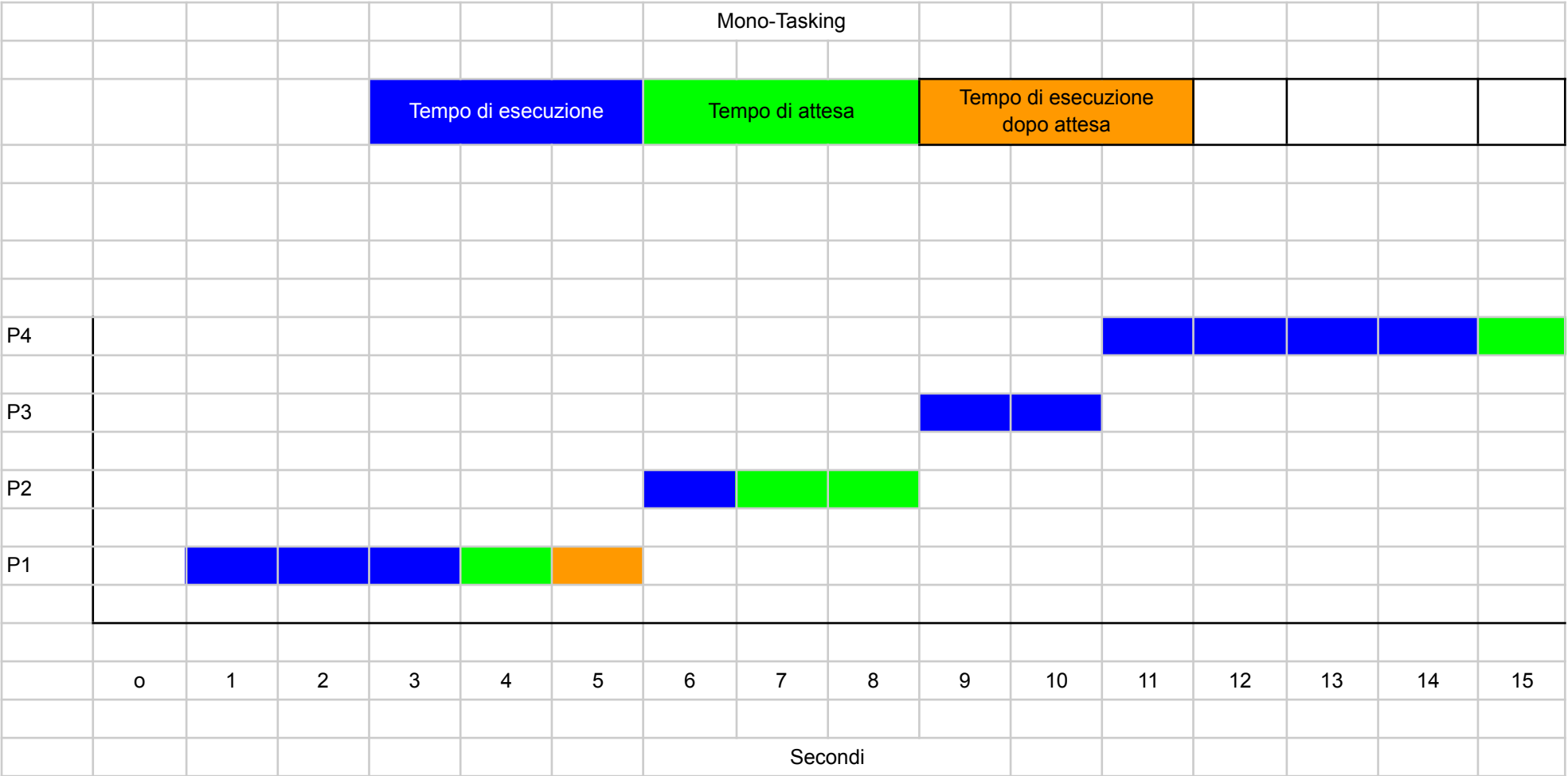


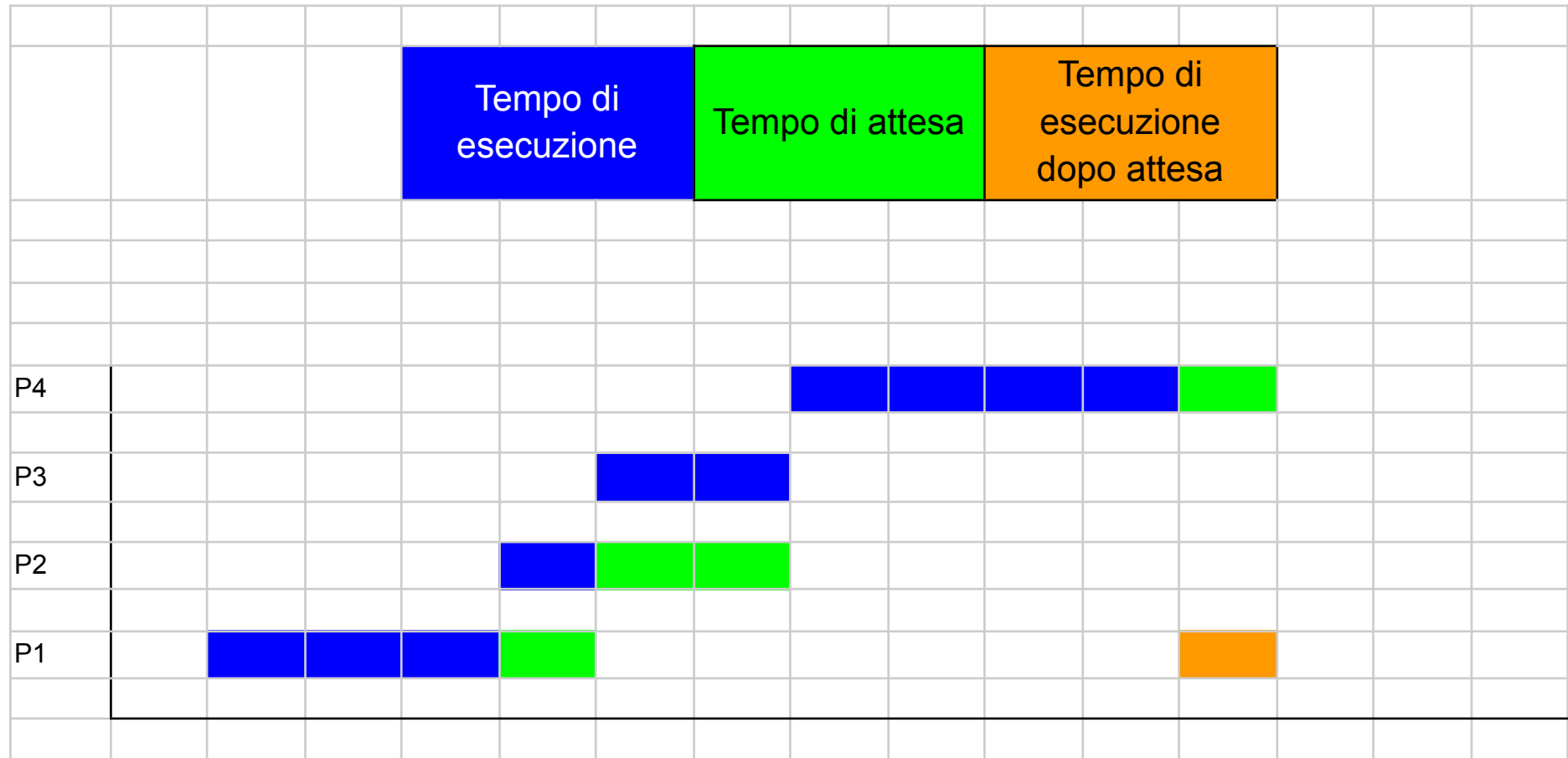
## Leggenda

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

## Mono-Tasking



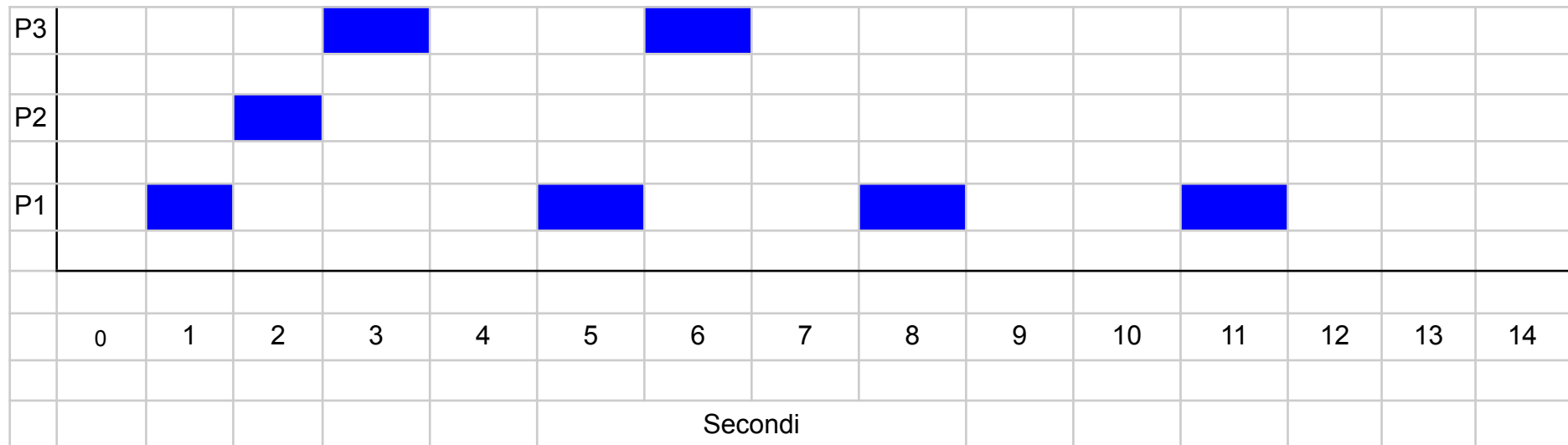
# Multi-Tasking



	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
							Secondi								

## Time-Sharing





## Analisi Specifica dei Dati

- Processo 1 richiede 3 secondi di esecuzione, con un'attesa di 1 secondo e un'ulteriore esecuzione di 1 secondo.
- P2 richiede 1 secondo di esecuzione con un'attesa di 2 secondi.
- P3 richiede 2 secondi di esecuzione senza attesa.
- P4 richiede 4 secondi di esecuzione con un'attesa di 1 secondo.

**Considerando questi dati, possiamo fare le seguenti considerazioni:**

- **Mono-Tasking** potrebbe causare un'attesa significativa per i processi P2, P3 e P4, soprattutto se Processo 1 è in esecuzione.
- **Multi-Tasking** può offrire un buon compromesso tra efficienza e reattività, gestendo contemporaneamente le risorse e riducendo il tempo di attesa complessivo.
- **Time-Sharing** sarebbe adatto se la priorità fosse la reattività e la riduzione del tempo di attesa percepito dagli utenti.

## **Conclusione**

**Il Multi-Tasking sembra essere il metodo più appropriato per gestire questi processi, poiché può bilanciare l'uso delle risorse della CPU e minimizzare i tempi di attesa. Tuttavia, se il sistema è semplice e non richiede molta interattività, il Mono-Tasking potrebbe essere sufficiente, mentre il Time-Sharing è ottimale per ambienti altamente interattivi e con molti processi concorrenti.**