Operációs rendszerek – 8. Gyakorlat

Ütemezési algoritmusok, teljesítmény értékek meghatározása

Az elkészült feladatokat töltse fel az aktuális mappába: Neptunkod_....

Jegyzőkönyv neve: neptunkod_Gyak8.pdf

Forrás fájlok

Határidő: aktuális gyakorlat időpontja, ill. módosítás esetén 2022.04.03.

Irodalom

Tanulmányozzák Vadász Dénes: Operációs rendszerek, 2006. ME, jegyzet/diasor, ill.

Vincze Dávid: Operációs rendszerek - diasort.

Szintén tanulmányozzák az előadáson bemutatott/kivetített mintapéldát és az URL linkhez tartozó irodalmat, majd oldják meg a feladatot.

Feladatok

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

	P1	P2	Р3	P4
Érkezés	0	7	11	20
CPU idő	14	8	36	10
Indulás				
Befejezés				
Várakozás				

Külön táblázatba számolja a teljesítmény értékeket!

Algoritmus neve					
CPU kihasználtság					
Körülfordulási idők átlaga					
Várakozási idők átlaga					
Válaszidők átlaga					

CPU kihasználtság: számolni kell a cs: 0,1(ms) és sch: 0,1 (ms)

2. 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 óraütés 1 indul, a befejezés legyen 301. óraütés-ig.

- **a.**) Határozza meg az ütemezést *RR nélkül 301 óraütésig* és *RR*-nal *201 óraütésig* különkülön táblázatba!
- ${f b.}$) Minden óraütem esetén határozza meg a processzek sorrendjét óraütés előtt/után.
- c.) Igazolja a számítással a tanultak alapján.

A táblázat javasolt formája RR/RR nélkül a következő:

A process		B process		C process		D process		Reschedule	
p_uspri	p_cpu	p_uspri	p_cpu	p_uspri	p_uspri	p_uspri	p_cpu	running before	running after
60	0	60	0	60	0	60	0		
	:	:	:	:	:	:	:	:	:
:	:	:	:	:	:	:	:	:	:
	p_uspri	p_uspri p_cpu	p_uspri p_cpu p_uspri	p_uspri p_cpu p_uspri p_cpu	p_uspri p_cpu p_uspri p_cpu p_uspri	p_uspri p_cpu p_uspri p_cpu p_uspri p_uspri	p_uspri p_cpu p_uspri p_cpu p_uspri p_uspri p_uspri	p_uspri p_cpu p_uspri p_cpu p_uspri p_uspri p_uspri p_cpu	p_uspri p_cpu p_uspri p_cpu p_uspri p_uspri p_uspri p_cpu running before