

Operációs rendszerek BSc

8. Gyak.

2022. 03. 28.

Készítette:

Martinák Mátyás Bsc

Programtervező informatikus

KLNSPG

Miskolc, 2022

1. feladat - Adott a következő ütemezési feladat, amit a FCFS, SJF és Round Robin (RR:10ms) ütemezési algoritmus alapján határozza meg következő teljesítmény értékeket, metrikákat

FCFS	P1	P2	P3	P4
Érkezés	0	7	11	20
CPU idő	14	8	36	10
Indulás	0	14	22	58
Befejezés	14	22	58	68
Várakozás	0	7	11	38
Körülfordulási idő	14	15	47	48

Algoritmus neve	FCFS	SJF	RR: 10 ms
CPU kihasználtság			
Körülfordulási idők átlaga	31	24,5	41
Várakozási idők átlaga	14	7,5	11
Válaszidők átlaga			

SJF	P1	P2	P3	P4
Érkezés	0	7	11	20
CPU idő	14	8	36	10
Indulás	0	14	32	22
Befejezés	14	22	68	32
Várakozás	0	7	21	2
Körülfordulási idő	14	15	57	12

RR: 10ms	P1	P1 (2)	P2	P3	P3 (2)	P3 (3)	P3 (4)	P4
Érkezés	0	10	7	11	32	52	62	20
CPU idő	14	4	8	36	26	16	6	10
Indulás	0	18	10	22	42	52	62	32
Befejezés	10	22	18	32	52	62	68	42
Várakozás	0	8	3	11	10	0	0	12
Körülfordulási idő	14	12	11	47	36	16	6	22
Várakozási idők szummázva	44							
Körülfordulási idő szummázva	164							

2. feladat - Adott négy processz a rendszerbe, melynek a ready sorban a 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 301. óráütés-ig.
- a.) Határozza meg az ütemezést RR nélkül 301 óráütésig és RR-nal 201 óráütésig – külön külön táblázatba!
- b.) Minden óráütem esetén határozza meg a processzek sorrendjét óráütés előtt/után.
- c.) Igazolja a számítással a tanultak alapján.

RR nélküli:

	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	A
1	60	1	60	0	60	0	60	0	A	A
...	A	A
9	60	9	60	0	60	0	60	0	A	A
10	60	10	60	0	60	0	60	0	A	B
...
19	60	10	60	9	60	0	60	0	B	B
20	60	10	60	10	60	0	60	0	B	C
...
29	60	10	60	10	60	9	60	0	C	C
30	60	10	60	10	60	10	60	0	C	D
...
39	60	10	60	10	60	10	60	9	D	D
40	60	10	60	10	60	10	60	10	D	A
50	60	20	60	10	60	10	60	10	A	B
60	60	20	60	20	60	10	60	10	B	C
70	60	20	60	20	60	20	60	10	C	D
80	60	20	60	20	60	20	60	20	D	A
90	60	30	60	20	60	20	60	20	A	B
100	67	26	67	26	64	17	64	27	B	C
...
199	67	46	67	46	64	37	64	46	D	D
200	70	39	70	39	68	31	70	40	D	A
201	70	40	70	39	68	31	70	40	A	A

$$p_cpu = 100 / 0,5 = 50$$

$$p_uspri(1) = P_USER + 50 / 4 + 2 * p_nice = 73$$

$$p_uspri(2) = P_USER + 25 / 4 + 2 * p_nice = 66$$

RR-rel:

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	A
1	60	1	60	0	60	0	60	0	A	A
...	A	A
99	60	99	60	0	60	0	60	0	A	A
100	73	50	60	0	60	0	60	0	A	B
101	73	50	60	1	60	0	60	0	B	B
...		
199	73	50	60	99	60	0	60	0	B	B
200	66	25	73	50	60	0	60	0	B	C
201	66	25	73	50	60	1	60	0	C	C

100. óraütesnél:

- $KF = 2 * FK / 2 * FK + 1 = (2 * 3) / (2 * 3 + 1) = 0,85$
- A p_cpu = $30 * 0,85 = 26$ A p_uspri = $60 + (26 / 4) = 67$
- B p_cpu = $30 * 0,85 = 26$ B p_uspri = $60 + (26 / 4) = 67$
- C p_cpu = $20 * 0,85 = 17$ C p_uspri = $60 + (17 / 4) = 64$
- D p_cpu = $20 * 0,85 = 17$ D p_uspri = $60 + (17 / 4) + 10 = 74$

200. óraütesnél:

- $KF = 2 * FK / 2 * FK + 1 = (2 * 3) / (2 * 3 + 1) = 0,85$
- A p_cpu = $30 * 0,85 = 39$ A p_uspri = $60 + (26 / 4) = 70$
- B p_cpu = $30 * 0,85 = 39$ B p_uspri = $60 + (26 / 4) = 70$
- C p_cpu = $20 * 0,85 = 31$ C p_uspri = $60 + (17 / 4) = 68$
- D p_cpu = $20 * 0,85 = 40$ D p_uspri = $60 + (17 / 4) + 10 = 70$