**Operációs rendszerek BSc**

9. Gyak.

2022. 04. 06.

**Készítette:**Sikora Dávid Ádám Bsc  
Mérnökinformatika  
IRE699

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.

**2.Feladat** 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\_tobbszignal.c

**3.Feladat** Adott a következő ütemezési feladat, amit a FCFS, SJF és Round Robin (RR: 4 ms) ütemezési algoritmus alapján határozza meg következő teljesítmény értékeket, metrikákat (külön-külön táblázatba):  
Külön táblázatba számolja a teljesítmény értékeket!   
CPU kihasználtság: számolni kell a cs: 0,1(ms) és sch: 0,1 (ms) értékkel is

**4.Feladat** Írjon C nyelvű programot, amelyik kill() seg.-vel SIGALRM-et küld egy argumentumként megadott PID-u processznek, egy másik futó program a SIGALRM-hez rendeljen egy fv.-t amely kiírja pl. neptunkodot, továbbá pause() fv.-el blokkolódjon, majd kibillenés után jelezze, hogy kibillent és terminálódjon.  
Mentés. neptunkod\_gyak9\_1.c

**5.Feladat** Írjon C nyelvű programot, amelyik a SIGTERM-hez hozzárendel egy fv-t., amelyik kiírja az int paraméter értéket, majd végtelen ciklusban fusson, 3 sec-ig állandóan blokkolódva elindítás után egy másik shell-ben kill paranccsal (SIGTERM) próbálja terminálni, majd SIGKILL-el.”  
Mentés. neptunkod\_gyak9\_2.