

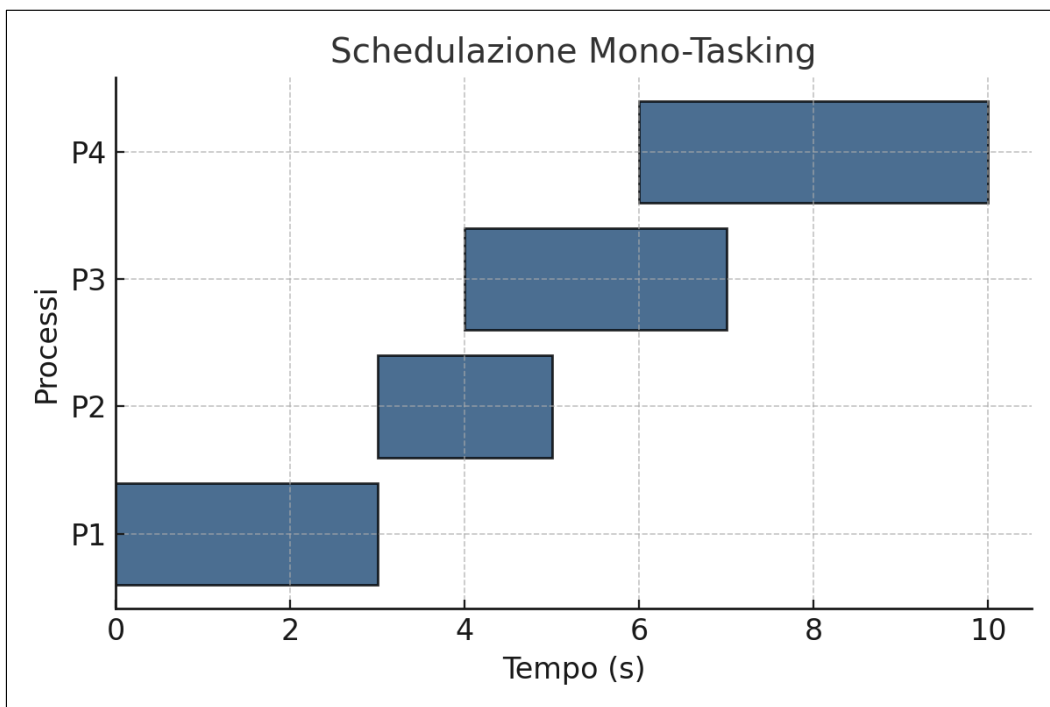
W4D1 - Scheduling della CPU

Lo scheduling dei processi è un aspetto fondamentale nei sistemi operativi, poiché determina l'ordine e l'efficienza con cui la CPU esegue i task in esecuzione. In questo report, analizzeremo tre diverse strategie di scheduling: Mono-Tasking, Time-Sharing, Multi-Tasking e Round Robin applicate a due scenari:

1. Quattro processi (P1-P4) con esecuzione sequenziale;
2. Cinque processi (P1-P5) con tempi di arrivo e durata variabili (esercizio facoltativo).

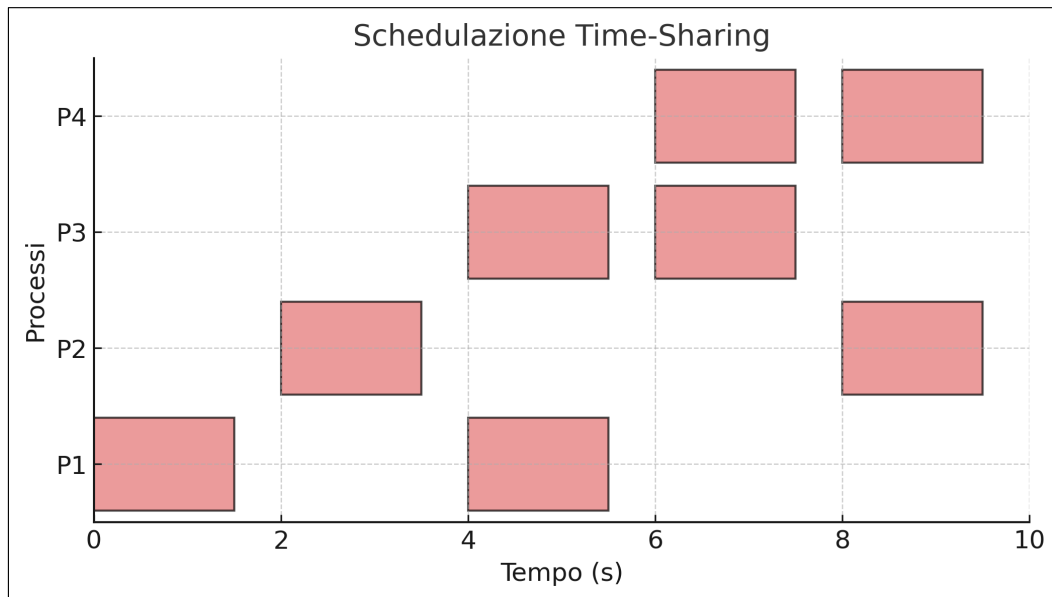
Sulla base dei diagrammi di Gantt, valuteremo quale strategia risulta più efficiente per la gestione dei processi, considerando sia l'ottimizzazione delle risorse che i tempi di completamento.

Schedulazione Mono-Tasking



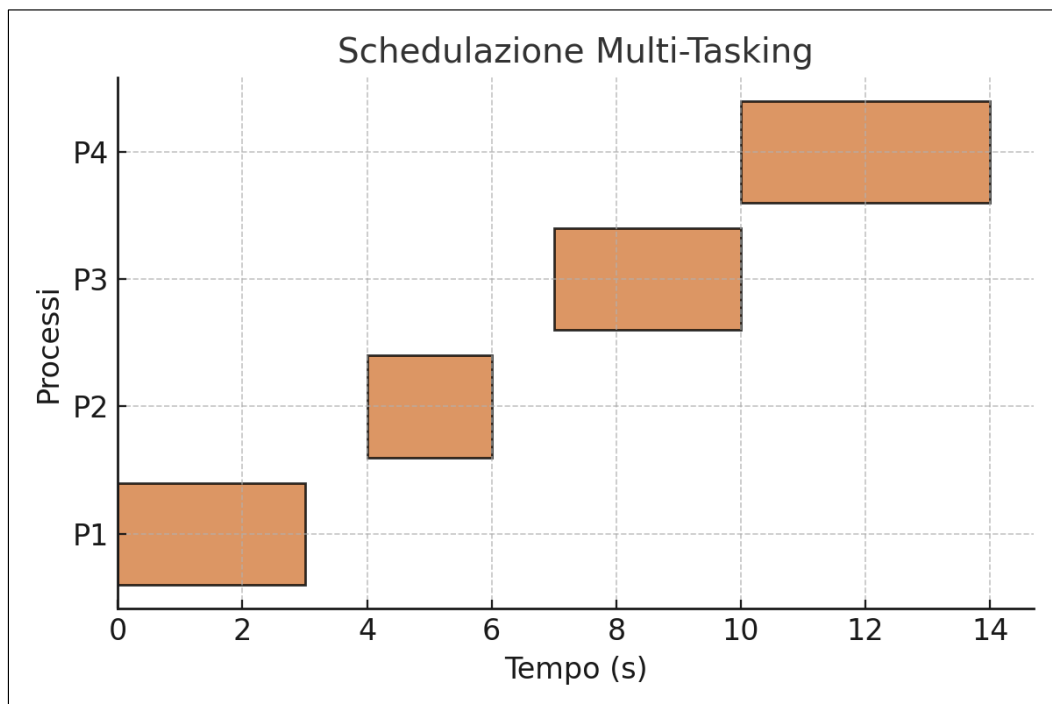
1. Caratteristiche: esecuzione sequenziale dei processi, uno alla volta;
2. Tempo totale: 10 secondi;
3. Ordine esecuzione: P1->P2->P3->P4;
4. Svantaggio: tempi morti durante gli I/O non vengono sfruttati.

Schedulazione Time-Sharing



1. Caratteristiche: assegnazione a turni della CPU ai processi;
2. Tempo totale: 10 secondi;
3. Esecuzione: alternanza rapida tra processi (quantum breve);
4. Vantaggio: migliore responsività, tutti i processi avanzano.

Schedulazione Multi-Tasking



1. Caratteristiche: ottimizzazione con sovrapposizione CPU-I/O;
2. Tempo totale: 14 secondi;
3. Esecuzione: quando un processo è in I/O, la CPU passa ad altro;
4. Vantaggio: massimo sfruttamento delle risorse.

Confronto tra le strategie

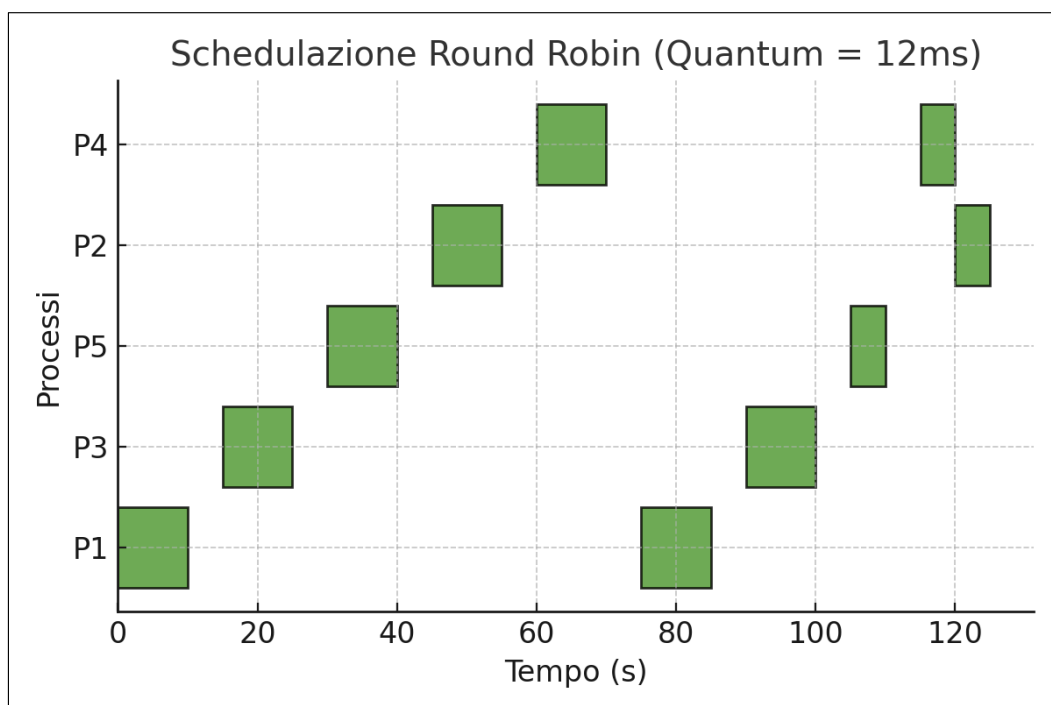
Metodo	Tempo totale	Responsività	Utilizzo CPU	Complessità
Mono-Tasking	10s	Bassa	Basso	Semplice
Time-Sharing	10s	Alta	Medio	Medio
Multi-Tasking	14s	Alta	Alto	Complessa

Il metodo **Time-Sharing** sembra essere il più efficace per questo scenario perché:

1. Mantiene lo stesso tempo totale del Mono-Tasking (10s vs 14s del Multi-Tasking);
2. Offre buona responsività (tutti i processi avanzano);
3. Ha una complessità implementativa ragionevole;

Il Multi-Tasking, pur ottimizzando l'utilizzo della CPU, risulta in un tempo totale maggiore in questo caso specifico. Il Mono-Tasking è troppo inefficiente nell'utilizzo delle risorse.

Esercizio Facoltativo



1. Ordine di esecuzione: P1->P3->P5->P2->P4->(cicli successivi);
2. Tempo totale: 120 ms;
3. Preemption: ogni processo cede la CPU dopo 12 ms, anche se non completato.

Calcolo dei tempi

1. Turnaround:
 - P1: 58 ms (completo al secondo ciclo);
 - P2: 88 ms (completo al quarto ciclo);
 - P3: 112 ms (completo al quinto ciclo);
 - P4: 120 ms (completo al quinto ciclo);
 - P5: 106 ms (completo al quarto ciclo);
 - Media turnaround: 96.8 ms.
2. Tempo di attesa:
 - P1: $58 - 14 = 44$ ms;
 - P2: $88 - 16 = 72$ ms;
 - P3: $112 - 40 = 72$ ms;
 - P4: $120 - 26 = 94$ ms;
 - P5: $106 - 28 = 78$ ms
 - Media attesa: 72 ms.

Conclusioni

1. Per i quattro processi sequenziali, il Time-Sharing è risultato ottimale;
2. Per i cinque processi con arrivi scalati, il Round Robin garantisce equità, ma con tempi di attesa medi più alti (72 ms).