

Operációs rendszerek BSc

6.gyak.

2021. 03. 24.

Készítette:

Takács Bálint Bsc

Programtervező informatikus

P2GNFT

Miskolc, 2021

1. feladat - Adott négy processz a rendszerbe, melynek beérkezési sorrendje: A, B, C és D. Minden processz USER módban fut és mindegyik processz futásra kész. Kezdetben mindegyik processz $p_{uspri} = 60$. Az A, B, C processz $p_{nice} = 0$, a D processz $p_{nice} = 5$. Mindegyik processz $p_{cpu} = 0$, az óráütés 1 indul, a befejezés legyen 201. óráütés-ig.

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

b.) Minden óráütem esetén határozza meg a processzek sorrendjét óráütés előtt/után.

c.) Igazolja a számítással a tanultak alapján.

RR nélkül:

Clock tick	A process		B process		C process		D process		Reschedule	
	p_{uspri}	p_{cpu}	p_{uspri}	p_{cpu}	p_{uspri}	p_{cpu}	p_{uspri}	p_{cpu}	running before	running after
Starting point	60	0	60	0	60	0	60	0	-	A
1	60	1	60	0	60	0	60	0	A	A
...
9	60	9	60	0	60	0	60	0	A	A
10	60	10	60	0	60	0	60	0	A	A
11	60	11	60	0	60	0	60	0	A	A
...
98	60	98	60	0	60	0	60	0	A	A
99	60	99	60	0	60	0	60	0	A	A
100	72	50	60	0	60	0	70	0	A	B
101	72	50	60	1	60	0	70	0	B	B
102	72	50	60	2	60	0	70	0	B	B
...
198	72	50	60	98	60	0	70	0	B	B
199	72	50	60	99	60	0	70	0	B	B
200	66	25	72	50	60	0	70	0	B	C
201	66	25	72	50	60	1	70	0	C	C

$$KF = \frac{1}{2}$$

$$p_{user} = 60$$

	A processz	B processz	C processz	D processz
Clock tick = 100	$p_cpu = 100 * \frac{1}{2} = 50$ $p_uspri = 60 + 50/4 + 0 = 72$	$p_cpu = 0$ $p_uspri = 60 + 0 + 0 = 60$	$p_cpu = 0$ $p_uspri = 60 + 0 + 0 = 60$	$p_cpu = 0$ $p_nice = 5$ $p_uspri = 60 + 0 + 10 = 70$
Clock tick = 200	$p_cpu = 50 * \frac{1}{2} = 25$ $p_uspri = 60 + 25/4 + 0 = 66$	$p_cpu = 100 * \frac{1}{2} = 50$ $p_uspri = 60 + 50/4 + 0 = 72$	$p_cpu = 0$ $p_uspri = 60 + 0 + 0 = 60$	$p_cpu = 0$ $p_nice = 5$ $p_uspri = 60 + 0 + 10 = 70$

RR-nal:

	A process		B process		C process		D process		Reschedule	
Clock tick	p_uspri	p_cpu	p_uspri	p_cpu	p_uspri	p_cpu	p_uspri	p_cpu	running before	running after
Starting point	60	0	60	0	60	0	60	0	-	A
1	60	1	60	0	60	0	60	0	A	A
...
9	60	9	60	0	60	0	60	0	A	A
10	60	10	60	0	60	0	60	0	A	B
11	60	10	60	1	60	0	60	0	B	B
...
19	60	10	60	9	60	0	60	0	B	B
20	60	10	60	10	60	0	60	0	B	C
21	60	10	60	10	60	1	60	0	C	C
...
29	60	10	60	10	60	9	60	0	C	C
30	60	10	60	10	60	10	60	0	C	D
31	60	10	60	10	60	10	60	1	D	D
...
39	60	10	60	10	60	10	60	9	D	D
40	60	10	60	10	60	10	60	10	D	A
41	60	11	60	10	60	10	60	10	A	A
...
49	60	19	60	10	60	10	60	10	A	A
50	60	20	60	10	60	10	60	10	A	B
51	60	20	60	11	60	10	60	10	B	B
...
59	60	20	60	19	60	10	60	10	B	B
60	60	20	60	20	60	10	60	10	B	C
61	60	20	60	20	60	10	60	10	C	C
...
69	60	20	60	20	60	10	60	10	C	C
70	60	20	60	20	60	10	60	10	C	D
71	60	20	60	20	60	11	60	11	D	D
...
79	60	20	60	20	60	19	60	19	D	D
80	60	20	60	20	60	20	60	20	D	A
81	60	21	60	20	60	20	60	20	A	A
...
89	60	29	60	20	60	20	60	20	A	A
90	60	30	60	20	60	20	60	20	A	B
91	60	30	60	21	60	20	60	20	B	B
...
99	60	30	60	29	60	20	60	20	B	B
100	63	15	63	15	62	10	72	10	B	C
101	63	15	63	15	60	11	72	10	C	C
...
199	63	15	63	15	60	109	72	10	C	C
200	61	7	61	7	73	55	71	5	C	A
201	61	8	61	7	73	55	71	5	A	A

$$KF = \frac{1}{2}$$

$$p_{\text{user}} = 60$$

	A processz	B processz	C processz	D processz
Clock tick =100	$p_{cpu} = 30 * \frac{1}{2} = 15$ $p_{uspri} = 60 + 3 = 63$	$p_{cpu} = 30 * \frac{1}{2} = 15$ $p_{uspri} = 60 + 3 = 63$	$p_{cpu} = 20 * \frac{1}{2} = 10$ $p_{uspri} = 60 + 2 = 62$	$p_{cpu} = 20 * \frac{1}{2} = 10$ $p_{nice} = 5$ $p_{uspri} = 60 + 2 + 10 = 72$
Clock tick =200	$p_{cpu} = 15 * \frac{1}{2} = 7$ $p_{uspri} = 60 + 1 = 61$	$p_{cpu} = 15 * \frac{1}{2} = 7$ $p_{uspri} = 60 + 1 = 61$	$p_{cpu} = 110 * \frac{1}{2} = 55$ $p_{uspri} = 60 + 13 = 73$	$p_{cpu} = 10 * \frac{1}{2} = 5$ $p_{nice} = 5$ $p_{uspri} = 60 + 1 + 10 = 71$

2. 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. 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() – pozicionálja 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 : 0.048 s  
Press any key to continue.  
_
```