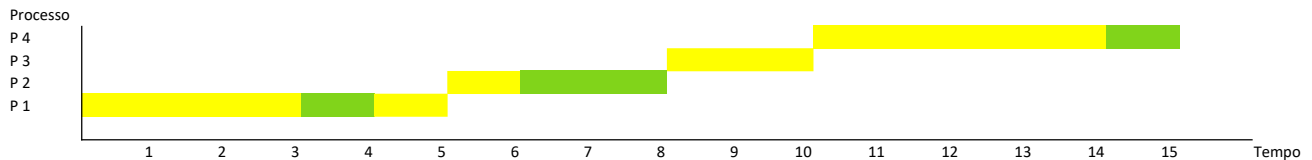


## DIFFERENZE TRA LE TIPOLOGIE DI PIANIFICAZIONE DEI PROCESSI DELLA CPU

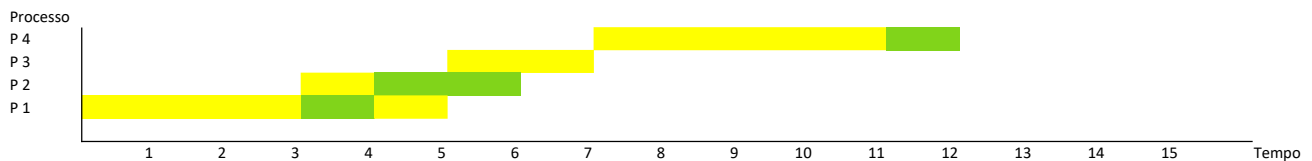
Le CPU utilizzate per sistemi come MS-DOS utilizzavano una pianificazione dei processi che permetteva l'esecuzione di esso uno alla volta, questa era detta **mono-tasking**. Durante l'esecuzione di un processo possono verificarsi dei tempi di attesa, potremmo dire che nel caso della pianificazione mono-tasking ci siano dei veri e propri tempi morti. Pertanto questa metodologia era piuttosto inefficiente.

### Sistema mono-tasking



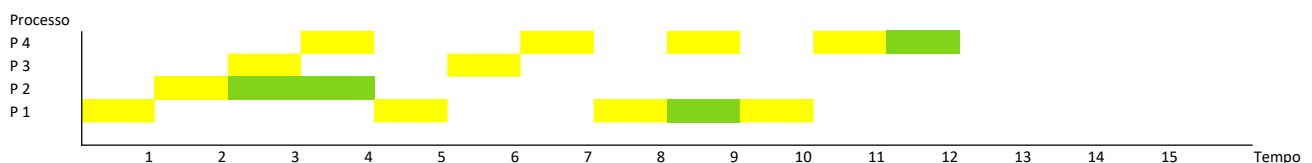
Con l'evoluzione dei sistemi operativi come Windows la gestione della CPU cambia permettendo l'esecuzione contemporanea di più programmi, questo sistema è detto **multi-tasking**. Il sistema multi-tasking utilizza un metodo di prelazione, quindi durante i periodi di attesa dell'esecuzione di un processo la CPU inizia a lavorare un altro processo per la durata dell'attesa.

### Sistema multi-tasking





Potremmo infine definire un aggiornamento del sistema di multi-tasking il sistema di **time-sharing**. Il time-sharing utilizza lo scheduling per dividere i processi in piccoli intervalli assegnando ad ognuno di esso una misura fissa detta "quanto di tempo". Con questo sistema si permette alla CPU una migliore gestione dei processi e a rendere l'interazione con l'utente più fluida.

### Sistema time sharing



Nei vari grafici su riportati possiamo notare le differenze delle tempistiche delle varie pianificazioni e come col cambiare delle metodologie c'è stata anche un'ottimizzazione delle tempistiche e dell'utilizzo della CPU diminuendo i tempi di inutilizzo

		Tempo di esecuzione	Tempo di attesa	Tempo di esecuzione Dopo attesa
Processo 1	P 1	3 secondi	1 secondo	1 secondo
Processo 2	P 2	1 secondo	2 secondi	/
Processo 3	P 3	2 secondi	/	/
Processo 4	P 4	4 secondi	1 secondo	/

	Tempo di esecuzione e utilizzo CPU
	Tempo di attesa

Nel seguente report, con una CPU che utilizza una politica di Round Robin e un time slice di 12 millisecondi andremo a simulare 5 processi in esecuzione.

			Time Slice	1	2	3	4	5	6	7	8	9	10	11	12	13
Tempi standard simulazione			Tempo totale In millisecondi	12	12	2	12	12	12	12	12	12	4	4	4	4
			Processo Eseguito	P1	P3	P1	P5	P3	P2	P5	P4	P3	P2	P5	P4	P3
Processo	T o	T x	Tf (Tempo di fine)	12	24	26	38	50	62	74	86	98	102	106	110	114
P1	0	14	To 0 Tx 2	To 0 Tx 2	Fine	/	/	/	/	/	/	/	/	/	/	/
P2	30	16	To 18 Tx 16	To 6 Tx 16	To 4 Tx 16	To 0 Tx 16	To 0 Tx 16	To 0 Tx 4	To 0 Tx 4	To 0 Tx 4	To 0 Tx 4	fine	/	/	/	/
P3	6	40	To 0 Tx 40	To 0 Tx 28	To 0 Tx 28	To 0 Tx 28	To 0 Tx 16	To 0 Tx 16	To 0 Tx 16	To 0 Tx 16	To 0 Tx 16	To 0 Tx 4	To 0 Tx 4	To 0 Tx 4	To 0 Tx 4	Fine
P4	46	26	To 34 Tx 26	To 22 Tx 26	To 20 Tx 26	To 8 Tx 26	To 0 Tx 26	To 0 Tx 26	To 0 Tx 26	To 8 Tx 14	To 0 Tx 14	To 0 Tx 4	To 0 Tx 4	To 0 Tx 4	Fine	/
P5	22	28	To 10 Tx 28	To 0 Tx 28	To 0 Tx 28	To 0 Tx 16	To 0 Tx 16	To 0 Tx 16	To 0 Tx 16	To 0 Tx 4	To 0 Tx 4	To 0 Tx 4	To 0 Tx 4	Fine	/	/
			In coda	P3	P1-P5	P5-P3	P3-P2	P2-P5-P4	P5-P4-P3	P4-P3-P2	P3-P2-P5	P2-P5-P4	P5-P4-P3	P4-P3	P3	FINE

Come possiamo vedere in tabella la Cpu, utilizzando la politica di Round Robin, gestisce i processi senza creare tempi di inutilizzo.  
Il Round Robin gestisce i processi in base al tempo di arrivo, quindi dando ad essi un lasso di tempo standard per l'esecuzione.  
Nel caso in cui il processo non dovesse completare la sua esecuzione ed avesse bisogno di più tempo esso viene messo in fase di attesa per poi essere messo in coda.  
In questo modo permette al prossimo processo di essere eseguito e la stessa cosa avviene con quest'ultimo, il secondo processo ha un tempo standard e se durante questo lasso di tempo non dovesse completare l'esecuzione viene interrotto e passato in coda per poter eseguire un terzo processo o ritornare al primo se dovesse essere l'unico in attesa.  
Ci sono alcuni casi invece dove i processi vengono interrotti perché hanno bisogno di qualche input dall'esterno, in questi casi la Cpu decide di metterlo in attesa e quindi in coda per permettere al successivo di ricominciare e non fermare il ciclo.

Leggenda della simulazione in tabella	
	Tempo di esecuzione e utilizzo CPU
	Tempo di attesa
	Il processo è stato eseguito ed è terminato

Calcolando il tempo di Turnaround medio in base alla nostra simulazione troviamo che esso è 70,8 millisecondi

Il tempo di attesa medio invece è di 46 millisecondi

Processo	T o	T x	Tf	Turnarou nd Tt=Tf-To	attesa Ta= Tt-Tx
P1	0	14	26	26	12
P2	30	16	102	72	56
P3	6	40	114	108	68
P4	46	26	110	64	38
P5	22	28	106	84	56
Totale				354	230
Media				70,8	46