

Operációs rendszerek BSc

7.gyak.

2021. 03. 27.

Készítette:

Juhász Marcell Tibor
Bsc

Programtervező
informatikus

O9V4M0

Miskolc, 2021

1. Feladat:

RR nélkül

- A,B,C,D Processz ebben a sorrendben

- $p_uspri = 60$

- A,B,C $p_nice = 0$, D $p_nice = 5$

- $p_cpu = 0$

- 1-től 201-ig

$p_cpu = p_cpu * KF$, ahol KF értéke $\frac{1}{2}$

$p_pri = P_USER + p_cpu / 4 + 2 * p_nice$

P_USER konstans, értéke 50

RR Nélkül: RR_nelkul.pdf

RR algoritmussal: RR.pdf

2. Feladat:

A tanult rendszer hívásokkal (open(), read()/write(), close()) írjanak egy neptunkod_openclose.c programot, amely megnyit egy fájlt – neptunkod.txt, tartalma: hallgató neve, szak, neptun kód. A program következő műveleteket végezze el:

- olvassa be a neptunkod.txt-t, attribútuma: O_RDWR

- hiba ellenőrzés

- write()

- read()

- lseek() - SEEK_SET-et a fájl elejére állítva

Az open() és a fileDescriptor változó használatához az fcntl.h headert kell importálni. Az open() parancs 2 argumentumot fogad: A file elérési útvonalát és a file kezelési módszerét. Az open() parancs visszatérési értéke (helyes működés esetén) egy egész, amely az azonosítót adja vissza az argumentumban megadott file-hoz, amit a processz elér. Hiba esetén a visszatérési érték -1. A feladatban a visszatérési értéket a fileDescriptor változóban kell tárolni.

Az lseek() az unist.h header könyvtárban található, ezért azt is importálni kell. Az lseek metódus 3 argumentumot fogad és visszatérési értéke egész, -1 ha nem sikerült a pozícionálás.