultságok:

fájlhozzáférési jogok | IPC_CREAT | IPC_EXCL

Értéke: nem negatív szemaforkészlet azonosító (sem_id), vagy -1 ha hiba.

SystemV szemaforkészlet inicializálása és törlése

- **semctl(*sem_id*, *szemafor*, *parancs*)** vagy
- **semctl(*sem_id*, *szemafor*, *parancs*, *érték*)**
Elvégzi a *sem_id* szemaforkészlet *szemafor* azonosítójú szemaforán a *parancs* által meghatározott műveletet.

parancs:

- IPC_RMID: Törli a szemaforcsaládot, *szemafor* értéke lényegtelen (0), *érték* paraméter nem kell.
- SETVAL: Beállítja a szemafor értékét (*semval*) az *érték* paraméterben megadott értékre.
- ...

Értéke: parancs függő, vagy -1 ha hiba.

SystemV szemaforműveletek végrehajtása



- **semop**(*sem_id*, &*műveletek tömbje*,
műveletek száma)

Sorban végrehajtja a {*műveletek tömbjé*}-ben meghatározott *műveletek száma* darab műveletet a *sem_id* szemaforkészlet tagjain.

műveletek tömbje: egy **sembuf** típusú struktúrák alkotta tömb

Értéke: 0 ha sikeres, vagy -1 ha hiba.

SystemV szemaforműveletek megadása



- **sembuf**: struktúra, adatai:
 - sem_num (unsigned short): a szemafor száma, melyen a művelet elvégzendő,
 - sem_flg (short):
 - IPC_NOWAIT: nem vár a művelet sikeres végrehajtására,
 - SEM_UNDO: a folyamat terminálásakor visszavonásra kerül,
 - 0: vár és termináláskor nem kerül visszavonásra.
 - sem_op (short): a végrehajtandó művelet.



SystemV szemaforműveletek megadása



`sem_op`:

- pozitív: növeli `sem_op` értékével a szemafor értékét (`semval`),
- 0: nullára várakozás művelet. Felfüggeszti a folyamat végrehajtását, míg:
 - a szemafor értéke (`semval`) 0 nem lesz. Ha eleve 0, rögtön folytatja.
 - a szemaforkészletet törlik,
 - szignál érkezik.
- negatív:
 - ha `sem_op` abszolútértéke kisebb vagy egyenlő mint a szemafor értéke (`semval`): levonja `sem_op` értékét a szemafor értékéből és a végrehajtást rögtön folytatja,
 - ha `sem_op` abszolútértéke nagyobb mint a szemafor értéke (negatív lenne a kivonás eredménye) és nincs `IPC_NOWAIT` beállítva, akkor vár, míg:
 - a szemafor értéke nagyobb lesz, mint `sem_op` abszolútértéke, majd levonja azt a szemafor értékéből és folytatja a folyamat végrehajtását,
 - a szemaforkészletet törlik,
 - szignál érkezik,
 - ha `sem_op` abszolútértéke nagyobb mint a szemafor értéke és be van állítva `IPC_NOWAIT` flag, akkor hibát generál.

gy6/pos_szemafor.c, gy6/pek_vasarlo.c

POSIX szemaforok:

- **sem_init**: név nélküli szemafort hoz létre.
- **sem_open**(*nev*, *mód*) vagy
- **sem_open**(*nev*, *mód*, *hozzáférési engedélyek*, *kezdőérték*)
Létrehoz egy nevesített szemafort *kezdőérték* indulóértékkel, vagy kapcsolódik egy már létezőhöz.

nev: char* típusú (pl. "/tmp")

mód:

- O_CREAT,
- O_EXCL,
- ...

hozzáférési engedélyek: fájlhozzáférési engedélyek
(S_IRUSR, S_IWUSR, ...)

Értéke: (sem_t) típusú, a szemafor címe (semaddr), vagy SEM_FAILED ha hiba.

POSIX szemaforok

- **sem_post**: 1-gyel növeli a szemafor értékét (felnyitja, unlock).
- **sem_wait**: 1-gyel csökkenti a szemafor értékét (lezárja, lock). Ha a szemafor értéke 0, akkor vár, míg >0 lesz, és csak akkor dekrementálja.
- **sem_trywait**: működése megegyezik `sem_wait` működésével, de ha 0 a szemafor értéke, nem várakozik, hanem -1 értékkel tér vissza.

POSIX szemaforok

- **sem_timedwait:** működése megegyezik `sem_wait` működésével, de ha 0 a szemafor értéke, akkor legfeljebb *timespec*-ben megadott időpontig vár.
- **sem_destroy:** törli a névtelen szemafort.
- **sem_unlink:** törli a nevesített szemafort.

