

Scheduling

CPU scheduling decisions:

1. running \rightarrow waiting state
2. running \rightarrow ready state
3. waiting \rightarrow ready state
4. terminates

Dacă sunt doar 1. și 4. \Rightarrow nonpreemptive

În rest \Rightarrow preemptive

Preemptive scheduling \Rightarrow condiție de cursă

CPU utilization : time CPU ocupat

Throughput : # de procese completate per time unit

Turnaround time : timpul în care se execută un anumit proces

Waiting time : timpul pe care l-a așteptat un proces în ready queue

Response time : timpul dintre momentul de request și primul răspuns produs

FCFS (Primul venit primul servit)

Process	Burst Time
P_1	24
P_2	3
P_3	3

- Presupunând că procesele vin într-o anumită ordine: P_1, P_2, P_3



Waiting time:

$$P_1 = 0$$

$$P_2 = 24$$

$$P_3 = 27$$

$$\text{Average waiting time: } (0 + 24 + 27) / 3 = 17$$

- Presupunând că procesele vin în ordinea: P_2, P_3, P_1



Waiting time:

$$P_1 = 6$$

$$P_2 = 0$$

$$P_3 = 3$$

$$\text{Average waiting time: } (6 + 0 + 3) / 3 = 3 \Rightarrow \text{un rezultat mult mai bun decât cel anterior}$$

Procesele vin în funcție de Arrival Time.

Dacă Arrival Time nu e în tabel \Rightarrow vin toate în același timp

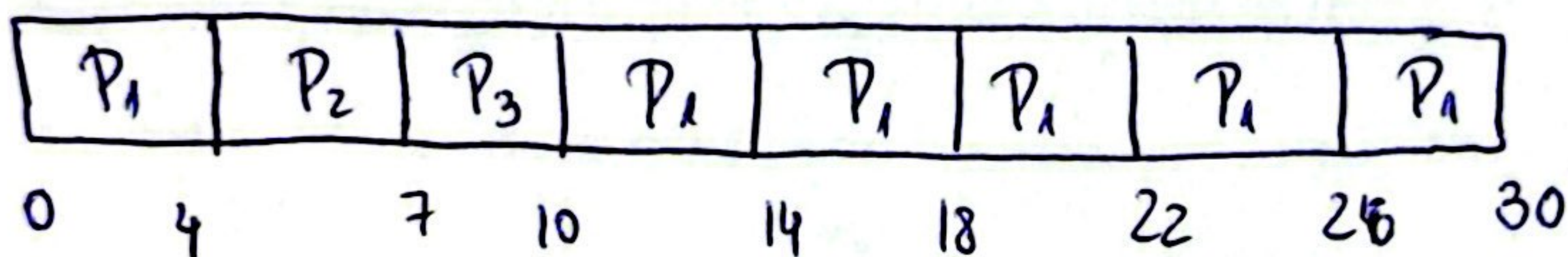
Round Robin

Dacă q este : $\left\{ \begin{array}{l} \text{mare} \Rightarrow \text{FIFO (adică FCFS)} \\ \text{mic} \Rightarrow \text{Round Robin} \end{array} \right.$

Niciun proces nu așteaptă mai mult de $(n-1)q$ (unde n este numărul total de procese). Deci dacă $W/T > (n-1)q \Rightarrow$ Algoritm greșit

ex:

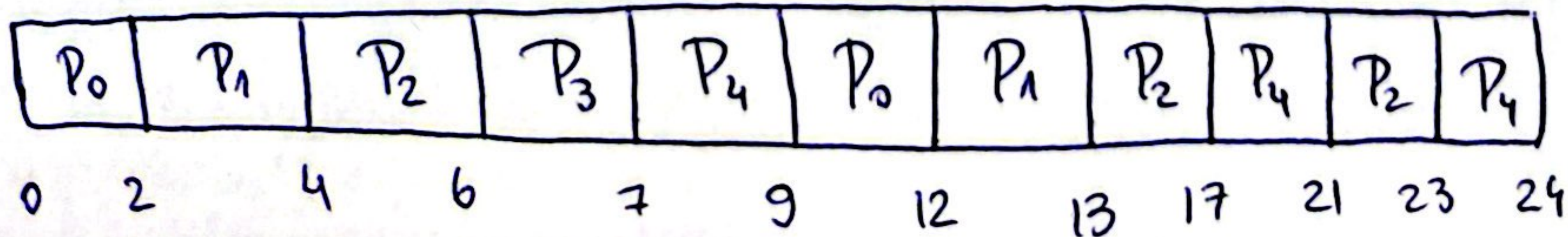
Process	Burst Time
P_1	24
P_2	3
P_3	3



Dacă în tabel gs fi avut și Arrival Time atunci el dicta în ce ordine intră procesele

ex:

Process	Arrival time	Burst Time
P_0	0	5
P_1	0	3
P_2	2	8
P_3	4	1
P_4	6	7



SJF (Shortest Job First)

Process	Burst Time
P ₁	6
P ₂	8
P ₃	7
P ₄	3

P_4	P_1	P_3	P_2	
0	3	9	16	24

Average waiting time:
 $(3+16+9+0)/4 = 7$

SJF preemptive (numit zi Shortest remaining time first)

Process	Arrival Time	Burst Time
P ₁	0	8
P ₂	1	4
P ₃	2	9
P ₄	3	5

P_1	P_2	P_4	P_1	P_3	
0	1	5	10	17	26

Average waiting time : $[(10-1) + (1-1) + (17-2) + (5-3)]/4 = 26/4 = 6.5$

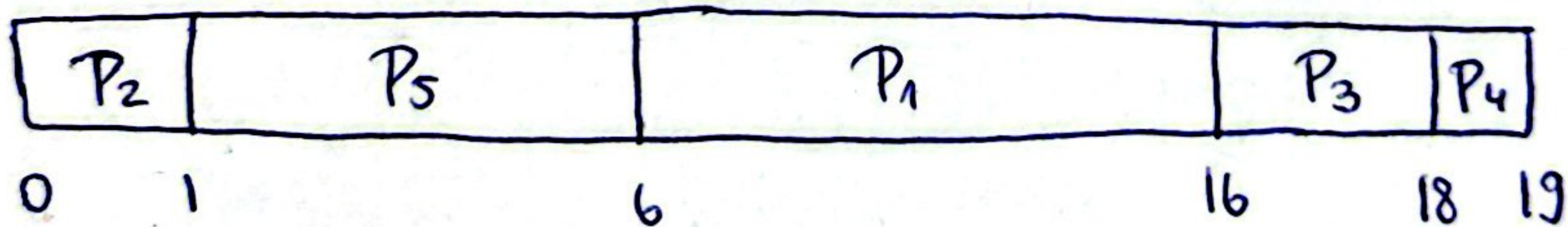
Priority scheduling

Dacă CPU $\left\{ \begin{array}{l} \text{este mare (prioritate mică)} \\ \text{este mic (prioritate mare)} \end{array} \right.$

Deci se începe de la cel mai mic CPU

Problema este starvation (procesele cu low priority pot să nu fie niciodată executate), iar soluția este aging (cu timpul care trece se increas-mieste prioritatea procesului)

Process	Burst Time	Priority
P_1	10	3
P_2	1	1
P_3	2	4
P_4	1	5
P_5	5	2



Average waiting time: $(0 + 1 + 6 + 16 + 18) / 5 = 8,2$

Priority scheduling w/ Round Robin

Run la procesul cu prioritatea cea mai mare

La procesele cu aceeasi prioritate se face RR (daca ele)

Process	Burst time	Priority
P_1	4	3
P_2	5	2
P_3	8	2
P_4	7	1
P_5	3	3

$Q=2$:

