

Il primo algoritmo di scheduler FCFS First-come First-served è adatto soprattutto a sistemi batch con un'interazione minima dell'utente, quindi in questo caso è da scartare.

Il secondo tipo di algoritmo SJF shortest Job First in questo caso garantisce tempi medi di attesa molto bassi, però bisogna partire dal presupposto di conoscere il CPU burst a priori quando il processo entra nella coda dei processi pronti.

Il terzo algoritmo di scheduler è il Round Robin, molto adatto ai sistemi interattivi in quanto assegna dei quanti di tempo nei quali il processo ha dei tempi da rispettare e se non termina nel quanto di tempo assegnato verrà data priorità ad un altro processo.

Il quarto algoritmo è a base prioritaria, vengono assegnate ai processi delle etichette di priorità e quando un processo entra nella coda dei pronti viene confrontato col processo in esecuzione, se ha priorità maggiore verrà eseguito, altrimenti resta in attesa.

Esistono anche gli algoritmi a schedulazione "multi-livello" e real-time che sono le più utilizzate oggi.

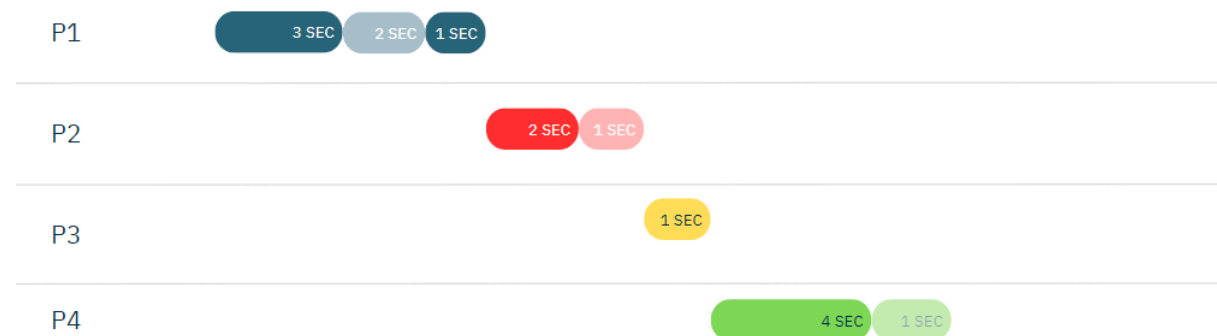
Il primo crea più code dove quelle più in cima utilizzano algoritmo RR e se il processo continua a non finire nel quanto di tempo andrà in fondo alla coda dove si utilizza algoritmo FCFS.

Hard Real-time è utilizzato soprattutto in quei sistemi che devono assolutamente finire il task nel tempo prestabilito o ci sarebbero ripercussioni (elettronica dell'auto, aerei ecc)

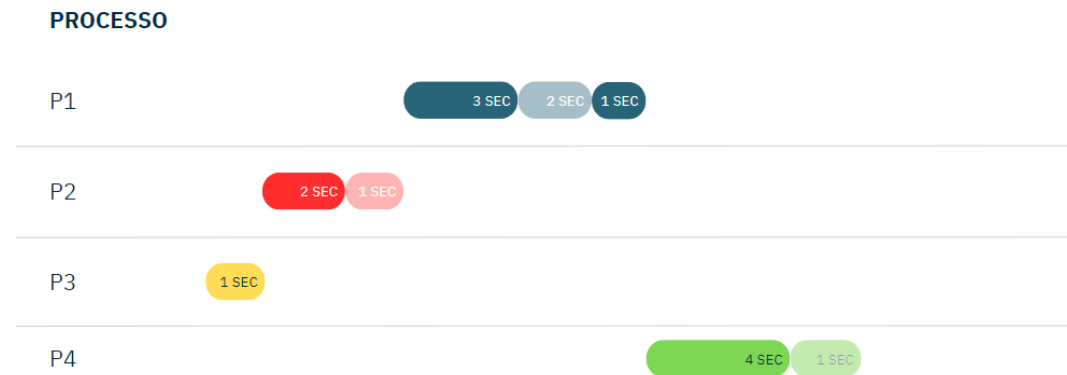
Soft real-time se non finisce nei tempi prestabiliti si ha un'esperienza utente poco piacevole (sistemi multimediali).

FCFS: Molto lento in questo caso, T di attesa medio = $25/4 = 6.25$ secondi

PROCESSO



SJF: Veloce, T di attesa medio = $15/4 = 3.75$ secondi



RR: Penalizza i processi che hanno tempi di esecuzione molto brevi come P3 e P2 e il T di attesa medio = $26/4 = 6.5$ secondi



Multi-level feedback piu veloce a terminare

