

## PRATICA S3/L1

**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.

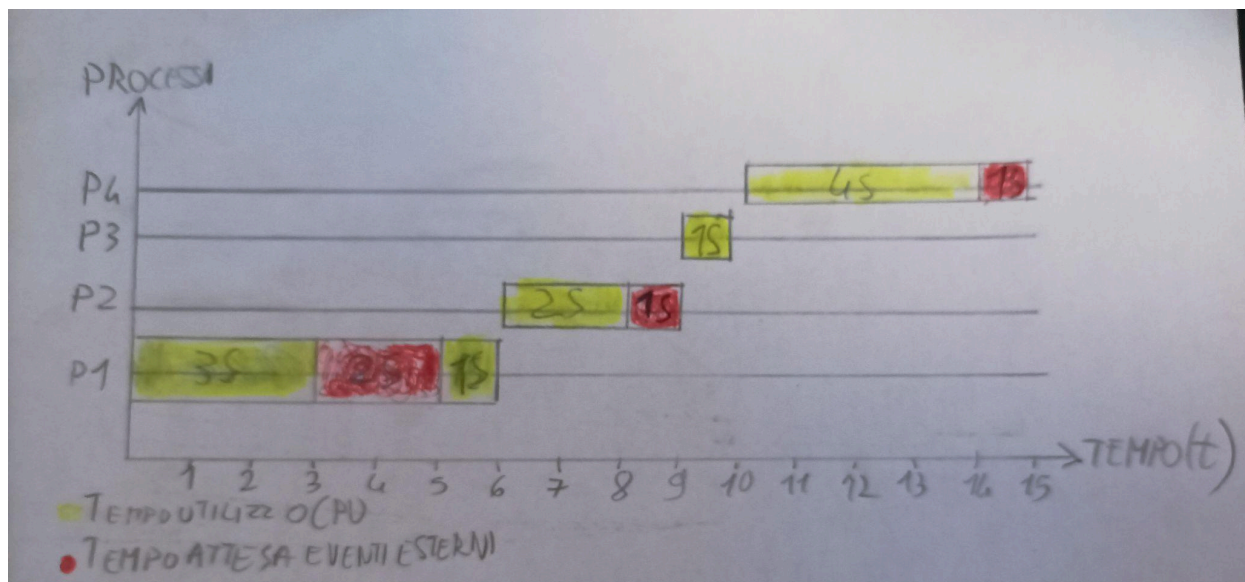
Processo	Tempo di esecuzione	Tempo di attesa	Tempo di esecuzione dopo attesa
P1	3 secondi	2 secondi	1 secondo
P2	2 secondi	1 secondo	
P3	1 secondo		
P4	4 secondi	1 secondo	

Per una valutazione del sistema di pianificazione più efficiente, consideriamo 3 scenari:

1. Sistemi mono-tasking
2. Sistemi multi-tasking
3. Sistemi time-sharing

### Metodo #1: Sistemi mono-tasking

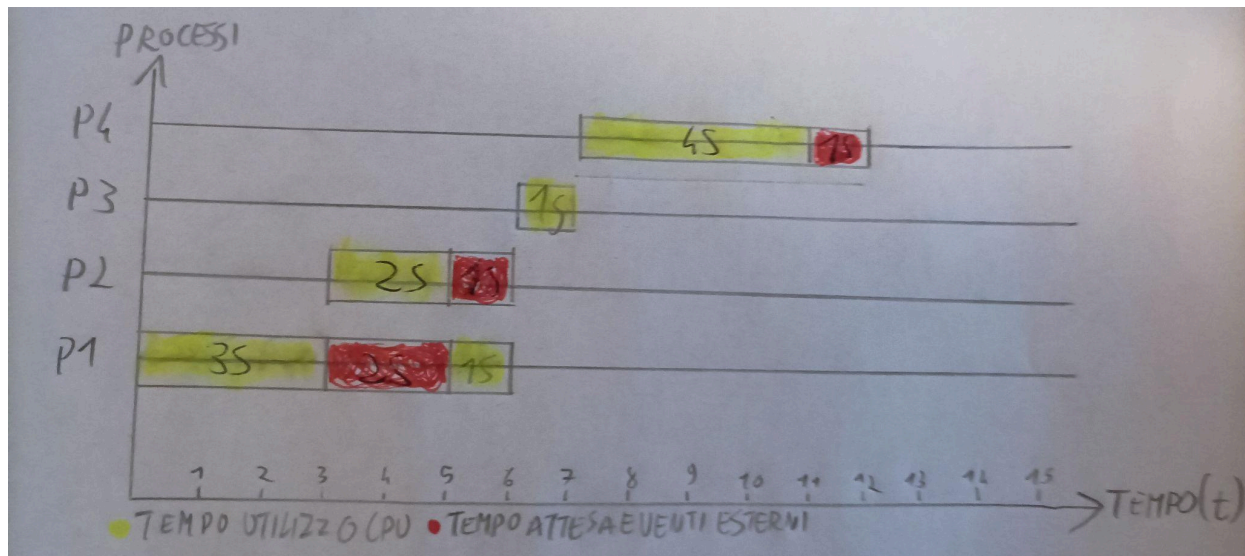
In un sistema mono-tasking, la CPU può eseguire solo un'operazione alla volta. Ogni processo deve aspettare che i tempi d'attesa di quelli che sono già in esecuzione siano già finiti prima di essere eseguito, quindi non ci possono essere sovrapposizioni tra tempi di esecuzione e di attesa.



I processi si susseguono linearmente, partendo dal primo P1, proseguendo per P2, P3, e infine P4. Questo modello ormai antiquato veniva usato per CPU a basso numero di clock cycles che non permettevano operazioni in multitasking.

## Metodo #2: Sistemi multi-tasking

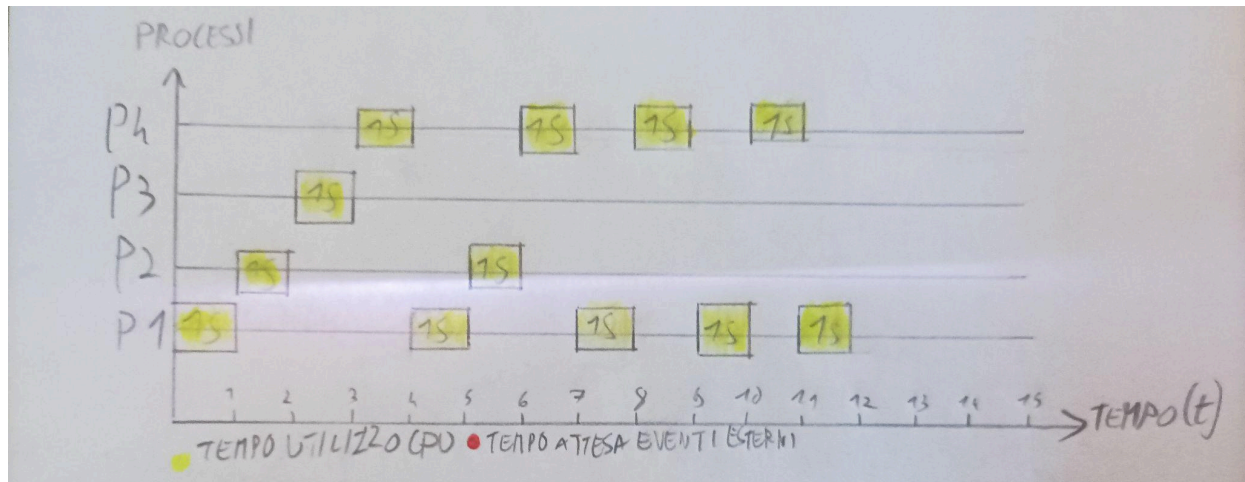
In un sistema multitasking la CPU può eseguire più processi in maniera simultanea. Durante i tempi di attesa di un processo la CPU passa all'esecuzione di un altro, ottimizzando i tempi.



Mentre il processo P1 è in pausa la CPU esegue il processo P2, poi quando P2 è in pausa finisce con P1. Questo è un metodo efficiente per gestire una CPU, dato che riduce i tempi di esecuzione dei processi utilizzando al massimo tutto il potenziale dell'hardware.

### Metodo #3: Sistemi time-sharing

In un sistema time-sharing ogni processo viene eseguito in maniera ciclica, dopo un secondo di tempo eseguito per un'operazione la CPU passa al prossimo processo, quando un altro secondo è passato la CPU attiva l'esecuzione del terzo processo, e così via.



Questo metodo è un'evoluzione del multitasking, e in quanto tale si ottimizza l'uso della CPU. Tuttavia potrebbe risultare inefficiente in caso che si debbano eseguire molti processi con un solo secondo di durata.

### Conclusioni

I metodi più efficienti sono il multitasking e il time-sharing, anche se quest'ultimo potrebbe risultare inefficiente in casi specifici in quanto il valore dei "quanti" è definito a monte. In un sistema time-sharing non importa quanti "quanti" abbia un processo, dopo un solo secondo di esecuzione questo verrà interrotto per poi passare al prossimo. Mentre in un sistema multitasking un processo non verrà mai interrotto a meno che non passi in stato di "awaiting input" (per esempio se ci sia urgenza di eseguire un altro processo con priorità più alta).