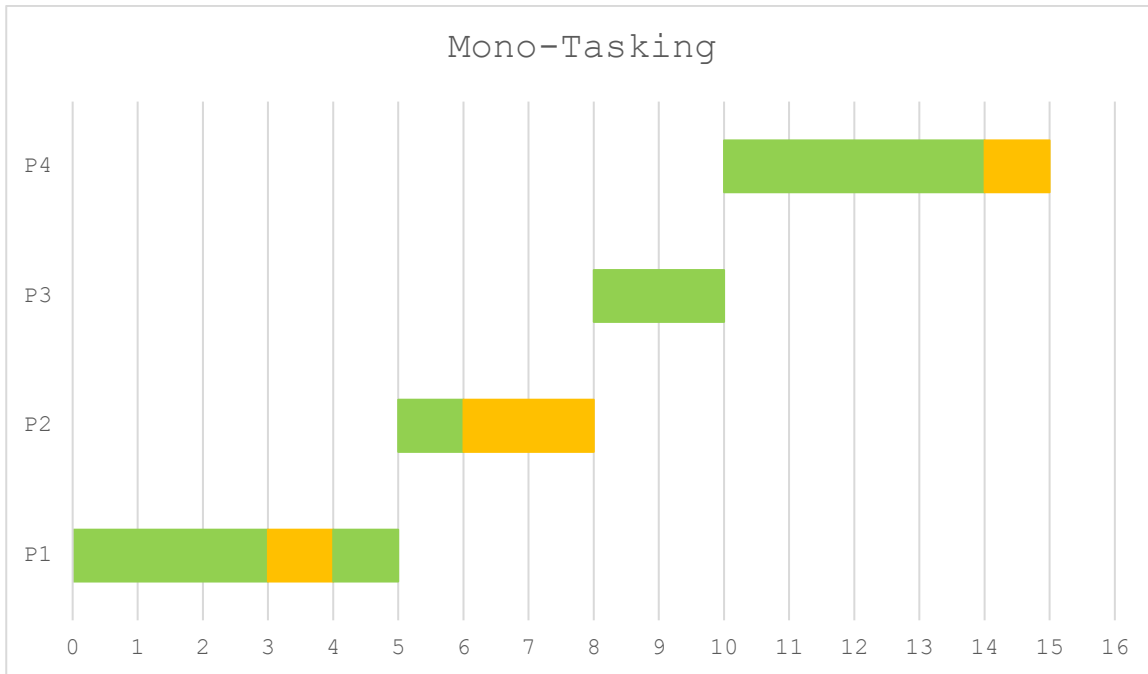
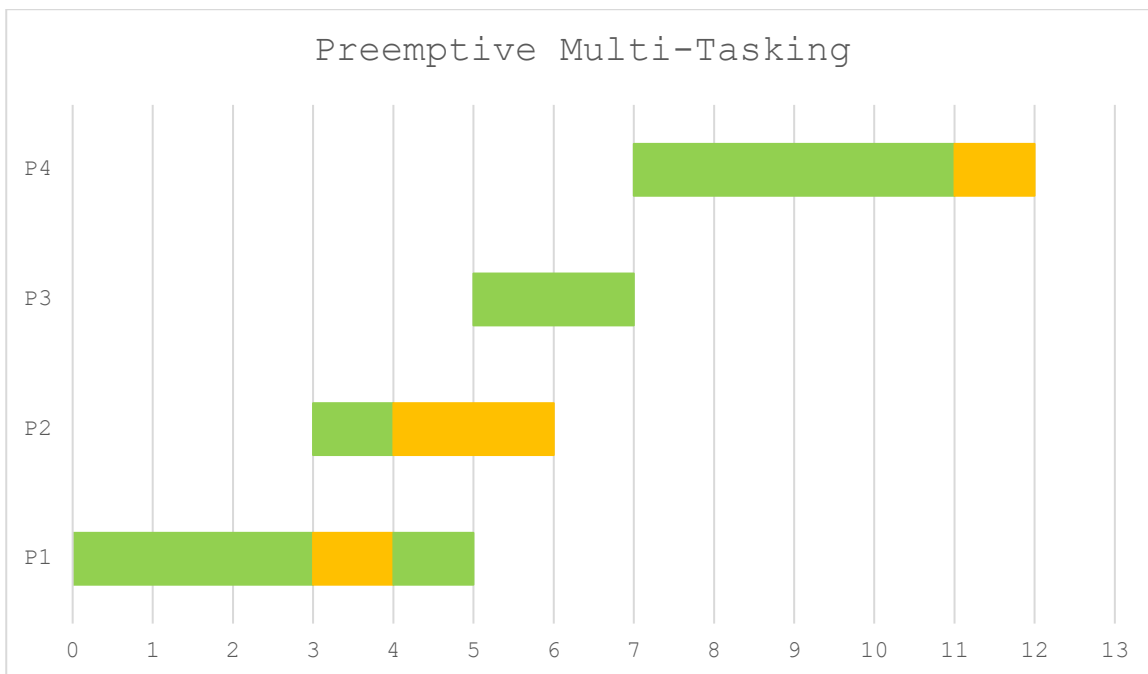


Scheduling della CPU

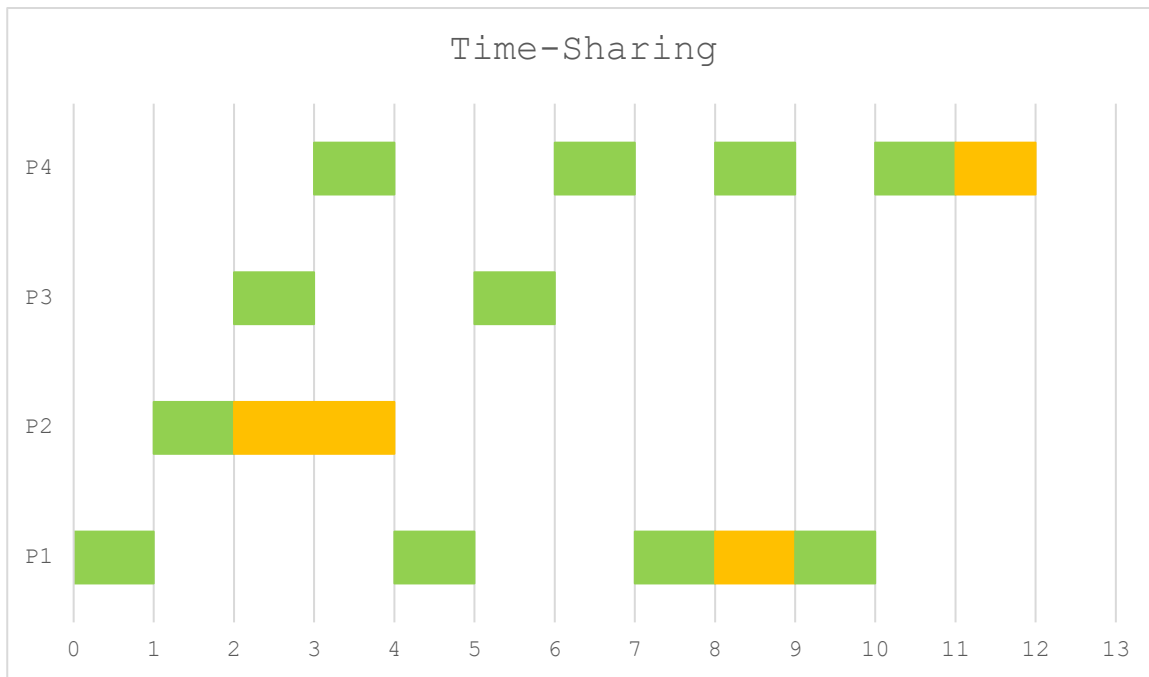
In questa esercitazione rappresenterò in grafici l'esecuzione di 4 processi con diverse metodologie:



Tempo di completamento in Mono-Tasking: 15 secondi

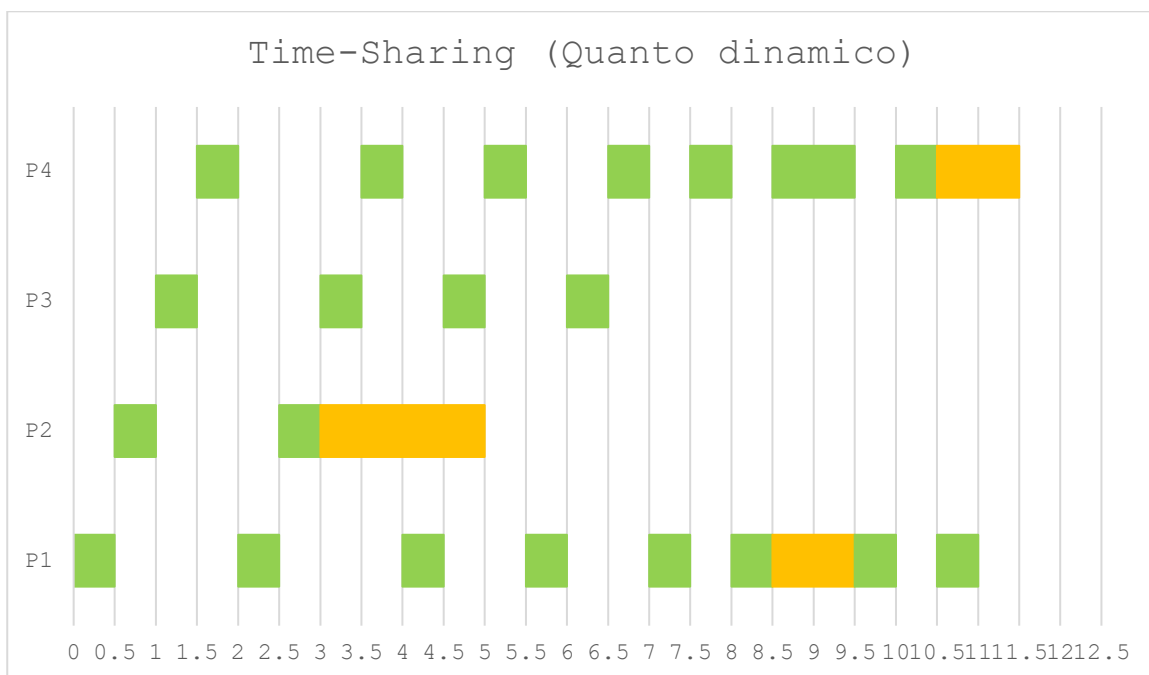


Tempo di completamento in Preemptive: 12 secondi



Tempo di completamento in Time-Sharing: 12 secondi

Di seguito riporto un caso di Time-Sharing in cui il Quanto è dinamico.[Report extra]



Tempo di completamento in Time-Sharing(Quanto dinamico): 11,5 secondi

Come tempo di Quanto ho applicato 0,5 secondi, tuttavia in questo caso può variare.

In Time-Sharing dal secondo 8,5 il P4 ha eseguito un Quanto di 1 secondo invece di 0,5 dato che il P1 era in attesa e lo Scheduler non era in nessun'altra task. Proprio per questo ho guadagnato 0,5 per il completamento. Questo sarebbe possibile solo se il Quanto fosse dinamico.

Conclusione

I processi più efficienti sono Multi-Tasking Preemptive e Time-Sharing. Il caso di Time-Sharing con Quanto dinamico è solo ipotetico.

Esercizio Facoltativo

In questo esercizio rappresenterò la tabella di esecuzione in cui vengono gestiti 5 processi, di seguito la tabella in cui si descrivono, per tutti e 5 i processi, il tempo di partenza (T0) e il tempo di esecuzione per finire il processo (Tx) il tutto in millisecondi.

Processo	Tempo di partenza (T0)	Tempo di esecuzione (Tx)
P1	0	14
P2	30	16
P3	6	40
P4	46	26
P5	22	28

Lo Scheduler in questo caso mette a disposizione Quanti di 12 millisecondi.

Risoluzione:

Time slice	Processo	Inizio(ms)	Fine(ms)	Rimanente(ms)	In coda
1	P1	0	12	2	P3(6x)
2	P3	12	24	28	P1(2) P5(22x)
3	P1-> FINE	24	26	0	P5(22x) P3(28)
4	P5	26	38	16	P3(28) P2(30x)
5	P3	38	50	16	P2(30x) P5(16) P4(46x)
6	P2	50	62	4	P5(16) P4(46x) P3(16)
7	P5	62	74	4	P4(46x) P3(16) P2(4)
8	P4	74	86	14	P3(16) P2(4) P5(4)
9	P3	86	98	4	P2(4) P5(4) P4(14)
10	P2-> FINE	98	102	0	P5(4) P4(14) P3(4)
11	P5-> FINE	102	106	0	P4(14) P3(4)
12	P4	106	118	2	P3(4)
13	P3-> FINE	118	122	0	P4(2)
14	P4-> FINE	122	124	0	

Nella colonna **In coda** ho scritto i processi che sono prossimi ad essere eseguiti, vengono messi in coda perché il loro tempo di arrivo è già passato oppure perché sono processi iniziati ma non ancora terminati, nel primo caso segno i millisecondi in cui sarebbe dovuto iniziare il processo con "x", nel secondo caso riporto solo i millisecondi rimanenti per terminare il processo in questione. La coda posiziona in precedenza il processo che si è inserito per prima in coda.

Nella colonna **Rimanente** ho scritto i millisecondi rimanenti per completare il processo nel time slice corrente (di quella riga).

Conclusione

Per completare tutti i processi ci sono voluti 124 millisecondi.