

# JEGYZŐKÖNYV

Operációs rendszerek BSc

2022. tavasz féléves feladat

Készítette: Stremler László

Neptunkód: AQYO8L

## A feladat leírása:

### IPC mechanizmusok:

Készítsen olyan C programot, amely egy nevesített csővezetéken keresztül fogad szavakat és ezeket egy fileba menti, egy sorszámmal együtt. Illetve egy olyan kliensprogramot, ami ebbe a csőbe adatokat helyez el, amit a felhasználótól kér be. A fő program a SIGTERM szignál hatására szüntesse meg a nevesített csövet és lépjen ki.

### A feladat elkészítésének lépései:

- feladat átgondolása, kitervelése
- kliensprogram megírása
- főprogram elkészítése
- szignálkezelés javítása
- fájlba írás javítása
- tesztelés, hibajavítás

### A futtatás eredménye:

```
laszlo@DESKTOP-GF11QG6:/mnt/c/Users/László/Desktop/OS/AQY08L0sGyak/OSSemTask_AQY08L$ gcc write_to_fifo.c -o kliens
write_to_fifo.c: In function 'main':
write_to_fifo.c:28:13: warning: implicit declaration of function 'exit' [-Wimplicit-function-declaration]
   28 |         exit(0);
      |         ^~~~~
write_to_fifo.c:28:13: warning: incompatible implicit declaration of built-in function 'exit'
write_to_fifo.c:9:1: note: include <stdlib.h> or provide a declaration of 'exit'
    8 | #include <stdbool.h>
  +++ |+#include <stdlib.h>
    9 |
laszlo@DESKTOP-GF11QG6:/mnt/c/Users/László/Desktop/OS/AQY08L0sGyak/OSSemTask_AQY08L$ ./kliens
Ide írhatja azokat a dolgokat, amiket bele szeretne írni a csővezetékbe.
Az exit beírásával lehet kilepni ebből a modból.
teszt
Bevitt szöveg: teszt
teszt123
Bevitt szöveg: teszt123
teszt12345543636
Bevitt szöveg: teszt12345543636
asdasd
Bevitt szöveg: asdasd
negyhatnyolcketto
Bevitt szöveg: negyhatnyolcketto
```

```
laszlo@DESKTOP-GF11QG6:/mnt/c/Users/László/Desktop/OS/AQY08L0sGyak/OSSemTask_AQY08L$ gcc write_to_file.c -o foprog
laszlo@DESKTOP-GF11QG6:/mnt/c/Users/László/Desktop/OS/AQY08L0sGyak/OSSemTask_AQY08L$ ./foprog
Elkezdtem kiolvasni a csővezetékbe írt szöveget!
A csővezetékbe, majd fájlba írt szöveg: teszt
A csővezetékbe, majd fájlba írt szöveg: teszt123
A csővezetékbe, majd fájlba írt szöveg: teszt12345543636
A csővezetékbe, majd fájlba írt szöveg: asdasd
A csővezetékbe, majd fájlba írt szöveg: negyhatnyolcketto
^CLetoroltam a csővezetékbe!
laszlo@DESKTOP-GF11QG6:/mnt/c/Users/László/Desktop/OS/AQY08L0sGyak/OSSemTask_AQY08L$ |
```

```
≡ output.txt U X
≡ output.txt
1 1. teszt
2 2. teszt123
3 3. teszt12345543636
4 4. asdasd
5 5. negyhatnyolcketto
6
```

**A feladat leírása:**

**Ütemezési algoritmusok:**

Adott négy processz (A,B,C,D) a rendszerbe, induláskor a p\_cpu értéke A=0, B=0, C=0, D=0. A rendszerben a P\_USER = 60. Az óraütés 1 indul, a befejezés 301-ig.

Induláskor a p\_usrpri A=60, B=65, C=60, D=60.

Induláskor a p\_nice értéke A=0, B=5, C=0, D=0.

- a.) Határozza meg az ütemezést RR 301 óraütésig – táblázatba!
- b.) Minden óraütem esetén határozza meg a processzek sorrendjét óraütés előtt/után.
- c.) Igazolja a számítással a tanultak alapján.

**A feladat elkészítésének lépései:**

- feladat átgondolása, kitervelése
- táblázat vázának elkészítése
- számítás elkezdése
- hibajavítás

**A futtatás eredménye:**

óraütés	A		B		C		D		Előző proc	Következő proc		
	usr_pri	p_cpu	usr_pri	p_cpu	usr_pri	p_cpu	usr_pri	p_cpu				
0	60	0	65	0	60	0	60	0		A		p_nice: A:0, B:5, C:0, D:0 P_USER = 60
1	60	1	65	0	60	0	60	0	A	A		
2	60	2	65	0	60	0	60	0	A	A		10 órainterruptonként váltja a futó folyamatot
3	60	3	65	0	60	0	60	0	A	A		minden 100.interrupt: p_cpu=p_cpu*KF
4	60	4	65	0	60	0	60	0	A	A		p_usrpri = P_USER+p_cpu/4+2*p_nice
5	60	5	65	0	60	0	60	0	A	A		
6	60	6	65	0	60	0	60	0	A	A		Minden óraütésnél növeli a futó folyamat p_cpu értékét
7	60	7	65	0	60	0	60	0	A	A		Korrektációs faktor értéke: $2^{1/3}/2^{1/3}+1$
8	60	8	65	0	60	0	60	0	A	A		0,8
9	60	9	65	0	60	0	60	0	A	A		
10	60	10	65	0	60	0	60	0	A	C		
20	60	10	65	0	60	10	60	0	C	D		
30	60	10	65	0	60	10	60	10	D	A		100. interrupt: A: p_cpu = 40*0,8 = 32
40	60	20	65	0	60	10	60	10	A	C		p_usrpri = 60 + 31,2/4+2*0 = 68
50	60	20	65	0	60	20	60	10	C	D		B: p_cpu = 0*0,8 = 0
60	60	20	65	0	60	20	60	20	D	A		p_usrpri = 60 + 0/4 + 2*5 = 70
70	60	30	65	0	60	20	60	20	A	C		C: p_cpu = 30*0,8 = 24
80	60	30	65	0	60	30	60	20	C	D		p_usrpri = 60 + 24/4+2*0 = 66
90	60	30	65	0	60	30	60	30	D	A		D: p_cpu = 30*0,8 = 24
99	60	39	65	0	60	30	60	30	A	A		p_usrpri = 60+24/4+2*0 = 66
100	68	32	70	0	66	24	66	24	A	C		
110	68	32	70	0	66	34	66	24	C	D		
120	68	32	70	0	66	34	66	34	D	A		200. interrupt: A: p_cpu = 62*0,8 ~ 50
130	68	42	70	0	66	34	66	34	A	C		p_usrpri = 60 + 50/4+2*0 ~ 73
140	68	42	70	0	66	44	66	34	C	D		B: p_cpu = 0*0,8 = 0
150	68	42	70	0	66	44	66	44	D	A		p_usrpri = 60 + 0/4+2*5 = 70
160	68	52	70	0	66	44	66	44	A	C		C: p_cpu = 63*0,8 ~ 50
170	68	52	70	0	66	54	66	44	C	D		p_usrpri = 60 + 50/4+2*0 ~ 73
180	68	52	70	0	66	54	66	54	D	A		D: p_cpu = 54*0,8 ~ 43
190	68	62	70	0	66	54	66	54	A	C		p_usrpri = 60 + 43/4+2*0 ~ 71
199	68	62	70	0	66	63	66	54	C	C		
200	73	50	70	0	73	50	71	43	C	B		
210	73	50	70	10	73	50	71	43	B	D		300. interrupt: A: p_cpu = 70*0,8 = 56
220	73	50	70	10	73	50	71	53	D	A		p_usrpri = 60+56/4+2*0 = 74
230	73	60	70	10	73	50	71	53	A	C		B: p_cpu = 30*0,8 = 24
240	73	60	70	10	73	60	71	53	C	B		p_usrpri = 60+24/4+2*5 = 76
250	73	60	70	20	73	60	71	53	B	D		C: p_cpu = 70*0,8 = 56
260	73	60	70	20	73	60	71	63	D	A		p_usrpri = 60+56/4+2*0 = 74
270	73	70	70	20	73	60	71	63	A	C		D: p_cpu = 74*0,8 ~ 59
280	73	70	70	20	73	70	71	63	C	B		p_usrpri = 60+59/4+2*0 ~ 75
290	73	70	70	30	73	70	71	63	B	D		
299	73	70	70	30	73	70	71	73	D	D		
300	74	56	76	24	74	56	75	59	D	A		