

# **Operációs rendszerek BSc**

**9. Gyak.**

**2022. 04. 06.**

**Készítette:**

Palencsár Enikő Bsc

Mérnökinformatikus

YD11NL

**Miskolc, 2022**

## 1. feladat

A tanult rendszerhívásokkal (open(), read()/write(), close()) - ők fogják a rendszerhívásokat tovább hívni - írjanak egy neptunkod\_openclose.c programot, amely megnyit egy fájlt – neptunkod.txt, tartalma: hallgató neve, szak , neptunkod.

A program következő műveleteket végezze:

- olvassa be a neptunkod.txt fájlt, melynek attribútuma: O\_RDWR
- hiba ellenőrzést,
- write() - mennyit ír ki a konzolra.
- read() - kiolvassa a neptunkod.txt tartalmát és mennyit olvasott ki (byte), és kiírja konzolra.
- lseek() – pozícionálja a fájl kurzor helyét, ez legyen a fájl eleje: SEEK\_SET, és kiírja a konzolra

```
enikop@enikop-VirtualBox:~/Desktop/OS$ cat YD11NL.txt
Palencsar Eniko, mernokinformatikus, YD11NL
enikop@enikop-VirtualBox:~/Desktop/OS$ gcc YD11NL_openclose.c
enikop@enikop-VirtualBox:~/Desktop/OS$ ./a.out
Called read, 44 bytes were read: Palencsar Eniko, mernokinformatikus, YD11NL

lseek() : 0 is the new file cursor.
Called write, 7 bytes were written.
YD11NL
Called write, 7 bytes were written.
enikop@enikop-VirtualBox:~/Desktop/OS$ cat YD11NL.txt
YD11NL
ar Eniko, mernokinformatikus, YD11NL
enikop@enikop-VirtualBox:~/Desktop/OS$
```

## 2. feladat

Készítse el a következő feladatot, melyben egy szignálkezelő több szignált is tud kezelni:

- Készítsen egy szignál kezelőt (handleSignals), amely a SIGINT (CTRL + C) vagy SIGQUIT (CTRL + \) jelek fogására vagy kezelésére képes.
- Ha a felhasználó SIGQUIT jelet generál (akár kill paranccsal, akár billentyűzetről a CTRL+ \) a kezelő egyszerűen kiírja az üzenetet visszatérési értékét – a konzolra.
- Ha a felhasználó először generálja a SIGINT jelet (akár kill paranccsal, akár billentyűzetről a CTRL + C), akkor a jelet úgy módosítja, hogy a következő alkalommal alapértelmezett műveletet hajtson végre (a SIG\_DFL) – kiírás a konzolra.
- Ha a felhasználó másodszor generálja a SIGINT jelet, akkor végrehajt egy alapértelmezett műveletet, amely a program befejezése - kiírás a konzolra.

Mentés: neptunkod\_tobbsignal.c

```
enikop@enikop-VirtualBox: ~/Desktop/OS
File Edit View Search Terminal Help
enikop@enikop-VirtualBox:~/Desktop/OS$ gcc YD11NL_tobbsignal.c
enikop@enikop-VirtualBox:~/Desktop/OS$ ./a.out
PID: 1864
Waiting...
Waiting...
SIGINT lett megadva - 2
Waiting...
Waiting...
SIGQUIT lett megadva - 3
Waiting...
Waiting...

enikop@enikop-VirtualBox:~/Desktop/OS$ █

enikop@enikop-VirtualBox: ~
File Edit View Search Terminal Help
enikop@enikop-VirtualBox:~$ kill -SIGINT 1864
enikop@enikop-VirtualBox:~$ kill -SIGQUIT 1864
enikop@enikop-VirtualBox:~$ kill -SIGINT 1864
enikop@enikop-VirtualBox:~$ █
```

### 3. feladat

Adott a következő ütemezési feladat, amit a FCFS, SJF és Round Robin (RR:4ms) ütemezési algoritmus alapján határozza meg következő teljesítmény értékeket, metrikákat (külön-külön táblázatba):

#### FCFS:

	P1	P2	P3	P4
Érkezés	0	0	2	5
CPU idő	24	3	6	3
Indulás	0	24	27	33
Befejezés	24	27	33	36
Várakozás	0	24	25	28
Körülfordulási idő	24	27	31	31

#### SJF:

	P1	P2	P3	P4
Érkezés	0	0	2	5
CPU idő	24	3	6	3
Indulás	12	0	3	9
Befejezés	36	3	9	12
Várakozás	12	0	1	4
Várakozók	-	P1, P3	P1, P4	P1
Körülfordulási idő	36	3	7	7

Sorrend: P2, P3, P4, P1

#### RR:

RR: 10 ms	P1			P2	P3			P4
Érkezés	0	4	15	0	2	11		5
CPU idő	24	20	16	3	6	2		3
Indulás	0	11	20	4	7	18		15
Befejezés	4	15	36	7	11	20		18
Várakozás	0	7	5	4	5	7		10
Körülfordulási idő	36			7	18			13

Sorrend: P1, P2, P3, P1, P4, P3, P1

Külön táblázatba számolja a teljesítmény értékeket!

CPU kihasználtság: számolni kell a cs: 0,1(ms) és sch: 0,1 (ms)

Algoritmus neve	FCFS
CPU kihasználtság	$36/36.4 = 98.9\%$ (4 db cs)
Körülfordulási idők átlaga	$(24+27+31+31)/4 = 28.25$ ms
Várakozási idők átlaga	$(0+24+25+28)/4 = 19.25$ ms
Válaszidők átlaga	$(0+24+25+28)/4 = 19.25$ ms

Algoritmus neve	SJF
CPU kihasználtság	$36/36.4 = 98.9\%$ (4 db cs)
Körülfordulási idők átlaga	$(36+3+7+7)/4 = 13.25$ ms
Várakozási idők átlaga	$(12+0+1+4)/4 = 4.25$ ms
Válaszidők átlaga	$(12+0+1+4)/4 = 4.25$ ms

Algoritmus neve	RR
CPU kihasználtság	$36/37 = 97.3\%$ (7 db cs, 3 db sch)
Körülfordulási idők átlaga	$(36+7+18+13)/4 = 18.5$ ms
Várakozási idők átlaga	$(0+7+5+4+5+7+10)/4 = 9.5$ ms
Válaszidők átlaga	$(0+4+5+10)/4 = 4.75$ ms

#### 4. feladat.

Mentés. neptunkod\_gyak9\_1.c

```
enikop@enikop-VirtualBox:~/Desktop/05$ gcc YD11NL_gyak9_1blocker.c
enikop@enikop-VirtualBox:~/Desktop/05$ gcc YD11NL_gyak9_1blocker.c
enikop@enikop-VirtualBox:~/Desktop/05$ ./a.out 2348
enikop@enikop-VirtualBox:~/Desktop/05$
```

```
enikop@enikop-VirtualBox:~/Desktop/05$ ./a.out
PID: 2348
Signal kezelo atvetele: 0
Waiting...
YD11NL
Not blocked anymore. Terminating...
enikop@enikop-VirtualBox:~/Desktop/05$
```

## Mentés. neptunkod\_gyak9\_2.c

```
enikop@enikop-VirtualBox: ~  
File Edit View Search Terminal Help  
enikop@enikop-VirtualBox:~$ kill 1891  
enikop@enikop-VirtualBox:~$ kill 1891  
enikop@enikop-VirtualBox:~$ kill -SIGKILL 1891  
enikop@enikop-VirtualBox:~$  
enikop@enikop-VirtualBox:~/Desktop/OS$ gcc YD11NL_gyak9_2.c  
YD11NL_gyak9_2.c: In function 'main':  
YD11NL_gyak9_2.c:12:35: warning: format '%d' expects argument of type 'int', but  
argument 2 has type 'sig_handler_t' {aka 'void (*)(int)'} [-Wformat=]  
12 | printf("Signal kezelo atvetele: %d\n", signal(SIGTERM, signalHandler));  
   |                                     ^~  
   |                                     |  
   |                                     int      sig_handler_t {aka void (*)(int  
   |                                     )}  
enikop@enikop-VirtualBox:~/Desktop/OS$ ./a.out  
PID: 1891  
Signal kezelo atvetele: 0  
Waiting...  
Waiting...  
Waiting...  
The value of int argument is 15  
Waiting...  
Waiting...  
The value of int argument is 15  
Waiting...  
Waiting...  
Waiting...  
Killed  
enikop@enikop-VirtualBox:~/Desktop/OS$
```