

Report S3/L1

Matteo Congiu

L'esercizio di oggi verte sui meccanismi di pianificazione dell'utilizzo della CPU (o processore). In ottica di ottimizzazione della gestione dei processi, abbiamo visto come lo scheduler si sia evoluto nel tempo per passare da approccio mono-tasking ad approcci multi-tasking. Traccia: Si considerino 4 processi, che chiameremo P1,P2,P3,P4, con i tempi di esecuzione e di attesa input/output dati in tabella. I processi arrivano alle CPU in ordine P1,P2,P3,P4. Individuare il modo più efficace per la gestione e l'esecuzione dei processi, tra i metodi visti nella lezione teorica. Abbozzare un diagramma che abbia sulle ascisse il tempo passato da un istante «0» e sulle ordinate il nome del Processo.

Processo	Tempo di esecuzione	Tempo di attesa	Tempo di esecuzione dopo attesa
P1	3 secondi	2 secondi	1 secondo
P2	2 secondi	1 secondo	-
P3	1 secondi	-	-
P4	4 secondi	1 secondo	-

Svolgimento

Mono-Tasking

Nel primo grafico ho raffigurato il sistema Mono-Tasking.

Con il colore giallo ho raffigurato il tempo di utilizzo della CPU.

Con il colore verde ho raffigurato il tempo di attesa eventi esterni.

P1: 3s di esecuzione + 2s attesa + 1s esecuzione

T. esecuzione P1: 4s + 2s attesa, **tot 6s**

P2: 2s di esecuzione + 1s attesa

T. esecuzione P2: (6s di P1)+ 2s esecuzione + 1s attesa, **tot 9s**

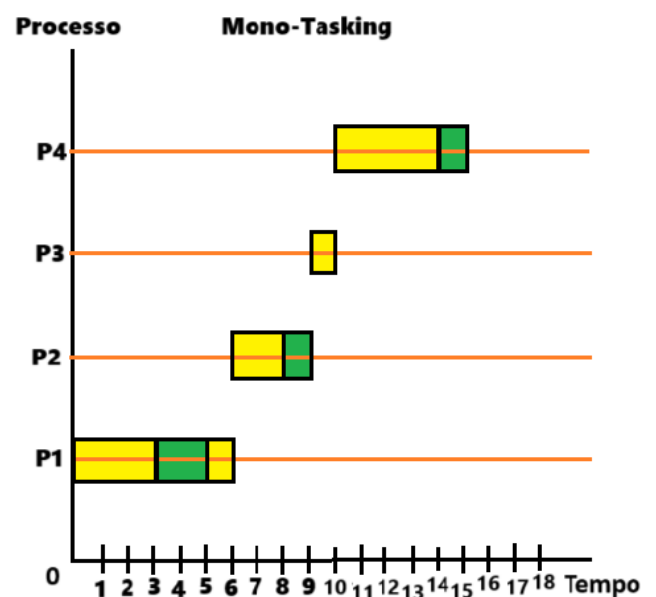
P3: 1s esecuzione

T. esecuzione P3: (6s P1)+(3s P2)+ (1s P3), **tot 10s**

P4: 4s esecuzione + 1s attesa

T. esecuzione P4: (6s P1)+(3s P2)+(1s P3)+(5s P4), **tot 15s**

Tempo Totale di esecuzione dei 4 Processi: 15s



Multi-Tasking

Nel secondo grafico ho raffigurato il sistema Multi-Tasking.

P1: 3s di esecuzione + 2s attesa + 1s esecuzione

T. esecuzione P1: 4s + 2s attesa, **tot 6s**

P2: 2s di esecuzione + 1s attesa

T. esecuzione P2: (3s di P1)+ 2s esecuzione + 1s attesa, **tot 6s**

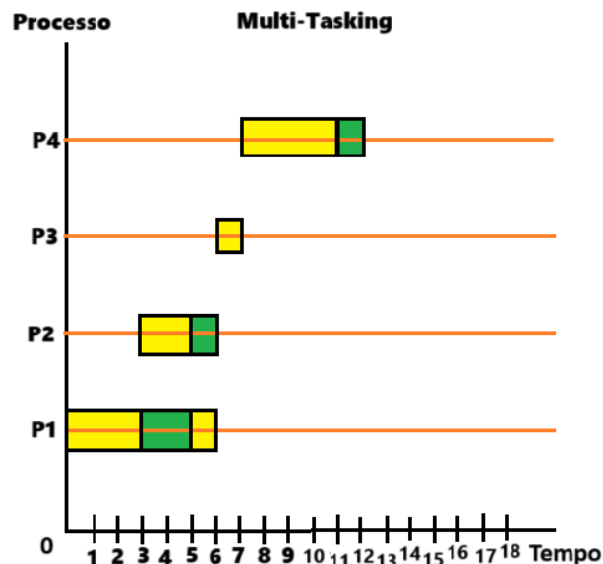
P3: 1s esecuzione

T. esecuzione P3: (6s P1+P2)+ (1s P3), **tot 7s**

P4: 4s esecuzione + 1s attesa

T. esecuzione P4: (6s P1+P2)+(1s P3)+(5s P4), **tot 12s**

Tempo Totale di esecuzione dei 4 Processi: 12s



Time-Sharing

Nel terzo grafico ho raffigurato il sistema Time-Sharing.

Nel time-sharing ho considerato 1 secondo il tempo di **quanto**.

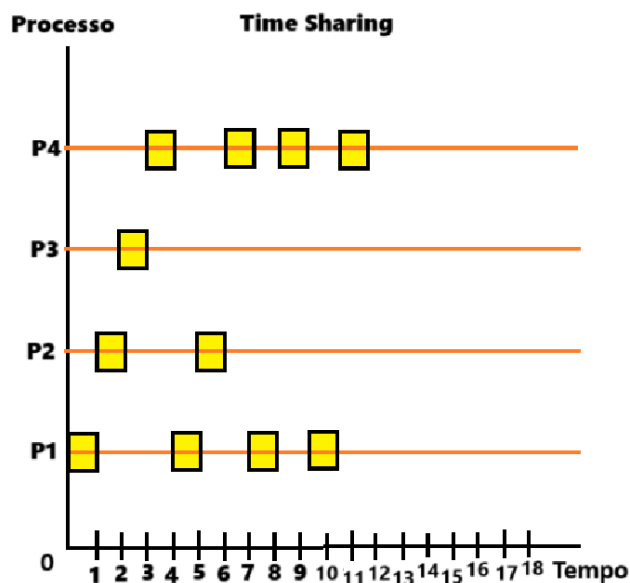
Tempo esecuzione P1 10s

Tempo esecuzione P2 6s

Tempo esecuzione P3 3s

Tempo esecuzione P4 11s

Tempo Totale di esecuzione dei 4 Processi: 11s



Conclusione

Il **mono-tasking** si è rivelato il metodo più lento, a causa del fatto che esegue un solo processo per volta, il processo seguente non inizia fino a quando non termina il processo precedente. Tempo di esecuzione dei 4 processi **15s**.

Il **multi-tasking** grazie alla sovrapposizione delle esecuzioni sulle pause ha impiegato 3s in meno rispetto al mono-tasking. Tempo di esecuzione **12s**.

Il metodo più veloce di esecuzione si è rivelato il **time-sharing** che con la sua esecuzione ciclica suddivide equamente i 4 processi. Tempo di esecuzione **11s**.