

OPERACIJSKI SUSTAVI

3.2.2003.

1. Koristeći sistemske pozive *fork*, *waitpid*, *exit*, *sleep* i *kill* napišite program u kojem proces roditelj kreira dijete i to dijete kreira još jedno dijete ('unuk' početnog procesa). Unuk treba ispisati poruku 'Ja sam unuk' i vrtiti se u beskonačnoj petlji. Nakon 60 sekundi dijete treba uništiti unuka i završiti svoje izvođenje. Roditelj treba pričekati proces dijete da završi i ispisati poruku 'Proces dijete završio'.
2. Zbog radova promet na mostu se odvija posebnom regulacijom. Potrebno je napisati algoritam za kontrolu prometa koji će poštovati sljedeća pravila:
 - a) na mostu se ne smiju u istom trenutku naći dva automobila koji idu u suprotnim smjerovima.
 - b) na mostu može biti više automobila koji idu u istom smjeru.
 - c) moguće je da se automobili koji idu u istom smjeru međusobno pretječu.
 - d) onemogućiti pojavu da kontinuirani protok automobila s jedne strane uzrokuje beskonačno čekanje s druge strane.
3. Uz pretpostavku da su svi pozivi funkcija za stvaranje procesa i rad s zajedničkom memorijom uspješno obavljani, što donji program odsječaj ispisuje te objasnite razlog takvog ispisa.

```
#include<stdio.h>
#include <sys/ipc.h>
#include <sys/shm.h>
#include <sys/sem.h>
main(){
    int SegID;int a=10;int *b;pid_t pid;
    SegID = shmget(IPC_PRIVATE,sizeof(int),0660);
    b = (int*)shmat(SegID,NULL,0);
    *b = 20;
    pid=fork();
    if (pid==0){
        a=a+5;*b=*b+5;
        printf("var1=%d , var2=%d\n", a, *b);
        shmdt(b);
        exit(0);
    }
    else{
        waitpid(pid);
        a=a+5;*b=*b+5;
        printf("var1=%d , var2=%d\n", a, *b);
    }
    shmdt(b);
    shmctl(SegID,IPC_RMID,NULL);
}
```

4. a) Navedite što je Belady-jeva anomalija i pokažite je na primjeru sljedećeg niza zahtjeva : 0 1 2 3 0 1 4 0 1 2 3 4.
b) Za niz zahtjeva pod a) skicirajte rad strategija FIFO, LRU i OPT i navedite koliko promašaja imaju strategije ako sustav ima 3 okvira.
5. Što je fragmentacija spremnika i kako nastaje?

Boris Milašinić