

# Introduzione ai processi

---

## Index

- [Requisiti di un SO](#)
  - [Cos'è un processo?](#)
  - [Elementi di un processo](#)
    - [Process Control Block](#)
  - [Traccia di un processo](#)
    - [Esempio](#)
- 

## Requisiti di un SO

Il compito fondamentale di un sistema operativo è quello di **gestire i processi**, deve quindi gestire tutte le varie computazioni.

Un sistema operativo moderno deve quindi:

- permettere l'esecuzione alternata di processi multipli → anche se ho meno processori che processi questo non deve essere un problema, bisogna far alternare i processi in esecuzione sui processori disponibili
  - assegnare le risorse ai processi (es. un processo richiede l'uso della stampante)
  - permettere ai processi di scambiarsi informazioni
  - permettere la sincronizzazione tra processi
- 

## Cos'è un processo?

Un **processo** è un'istanza di un programma, memorizzato generalmente sul archiviazione di massa, in esecuzione (ogni singola istanza di fatti dà vita a un processo).

Potrebbe però anche essere visto come un'entità che può essere assegnata ad un processore in esecuzione tipicamente caratterizzata dall'esecuzione da una sequenza di istruzioni di cui voglio conoscere l'esito, uno stato corrente e da un insieme associato di risorse

Questo è quindi composto da:

- **codice** → ovvero le istruzioni da eseguire
- un insieme di **dati**
- un numero di **attributi** che descrivono il suo stato

Un processo ha 3 macrofasi: creazione, esecuzione, terminazione. Quest'ultima può essere prevista (quando il programma è terminato o quando l'utente chiude volontariamente in programma tramite la X) oppure non prevista (processo esegue un'operazione non consentita che potrebbe risultare con una terminazione involontaria del processo)

---

## Elementi di un processo

Finché un processo è in esecuzione ad esso sono associati un certo insieme di informazioni, tra cui:

- identificatore
- stato (running etc.)
- priorità
- hardware context → attuale situazione dei registri
- puntatori alla memoria principale (definisce l'immagine del processo)
- informazioni sullo stato dell'input/output
- informazioni di accounting (quale utente ha eseguito il processo)

## Process Control Block

Per ciascuno processo attualmente in esecuzione è presente un **process control block**. Si tratta di un insieme di informazioni (gli elementi di un processo) raccolte insieme e mantenute nella zona di memoria riservata al kernel.

Questo viene interamente creato e gestito dal sistema operativo e il suo scopo principale è quello di permettere al SO di **gestire più processi contemporaneamente** (contiene infatti le informazioni sufficienti per bloccare un programma e farlo riprendere più tardi dallo stesso punto in cui si trovava)

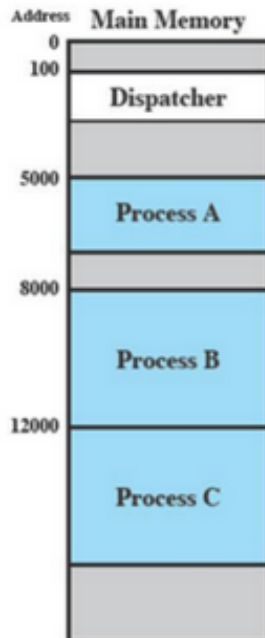
---

## Traccia di un processo

Un ulteriore aspetto importante in un processo è la **trace** ovvero l'insieme di istruzioni di cui è costituito un processo. Il **dispatcher** invece è un piccolo programma che sospende un processo per farne andare un altro in esecuzione

## Esempio

Si considerino 3 processi in esecuzioni, tutti caricati in memoria principale



La traccia, dal punto di vista del processo, appare come l'esecuzione sequenziale delle istruzioni del singolo processo

5000	8000	12000
5001	8001	12001
5002	8002	12002
5003	8003	12003
5004		12004
5005		12005
5006		12006
5007		12007
5008		12008
5009		12009
5010		12010
5011		12011

(a) Trace of Process A    (b) Trace of Process B    (c) Trace of Process C

La traccia, del punto di vista del processore, ci mostra come effettivamente vengono eseguiti i 3 processi

1	5000	
2	5001	
3	5002	
4	5003	
5	5004	
6	5005	
----- Timeout		
7	100	
8	101	
9	102	
10	103	
11	104	
12	105	
13	8000	
14	8001	
15	8002	
16	8003	
----- I/O Request		
17	100	
18	101	
19	102	
20	103	
21	104	
22	105	
23	12000	
24	12001	
25	12002	
26	12003	
----- Timeout		
27	12004	
28	12005	
----- Timeout		
29	100	
30	101	
31	102	
32	103	
33	104	
34	105	
35	5006	
36	5007	
37	5008	
38	5009	
39	5010	
40	5011	
----- Timeout		
41	100	
42	101	
43	102	
44	103	
45	104	
46	105	
47	12006	
48	12007	
49	12008	
50	12009	
51	12010	
52	12011	
----- Timeout		

 Le righe in blu sono gli indirizzi del dispatcher