

Operációs Rendszerek BSc

7.gyak.

2021.03.24

Készítette:

Orosz Dániel Bsc

Üzemmérnök-informatikus

C5S7FM

Miskolc, 2021

1. Adott négy processz a rendszerbe, melynek beérkezési sorrendje: A, B, C és D. minden processz USER módban fut és minden processz futásra kész. Kezdetben minden processz $p_uspri = 60$. Az A, B, C processz $p_nice = 0$, a D processz $p_nice = 5$. minden processz $p_cpu = 0$, az óraütés 1 indul, a befejezés legyen 201. óraütés-ig.

a.) Határozza meg az ütemezést RR nélkül és az ütemezést RR-nal - külön-külön táblázatba.

b.) minden óraütem esetén határozza meg a processzek sorrendjét óraütés előtt/után.

c.) Igazolja a számítással a tanultak alapján.

$KF = 2 \cdot FK / 2 \cdot FK + 1$ - korrekciós faktor;

$p_cpu = p_cpu * KF$, ahol KF értéke 1/2;

Minden processznál öregíti a p_cpu értékét az

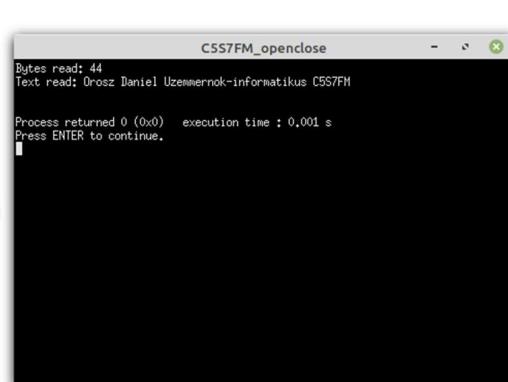
$p_pri = P_USER + p_cpu / 4 + 2 * p_nice$;

A		B		C		D			
p_pri	p_cpu	p_pri	p_cpu	p_pri	p_cpu	p_pri	p_cpu	Lépés	Futó foly.
60	0	60	0	60	0	60	0	1	A
60	1	60	0	60	0	60	0	2	A
...	A
60	99	60	0	60	0	60	0	99	A
72	50	60	0	60	0	60	0	100	A
72	50	60	1	60	0	60	0	101	B
...	B
72	50	60	99	60	0	60	0	199	B
66	25	72	50	60	0	60	0	200	B
66	25	72	50	60	1	60	0	201	C
...	C
66	25	72	50	60	99	60	0	299	C
63	13	66	25	72	50	60	0	300	C
63	13	66	25	72	50	60	1	301	D
...	D
63	13	66	25	72	50	60	99	399	D
62	7	63	13	66	25	72	50	400	D

2. A tanult rendszerhívásokkal (open(), read()/write(), close() - ók 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.



```
main.c
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <unistd.h>
4 #include <fcntl.h>
5
6 int main()
7 {
8     int file, sz;
9     char *data = (char *) calloc(100, sizeof(char));
10
11    file = open("C5S7FM.txt", O_RDWR);
12    if (file < 0){
13        perror("r1");
14        exit(1);
15    }
16
17    sz = write(file, "Orosz Daniel Uzemernok-informatikus C5S7FM", 42);
18    close(file);
19
20    file = open("C5S7FM.txt", O_RDONLY);
21
22    sz = read(file, data, 100);
23    data[sz] = '\0';
24    printf("Bytes read: %d\nText read: %s\n", sz, data);
25
26    lseek(file, 0, SEEK_SET);
27
28    return 0;
29
30 }
```

write(): A megadott nevű fájlba beleírja a megadott szöveget, 3.paraméterként a byteot kell megadni.

read(): A megadott nevű fájl tartalmát kiolvassa és azt eltárolja a 2.paraméterként megadott helyen.