

Operációs rendszerek BSc

7. Gyak.

2022. 03. 23.

Készítette:

Pázmán András Bsc

Szak Mérnökinformatikus

Neptunkód H2Z4X3

Miskolc, 2022

Operációs rendszerek – 7. Gyakorlat

Ütemezési algoritmusok (FCFS, SJF, RR)

Töltse fel az aktuális mappába: **Neptunkod_....**

Jegyzőkönyv neve: *neptunkod_Gyak7.pdf*

Forrás fájlok

Határidő: aktuális gyakorlat időpontja, ill. módosítás esetén 2022.03.27.

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 használatával készítsen el (külön-külön táblázatba):

	P1	P2	P3	P4
Érkezés	0	8	12	20
CPU idő	15	7	26	10
Indulás	0	15	22	48
Befejezés				
Várakozás				

Határozza meg:

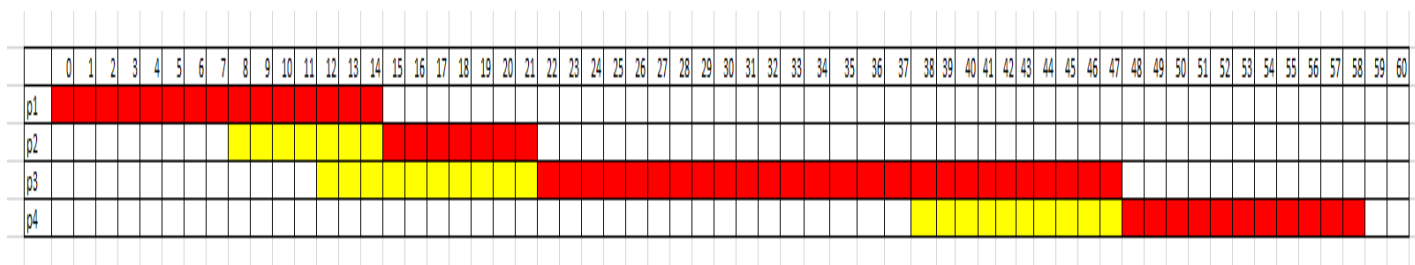
a.) A befejezési idő?

b.) A várakozási/átlagos várakozási idő, ill. a processzek végrehajtási sorrendjét?

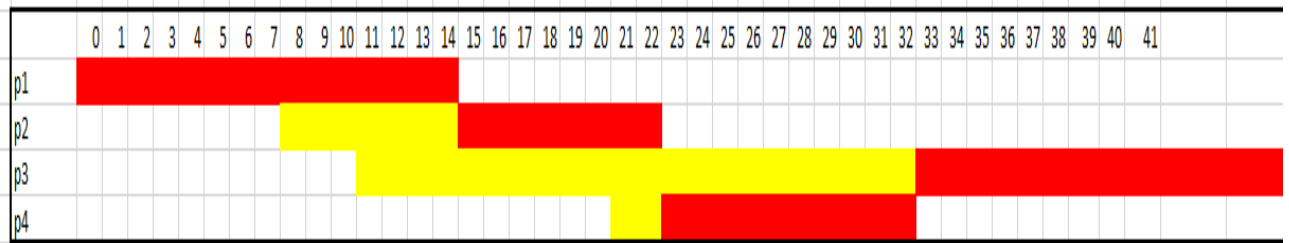
c.) Ábrázolja Gantt diagram segítségével az *aktív/várakozó processzek* futásának menetét.

Megj.: a Gantt diagram ábrázolása szerkesztő program segítségével vagy Excel programmal segítségével.

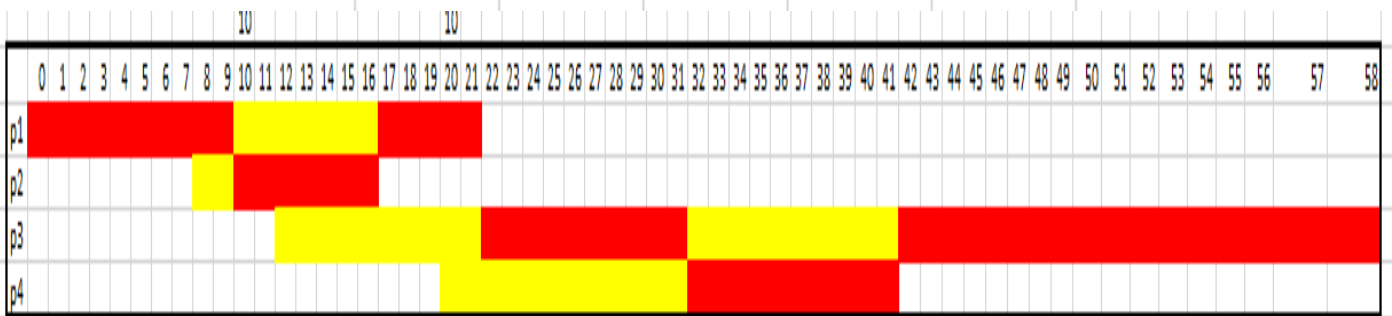
FCFS				
	P1	P2	P3	P4
Érkezés	0	8	12	20
CPU idő	15	7	26	10
Indulás	0	15	22	48
Befejezés	15	22	48	58
Várakozás	0	7	10	28



SJF				
	P1	P2	P3	P4
Érkezés	0	8	12	20
CPU idő	15	7	26	10
Indulás	0	15	22	48
Befejezés	15	22	48	58
Várakozás	0	7	10	28



RR				
	P1	P2	P3	P4
Érkezés	0	8	12	20
CPU idő	15	7	26	10
Indulás	0,17	10	22,42	32
Befejezés	9,21	16	31,56	41
Várakozás	7	2	10,1	12



Az FCFS alacsonyabb eszköz- és CPU-kihasználáshoz vezet, ezáltal csökkenti a rendszer hatékonyságát. Az SJF az alacsonyabb átlagos várakozási idő miatt a rendszer nagyobb hatékonyságához vezet. Az RR jobb átlagos válaszidőt biztosít. Ennek azonban általában magasabb az átlagos átfutási ideje.

2. Adott a következő ütemezési feladat, amit Round Robin (RR) ütemezési algoritmus használatával készítsen el 10 ms és 4 ms időszelét esetén. (külön-külön táblázatba):

RR	P1	P2	P3	P4	P5
Érkezés	0	3	3	6	8
CPU idő	3	10	3	6	3
Indulás					
Befejezés					
Várakozás					
Körülfordulási idő:					

Átlagos várakozási idő

Átlagos körülfordulási idő

Határozza meg:

a.) A befejezési időt, várakozási/átlagos várakozási időt, ill. a processzek végrehajtási sorrendjét?

b.) Határozza meg az *átlagos körülfordulási időt*, magyarázza melyik időszelettel jobb az átlagos körülfordulási idő és melyiknél rosszabb a CPU kihasználtság!

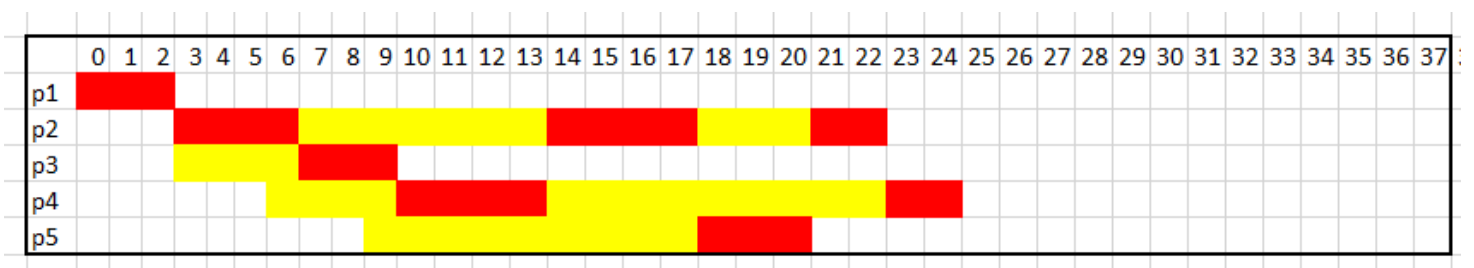
Megj.: *Átlagos körülfordulási idő*: $\Sigma \text{CPU idő} + \Sigma \text{várakozás} / n$

Egy processz a rendszerbe *helyezéstől a befejezésig eltelt idő*.

c.) Ábrázolja Gantt diagram segítségével az *aktív/várakozó processzek* futásának menetét!

Megj.: a Gantt diagram ábrázolása szerkesztő program segítségével vagy Excel programmal.

RR					
4ms	P1	P2	P3	P4	P5
Érkezés	0	3	3	6	8
CPU idő	3	10	3	6	3
Indulás	0	3,14	7	10	10,23
Befejezés	2	6,17	9	13	13,24
Várakozás	0	7,3	0	4,9	9
Kör.idő	2	3,17	6	7	5,24

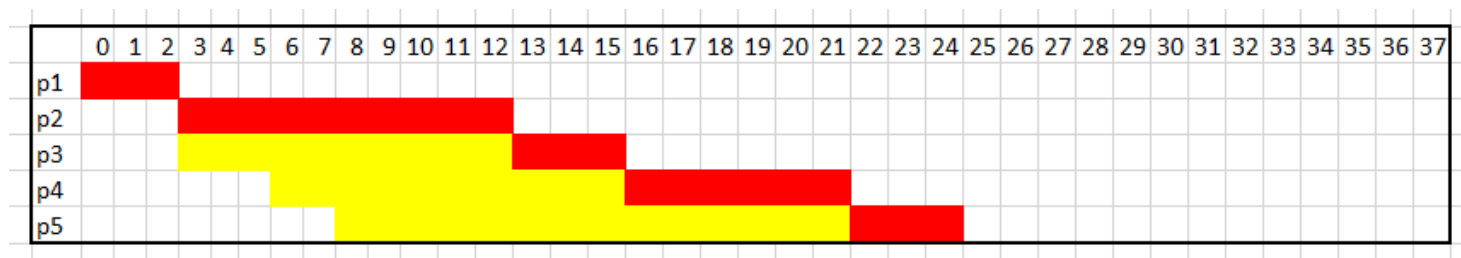


Átlagos várakozás: 4,68

Átlagos körülfordulási idő: 4,24

Ahol kevesebb a körülfordulási idő, jobb a cpu kihasznalts

RR					
10ms	P1	P2	P3	P4	P5
Érkezés	0	3	3	6	8
CPU idő	3	10	3	6	3
Indulás	0	3	13	16	22
Befejezés	2	12	10	11	14
Várakozás	0	0	10	10	14
Kör.idő	2	9	7	5	6



Átlagos várakozás: 6,8

Átlagos körülfordulási idő: 5,8