

# **Operációs rendszerek BSc**

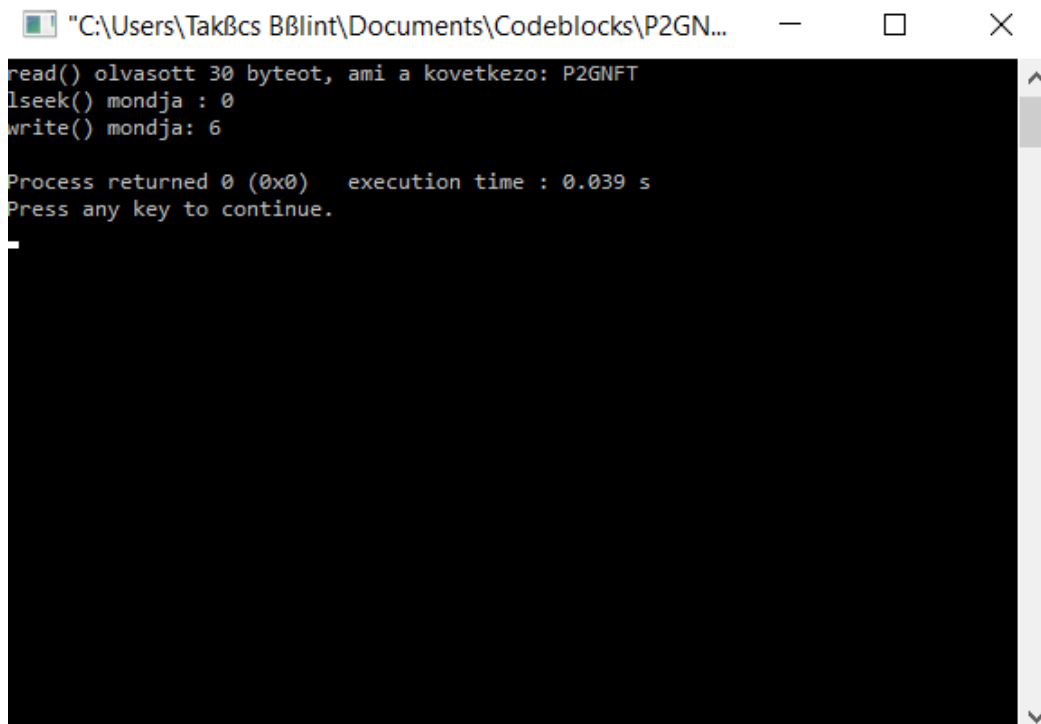
1. gyak.  
2022. 02. 09.

**Készítette:**

Takács Bálint Bsc  
Programtervező informatikus  
P2GNFT

**Miskolc, 2022**

1. . 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.



```
"C:\Users\TakBcs BBlint\Documents\Codeblocks\P2GN..."  
read() olvasott 30 byteot, ami a kovetkezo: P2GNFT  
lseek() mondja : 0  
write() mondja: 6  
  
Process returned 0 (0x0) execution time : 0.039 s  
Press any key to continue.
```

2. Készítse el a következő feladatot, melyben egy szignálkezelő  
több szignált is tud kezelni:
  - a.) Készítsen egy szignál kezelőt (handleSignals), amely a  
SIGINT (CTRL + C) vagy SIGQUIT (CTRL + \) jelek  
fogására vagy kezelésére képes.

- b.) Ha a felhasználó SIGQUIT jelet generál (akár kill paranccsal, akár billentyűzetről a CTRL + \) a kezelő egyszerűen kiírja az üzenetet visszatérési értékét – a konzolra.
- c.) 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.
- d.) Ha a felhasználó másodszor generálja a SIGINT jelet, akkor végrehajt egy alapértelmezett műveletet, amely a program befejezése - kiírás a konzolra. Mentés: neptunkod\_tobbszig.c

```
bash 02feladat$ ./tobbszig
PID: 649
waiting for signal...
received SIGQUIT
waiting for signal...
received SIGINT
waiting for signal...
received SIGQUIT
waiting for signal...
```

```
bash 02feladat$ kill -s QUIT 649
bash 02feladat$ kill -s INT 649
bash 02feladat$ kill -s QUIT 649
bash 02feladat$ kill -s INT 649
bash 02feladat$ |
```

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

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

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

	Érkezés	CPU idő	Indulás	Befejezés	Várakozás
P1	0	24	0	4	0
P2	0	3	4	7	4
P1*	4	20	7	11	3
P3	2	6	11	15	9
P1*	11	16	15	19	4
P3*	15	2	19	21	4
P1*	19	12	21	25	2
P4	5	3	25	28	20
P1*	25	8	28	32	3
P1*	32	4	32	36	0