

```

1  /*
2  Usar a opção -lrt no gcc
3  Ver shm_overview(7) e mmap(2)
4  */
5
6  #include <sys/types.h>
7  #include <unistd.h>
8  #include <stdlib.h>
9  #include <stdio.h>
10 #include <sys/mman.h>
11 #include <sys/stat.h>      /* For mode constants */
12 #include <fcntl.h>        /* For O_* constants */
13
14 //o tamanho do bloco de memória partilhado.
15 #define SHM_SIZE 1024
16
17 //O tamanho da área de mapeamento de um bloco de memória partilhada (mmap) é
18 //arredondado para o menor múltiplo do tamanho das páginas de memória
19 //igual ou superior ao valor indicado (4KiB na i386) (ver man mmap).
20 //
21 //Neste caso, o valor MMAP_WINDOW será automaticamente arredondado para 4096
22 //(o programa funciona correctamente mesmo com #define MMAP_WINDOW 1)
23 #define MMAP_WINDOW 1024
24
25 #define SHMNAME1 "shm1"
26
27
28 void *shm_com_nome()
29 {
30     void *buf;
31     int fd = shm_open(SHMNAME1, O_CREAT | O_RDWR, S_IRUSR | S_IWUSR);
32     if(fd < 0){
33         perror("sh_open");
34         exit(1);
35     }
36
37     if(ftruncate(fd, SHM_SIZE) < 0) {
38         perror("ftruncate");
39     }
40
41     buf = mmap(NULL, MMAP_WINDOW, PROT_READ | PROT_WRITE, MAP_SHARED, fd, 0);
42     if(buf == NULL){
43         perror("mmap");
44         exit(1);
45     }
46     close(fd);
47
48     return(buf);
49 }
50
51 void *shm_sem_nome()
52 {
53     void *buf;
54     buf = mmap(NULL, MMAP_WINDOW, PROT_READ | PROT_WRITE, MAP_SHARED | MAP_ANON, -1, 0);
55     if(buf == NULL){
56         perror("mmap");
57         exit(1);
58     }
59
60     return buf;
61 }
62
63
64 int main() {
65     char *buf = shm_sem_nome();
66
67     sprintf(buf, "Initial text");
68     printf("%d, %d: %s\n", getpid(), getppid(), buf);
69
70     int pid=fork();
71     if(pid==0){
72         printf("pid = %d, ppid = %d: %s\n", getpid(), getppid(), buf);
73         sleep(4);
74         printf("\npid = %d, ppid = %d: %s\n", getpid(), getppid(), buf);
75     } else {

```

```
76     sleep(2);
77     sprintf(buf, "Parent has exited");
78 }
79
80 printf("pid = %d: exiting\n", getpid());
81 return(0);
82 }
83
84
```