

Լաբորատոր աշխատանք 3

Միջարոցեսային հաղորդակցություն (IPC)

Սեմաֆորներ (Semaphores)

Սեմաֆորը ՕՅ միջուկում պահպանվող ամբողջ թիվ է, որի արժեքը կարող է լինել մեծ կամ հավասար 0-ից: Սեմաֆորները հնարավորություն են տալիս պրոցեսներին սինխրոնացնել գործողությունները: System V սեմաֆորները գտնվում են խմբերում, որոնք կոչվում են սեմաֆորների հավաքածու (semaphore set):

Սեմաֆորի կիրառման քայլերն են.

- Ստեղծել կամ բացել սեմաֆորների հավաքածուն `semget()` կանչով
- Սկզբնավորել հավաքածուի մեջ գտնվող սեմաֆորները՝ օգտագործելով `semctl()` կանչը և SETVAL կամ SETALL գործողությունը (այս քայլը պետք է իրականացնի միայն մեկ պրոցես)
- Գործողություններ իրականացնել սեմաֆորի արժեքների վրա՝ կիրառելով **`semop()`** կանչը
- Երբ բոլոր պրոցեսներն ավարտել են սեմաֆորների հավաքածուի հետ աշխատանքը, ջնջել հավաքածուն՝ օգտագործելով **`semctl()`** կանչը և **`IPC_RMID`** գործողությունը (այս քայլը պետք է իրականացնի միայն մեկ պրոցես)

Սեմաֆորների ստեղծումը

`semget()` կանչը ստեղծում է նոր սեմաֆորների հավաքածու կամ ստանում է արդեն գոյություն ունեցող հավաքածուի id-ն: Ֆունկցիայի պրոտոտիպը հետևյալն է.

```
#include <sys/sem.h>
```

```
int semget(key_t key, int nsems, int semflg);
```

- **key** – բանալին է, որը գեներացվում է **`IPC_PRIVATE`** հաստատունի կամ **`ftok()`** ֆունկցիայի միջոցով:
- **nsems** – եթե ստեղծվում է նոր հավաքածու, ապա nsems արգումենտը սահմանում է ստեղծվող հավաքածուի մեջ առկա սեմաֆորների քանակը և պետք է մեծ լինի 0-ից: Եթե `semget()` կանչն օգտագործվում է գոյություն ունեցող հավաքածուի id-ն ստանալու համար, ապա nsems-ը պետք է փոքր կամ հավասար լինի այդ հավաքածուի չափից, այլապես տեղի կունենա **`EINVAL`** error:
- **semflg** – սահմանում է թույլտվության բիթերը (օր.՝ 0600 կամ `S_IRUSR | S_IWUSR`): Բացի թույլտվության բիթերից կարող է ընդունել հետևյալ դրոշակները.
 - **`IPC_CREAT`** – եթե տրված բանալիով սեմաֆորների հավաքածու գոյություն չունի, ապա ստեղծում է նորը,

- **IPC_EXCL** – եթե նշված է **IPC_CREAT** դրոշակը, և տրված բանալիով սեմաֆորների հավաքածու արդեն գոյություն ունի, ապա կանչը ձախողվում է **EEXIST** error-ով:

Հաջող կանչի դեպքում **semget()**-ը վերադարձնում է նոր կամ գոյություն ունեցող հավաքածուի id-ն: Հավաքածուի մեջ սեմաֆորները համարակալվում են 0-ից սկսած:

sem_create.c ծրագիրը **semget()** կանչի միջոցով ստեղծում է նոր սեմաֆորների հավաքածու, որում առկա է մեկ սեմաֆոր, և սկզբնավորում է այն: Ծրագրի կատարվող ֆայլը ստանալու համար հարկավոր է կատարել հետևյալ հրամանը.

```
gcc sem_create.c -o create
```

Ղեկավարման գործողություններ

semctl() կանչը հավաքածուի կամ հավաքածուի մեջ գտնվող կոնկրետ սեմաֆորի համար իրականացնում է տարբեր ղեկավարման գործողություններ: Ֆունկցիայի պրոտոտիպը.

```
int semctl(int semid, int semnum, int cmd, ... /* union semun arg */);
```

- **semid** - հավաքածուի id-ն, որի վրա իրականացվելու է գործողությունը:
- **semnum** - այն գործողությունների համար, որոնք պետք է իրականացվեն հավաքածուի մեջ գտնվող միայն մեկ սեմաֆորի համար, **semnum**-ը ցույց է տալիս հավաքածուի մեջ այդ սեմաֆորի id-ն: Այն գործողությունների համար, որոնք պետք է իրականացվեն ամբողջ հավաքածուի համար, այս արգումենտն անտեսվում է և կարող է սահմանվել որպես 0:
- **cmd** - սահմանում է իրականացվելիք գործողությունը:
- **arg** - այս արգումենտը պահանջվում է միայն որոշ գործողությունների դեպքում: Այն միավորում է, որը պետք է հայտարարվի ծրագրավորողի կողմից:

cmd արգումենտը կարող է ընդունել հետևյալ արժեքներից որևէ մեկը.

- **GETVAL** – **semctl()** ֆունկցիան վերադարձնում է **semid** id-ով սահմանված սեմաֆորների հավաքածուի **semnum**-րդ սեմաֆորի արժեքը:
- **SETVAL** – **semid** id-ով սահմանված սեմաֆորների հավաքածուի **semnum**-րդ սեմաֆորի արժեքը սկզբնավորվում է **arg.val**-ում նշված արժեքով:
- **GETALL** – Ստանում է **semid** id-ով սահմանված սեմաֆորների հավաքածուի բոլոր սեմաֆորների արժեքները և տեղադրում դրանք **arg.array**-ի մեջ: Ծրագրավորողը պետք է ապահովի, որ այդ զանգվածը լինի բավարար չափի (հավաքածուի մեջ սեմաֆորների քանակը կարելի է ստանալ **semid_ds** ստրուկտուրայի **sem_nsems**

դաշտից, իսկ *semid_ds* ստրուկտուրան կարելի է ստանալ *IPC_STAT* գործողությամբ): *semnum* արգումենտն անտեսվում է:

- **SETALL** – Սկզբնավորում է *semid* id-ով սահմանված սեմաֆորների հավաքածուի բոլոր սեմաֆորները, օգտվելով *arg.array* զանգվածում եղած արժեքներից: *semnum* արգումենտն անտեսվում է:
- **GETPID** – Վերադարձնում է տվյալ սեմաֆորի վրա վերջին անգամ *semop()* կանչն իրականացրած պրոցեսի id-ն: Եթե որևէ պրոցես դեռ չի իրականացրել *semop()* կանչ, ապա վերադարձնում է 0:
- **GETNCNT** – Վերադարձնում է այն պրոցեսների քանակը, որոնք այդ պահին սպասում են սեմաֆորի արժեքի բարձրացմանը
- **GETZCNT** - Վերադարձնում է այն պրոցեսների քանակը, որոնք այդ պահին սպասում են սեմաֆորի արժեքի՝ 0-ի հավասարվելուն:

Սեմաֆորների հավաքածուի մասին մանրամասն տեղեկատվության ստացումը ներկայացված է *sem_get_info.c* ծրագրում: Ծրագրի կատարվող ֆայլը ստանալու համար հարկավոր է կատարել հետևյալ հրամանը.

```
gcc sem_get_info.c -o info
```

Սեմաֆորի արժեքի փոփոխումը

semop() համակարգային կանչը *semid* id-ով սահմանված հավաքածուի սեմաֆորների համար իրականացնում է մեկ կամ մի քանի գործողություններ: Ֆունկցիայի պրոտոտիպը հետևյալն է.

```
int semop(int semid, struct sembuf *sops, unsigned int nsops);
```

- **semid** - հավաքածուի id-ն, որի վրա իրականացվելու է գործողությունը:
- **sops** - ցուցիչ է զանգվածի վրա, որը պարունակում է իրականացվելիք գործողությունները:
- **nsops** - զանգվածի տարրերի քանակը:

sops զանգվածի տարրերը հետևյալ տիպի ստրուկտուրաներ են.

```
struct sembuf {  
    unsigned short sem_num; /* Semaphore number */  
    short sem_op; /* Operation to be performed */  
    short sem_flg; /* Operation flags (IPC_NOWAIT and SEM_UNDO) */  
};
```

sem_num դաշտը ցույց է տալիս հավաքածուի այն սեմաֆորի id-ն, որի վրա պետք է իրականացվի գործողությունը: *sem_op* դաշտը ցույց է տալիս իրականացվելիք գործողությունը:

- Եթե *sem_op* > 0, ապա *sem_op*-ի արժեքն ավելացվում է սեմաֆորի արժեքին: Արդյունքում, մյուս պրոցեսները, որոնք սպասում էին սեմաֆորի արժեքը նվազեցնելու համար, կարող են շարունակել աշխատանքը: Կանչող պրոցեսը պետք է ունենա գրելու թույլտվություն:
- Եթե *sem_op* == 0, ապա ստուգվում է, թե արդյոք սեմաֆորի արժեքը հավասար է 0: Եթե հավասար է, ապա գործողությունն անմիջապես ավարտվում է: Հակառակ դեպքում *semop()*-ը արգելափակվում է, մինչև սեմաֆորի արժեքը հավասարվի 0-ի: Կանչող պրոցեսը պետք է ունենա կարդալու թույլտվություն:
- Եթե *sem_op* < 0, ապա սեմաֆորի արժեքը նվազեցվում է *sem_op*-ի արժեքի չափով: Եթե սեմաֆորի ընթացիկ արժեքը մեծ կամ հավասար է *sem_op*-ի բացարձակ արժեքից, ապա գործողությունն անմիջապես ավարտվում է: Հակառակ դեպքում *semop()*-ը արգելափակվում է, մինչև սեմաֆորի արժեքն այնքան մեծանա, որ թույլ տա կատարել գործողությունը՝ առանց բացասական արժեք վերադարձնելու: Կանչող պրոցեսը պետք է ունենա գրելու թույլտվություն:

Սեմաֆորի արժեքի փոփոխությունը ներկայացված է *sem_op.c* ծրագրում: Ծրագրի կատարվող ֆայլը ստանալու համար հարկավոր է կատարել հետևյալ հրամանը.

```
gcc sem_op.c curr_time.c -o op
```

Առաջադրանքներ

1. Ստեղծել նոր սեմաֆորների հավաքածու, որը կպարունակի մեկ սեմաֆոր և կսկզբնավորի այն 2 արժեքով:
2. Իրականացնել գործողություն, որը կփորձի սեմաֆորի արժեքը նվազեցնել 3-ով: Բացատրել ծրագրի աշխատանքի արդյունքը:
3. 2-րդ կետում իրականացրած հրամանի աշխատանքը չընդհատելով, բացել նոր հրամանային տող և ցուցադրել սեմաֆորի մասին տեղեկատվություն: Բացատրել SEMNCNT դաշտի արժեքը:
4. 2-րդ կետում իրականացրած հրամանի աշխատանքը չընդհատելով, բացել նոր հրամանային տող և սեմաֆորի արժեքը մեծացնել 1-ով: Բացատրել 2-րդ հրամանի տողում ցուցադրված հաղորդագրությունը: