Υποβολή 2ης Άσκησης

Βάσιλας Κωνσταντίνος 5482

Ερώτημα 1

Λίστα τροποποιηθέντων αρχείων

/usr/src/servers/vfs/open.c

Τροποποιήσεις

/usr/src/servers/vfs/open.c

Line	Code
610	printf("New dir -> %s %o", fullpath, dirmode);

Screenshot Αποτελεσμάτων

Ερώτημα 2

Λίστα τροποποιηθέντων αρχείων

/usr/src/kernel/system/do_fork.c

Τροποποιήσεις

/usr/src/kernel/system/do_fork.c

Line	Code
52	printf("Process scheduled\n");

Screenshot Αποτελεσμάτων

Ερώτημα 3

Λίστα τροποποιηθέντων αρχείων

/usr/src/include/minix/callnr.h
/usr/src/servers/pm/table.c
/usr/src/servers/pm/proto.h
/usr/src/servers/pm/utility.c

Τροποποιήσεις

/usr/src/kernel/system/callnr.h

Line	Code
69	#define PROINFO 69

/usr/src/servers/pm/table.c

Line	Code
83	do_proinfo /* 69 = proinfo */

/usr/src/servers/pm/proto.h

Line	Code
50	Int do_proinfo(void);

/usr/src/servers/pm/utility.c

Line	Code
156	```int do_proinfo(void)
{
        struct mproc *rmp;
        clock_t user_time = 0;
        clock_t system_time = 0;
        int count_process = 0;

        for(rmp = &mproc[0] ; rmp < &mproc[NR_PROCS] ; rmp++)
        {
                if(rmp->mp_flags & 1)
                {
                        user_time += rmp->mp_child_utime ;
                        system_time += rmp->mp_child_stime ;
                        count_process++ ;
                }
        }

        printf("processes: %d\nuser time: %d\nsystem time: %d\n",
count_process, user_time, system_time);
        return(0);
}``` |

Screenshot Αποτελεσμάτων

```
-bash-4.2# ./test
Process scheduled
processes: 38
user time: 32
system time: 10
-bash-4.2# ./test
Process scheduled
processes: 38
user time: 32
system time: 11
-bash-4.2# _
```

```
Process scheduled
processes: 38
user time: 8
system time: 129
-bash-4.2# sh
Process scheduled
# ./test
Process scheduled
processes: 39
user time: 8
system time: 129
# _
```

Ερώτημα 4

Λίστα τροποποιηθέντων αρχείων

/usr/src/include/minix/callnr.h
/usr/src/servers/pm/table.c
/usr/src/servers/pm/proto.h
/usr/src/servers/pm/utility.c

Τροποποιήσεις

/usr/src/kernel/system/callnr.h

Line	Code
70	#define PROEXIST 70

/usr/src/servers/pm/table.c

Line	Code
84	do_proexist /* 70 = proexist */

/usr/src/servers/pm/proto.h

Line	Code
51	Int do_proexist(void);

/usr/src/servers/pm/utility.c

Line	Code
181	```int do_proexist()
{
        struct mproc *rmp;
        int p_id = m_in.m1_i1;

        for(rmp = &mproc[0] ; rmp < &mproc[NR_PROCS] ; rmp++)
        {
                if((rmp->mp_flags & 1)&&(rmp->mp_pid == p_id))
                {
                        return(1);
                }
        }
        return(0);
}``` |

Screenshot Αποτελεσμάτων

```
   -1 root       0            2613K          0:00    0.26% kernel
    7 root       5     0      1048K          0:00    0.18% vfs
   12 root       2     0      4644K          0:00    0.07% vm
   10 root       1     0       228K          0:00    0.04% tty
   89 service    7     0       120K          0:00    0.04% random
   27 root       7     0       836K    RUN   0:00    0.03% procfs
   62 root       7     0       192K          0:00    0.03% devman
   83 root       7     0       220K          0:00    0.02% devmand
   36 service    5     0      4788K          0:00    0.02% mfs
  156 root       7     0       580K          0:00    0.02% top
    5 root       4     0       248K          0:00    0.02% pm
    4 root       4     0      1364K          0:00    0.00% rs
  104 root       7     0       112K          0:00    0.00% lance
  108 service    7     0      1052K          0:00    0.00% inet
    8 root       3     0       104K          0:00    0.00% memory
    9 root       2     0       144K          0:00    0.00% log

-bash-4.2# ./test2 190
Process scheduled
0
-bash-4.2# ./test2 7
Process scheduled
1
-bash-4.2# _
```

test.c

```
#include <lib.h>
#include <stdio.h>

int main(void)
{
        message m ;
        _syscall(PM_PROC_NR, PROINFO, &m);
}
```

test2.c

```
#include <lib.h>
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char **argv) {
    if (argc < 2) exit(1);

    int i = atoi(argv[1]);
    message m;
    m.m1_i1 = i;

    printf("%d\n",_syscall(PM_PROC_NR, PROEXIST, &m) );
}
```