Operációs rendszerek BSc

11. gyak.

2021. 04. 28.

Készítette:

Simonyák János Bprof Üzemmérnök-informatikus MZ727W

1) **feladat** – Helyfoglalás

4	Α	В	С	D	Е	F	G	Н	1	J	K	L	М	N	0	P	Q
1	First Fit		Memóriaterület - szabad terület							Nex	t Fit	Memóriaterület - szabad terület					
2	Foglalás	si igény	30	35	15	25	75	45		Foglalá	si igény	30	35	15	.25	75	45
3		39	30	35	15	25		45			39	30	35	15	25	39, 36	45
4		40	30	35	15	25	75	40, 5			40	30	35	15	25	75	40, 5
5		33	30		15	25	75	45			33	30		15	25	75	45
6		20		35	15	25	75	45			20	30	35	15		75	45
7		21	30	35	15	21, 4	75	45			21	30	35	15	25	39, 21, 15	45
8														en a management			
9	Best Fit		Memóriaterület - szabad terület							Wor	st Fit	Memóriaterület - szabad terület					
10	Foglalás	si igény	30	35	15	25	75	45		Foglalá	si igény	30	35	15	25	75	45
11		39	30	35	15	25	75	39, 6			39	30	35	15	25	39, 36	45
12		40	30	35	15	25		45			40	30	35	15	25	75	40, 5
13		33	30		15	25	75	45			33	30	35	15	25	39, 33, 3	45
14		20	30	35	15		75	45			20	30		15	25	75	45
15		21	21, 9	35	15	25	75	45			21	21, 9	35	15	25	75	45

2) **feladat** – Szemafor

a) semset.c

```
D: > Projects > MZ727WOsGyak > MZ727W_0428 > C semset.c > ...
      #include <stdio.h>
      #include <sys/types.h>
      #include <sys/ipc.h>
      #include <sys/sem.h>
      #define SEMKEY 123456L /* kulcs a semget-nek; remelem, egyedi */
          int semid, nsems, semnum, rtn;
          int semflg;
          struct sembuf sembuf, *sop;
          union semun arg;
          int cmd;
11
12
      int main()
15
          nsems = 1; /* egyetlen semaphore a set-ben */
17
          semflg = 00666 | IPC CREAT;
          semid = semget (SEMKEY, nsems, semflg);
          if (semid < 0 ) {perror(" semget hiba"); exit(0);}</pre>
          else printf("\n semid: %d ",semid);
21
          printf ("\n kerem a semval erteket ");
          semnum = 0; /* 0-i semaphort azonositom */
          cmd = SETVAL;
26
          scanf("%d",&arg.val);
          rtn = semctl(semid,semnum, cmd, arg); /* a semid-vel azonositott
29
                               set 0-ik semaphorat ! */
          printf("\n set rtn: %d ,semval: %d ",rtn,arg.val);
          printf("\n");
```

b) semval.c

```
D: > Projects > MZ727WOsGyak > MZ727W_0428 > C semval.c > ...
     #include <stdio.h>
     #include <sys/types.h>
     #include <sys/ipc.h>
      #include <sys/sem.h>
      #define SEMKEY 123456L /* kulcs a semget-nek; remelem, egyedi */
          int semid, nsems, rtn;
          int semflg;
          struct sembuf sembuf, *sop;
11
         union semun arg;
         int cmd;
     int main()
     {
17
          nsems = 1;
          semflg = 00666 | IPC_CREAT;
          semid = semget (SEMKEY, nsems, semflg);
21
          if (semid < 0 ) {perror(" semget hiba"); exit(0);}</pre>
          else printf("\n semid: %d ",semid);
          printf ("\n");
24
          cmd = GETVAL; /* E parancsra a semctl visszaadja a currens
                     semaphor erteket. Itt az rtn-be. */
          rtn = semctl(semid,0, cmd, NULL);
          printf("\n semval: %d ",rtn);
          printf("\n");
```

c) semkill.c

```
D: > Projects > MZ727WOsGyak > MZ727W_0428 > C semkill.c > 😭 main()
     #include <stdio.h>
     #include <sys/types.h>
     #include <sys/ipc.h>
     #include <sys/sem.h>
     #define SEMKEY 123456L /* kulcs a semget-nek; remelem, egyedi */
         int semid, nsems, rtn;
         int semflg;
         struct sembuf sembuf, *sop;
11
         union semun arg;
         int cmd;
12
     int main()
     {
17
         nsems = 1;
         semflg = 00666 | IPC_CREAT;
         semid = semget (SEMKEY, nsems, semflg);
21
         if (semid < 0 ) {perror(" semget hiba"); exit(0);}</pre>
         else printf("\n semid: %d ",semid);
         printf ("\n");
         cmd = IPC RMID; /* Ez a megszuntetes parancsa */
          rtn = semctl(semid,0, cmd, arg);
          printf("\n kill rtn: %d ",rtn);
          printf("\n");
```

d) semup.c

```
D: > Projects > MZ727WOsGyak > MZ727W_0428 > C semup.c > 😭 main()
      #include <stdio.h>
      #include <sys/types.h>
      #include <sys/ipc.h>
      #include <sys/sem.h>
      #define SEMKEY 123456L /* kulcs a semget-nek; remelem, egyedi */
          int semid, nsems, rtn;
          unsigned nsops;
          int semflg;
          struct sembuf sembuf, *sop;
11
12
13
      int main()
14
      {
17
          nsems = 1; /* Egy semaphore legyen */
          semflg = 00666 | IPC CREAT;
          semid = semget (SEMKEY, nsems, semflg);
          if (semid < 0 ) {perror(" semget hiba"); exit(0);}</pre>
21
          else printf("\n semid: %d ",semid);
          printf ("\n");
                         /* Egy operacio van */
          nsops = 1;
          sembuf.sem_num = 0; /* A 0-ik semaphor-ral foglakozunk */
          sembuf.sem_op = 1; /* Inkrementaciot kerunk! */
26
          sembuf.sem_flg = 0666; /* Flag beallitas */
          sop = &sembuf;
                           /* Igy keri a semop az argumentumot */
          rtn = semop(semid, sop, nsops);
          printf("\n up rtn: %d ",rtn);
          printf("\n");
```