19l-1316 os in lab 9 :

#include <sys/wait.h>

#include <stdlib.h>

#include <assert.h>

#include <stdlib.h>

#include <unistd.h>

#include <sys/ipc.h>

#include <sys/shm.h>

#include <semaphore.h>

int main(int argc, char \*\*argv)

{

int i, nloop=10, \*ptr;

sem\_t mutex;

int shmid2,shmid1;

int SHMSIZE=1024;

sem\_t \*p\_mutex;

if((shmid2 = shmget(IPC\_PRIVATE,SHMSIZE,0666))<0)

{

perror("shmget");

exit(1);

}

p\_mutex = (sem\_t\*) shmat(shmid2,NULL,0);

if(p\_mutex==(sem\_t \*)-1)

{

perror("mutex shmat fails");

exit(0);

}

if(p\_mutex == (sem\_t\*)-1)

{

perror("semaphore initialization");

exit(0);

}

if (fork()==0)

{

sem\_wait(p\_mutex);

for (i=0;i<nloop;i++)

printf("child: %d\n",(\*ptr)++);

sem\_destroy(p\_mutex);

shmctl(shmid2,IPC\_RMID,(struct shmid\_ds\*)0);

shmctl(shmid1,IPC\_RMID,(struct shmid\_ds\*)0);

exit(0);}

}

sem\_post(p\_mutex);

for (i=0;i<nloop;i++)

printf("parent: %d\n",(\*ptr)++);

exit(0);

return 0;

}