

Laboratorio di Sistemi  
Operativi  
**Simulatori**

# Simulatori

- **SGPEMv2 della Dueffe**

- Simulatore di Gestione di Processi in un Elaboratore Programmato.
- Analizza l'ordinamento dei processi e dei meccanismi di gestione delle risorse in un elaboratore.

- **SiGeM della Stylosoft**

- Simulatore per lo studio dei meccanismi di gestione della memoria in un elaboratore monoprocessore multiprogrammato
- Simula il comportamento di vari algoritmi di rimpiazzo delle pagine e segmenti, sulla base dei dati specificati.

# Politiche di ordinamento dei processi (**SGPEMv2**)

# Politiche di ordinamento

- First-Come-First-Served (FCFS)
  - Senza prerilascio e senza priorità
- Shortest Job First (SJF)
  - Senza prerilascio
  - Richiede conoscenza dei tempi di esecuzione
  - Non equo con i processi non presenti all'inizio
- Shortest Remaining Time Next (SRTN)
  - Versione SJF con prerilascio
  - Equo con i processi non presenti all'inizio
- Round Robin (RR)
  - Con prerilascio senza priorità
- Round Robin con Priorità (RRP)
- Lotteria (Senza Garanzia) - Sistemi interattivi
  - Ogni processo riceve dei numeri da giocare
  - Priorità più alta = più numeri assegnati
  - Estrazioni periodiche

# Criteri di valutazione

- Tempo di attesa
  - Durata totale di attesa di un processo in stato di pronto
- Tempo di turn around
  - Tempo di completamento
- Tempo di risposta
  - Reattività rispetto alla richiesta di avvio di un processo

# Esempio 1

Abbiamo 3 processi: P0, P1 e P2.

- **P0**
  - **Tempo di arrivo: 0**
  - **Durata: 2**
- **P1**
  - **Tempo di arrivo: 1**
  - **Durata: 12**
- **P2**
  - **Tempo di arrivo: 3**
  - **Durata: 4**

1. Utilizzando la politica FCFS, trovare:
  - a. Il tempo di attesa medio
  - b. Il tempo di Turn Around
  - c. Il tempo di risposta

## Esempio 2

Cinque processi in batch P0, P1, P2, P3, P4 arrivano alla macchina agli istanti 0, 1, 2, 6, 7. I processi hanno un tempo stimato di 3, 7, 2, 3, 1 unità rispettivamente. Determinare:

- Il tempo medio di turn around
- Il tempo medio di attesa
- Il tempo di risposta

Utilizzare la politica Round Robin, time slice = 2.

	P0	P1	P2	P3	P4
Arrivo	0	1	2	6	7
Durata	3	7	2	3	1

# Esercizio

Analizzare le 3 politiche di ordinamento dei processi, configurando la simulazione con 7 processi P0, P1, P2, P3, P4, P5, P6:

**I processi P1, P3 e P5 appartengono ad una classe con maggiore priorità (numero più grande = maggiore priorità) con prerilascio.**

Utilizzando le politiche:

- **FCFS**
- **SJF**
- **RR con priorità (time slice: 2)**

Calcolare:

- a. Tempo di risposta medio
- b. Tempo di attesa medio
- c. Tempo di turn-around medio

# Esercizio

Processo	Tempo di arrivo	Durata esecuzione
P0	0	4
<b>P1</b>	1	3
P2	1	4
<b>P3</b>	2	2
P4	2	5
<b>P5</b>	5	2
P6	8	6

- FCFS
- SJF
- RR con priorità (time slice: 2)

1. Qual è la politica con i migliori tempi di completamento?

1. Qual è la miglior politica nel caso di sistemi interattivi?

1. Qual è la relazione tra tempo di risposta e tempo di attesa con FCFS e SJF?

# Gestione della memoria (**SiGeM**)

# Gestione della memoria virtuale

- Il gestore della memoria cerca di soddisfare le esigenze di memoria dei processi.
- In generale la memoria disponibile è inferiore a quella necessaria per tutti i processi attivi contemporaneamente.
- Oggi, anche l'intera memoria primaria può non essere sufficiente per ospitare un solo processo.
- La **memoria virtuale** è un concetto basato sul caricare nella memoria primaria solo la parte strettamente necessaria all'esecuzione corrente.
- Questa tecnica simula il possesso una RAM più grande di quella disponibile.
- Esistono due tecniche di gestione della memoria virtuale
  - **Paginazione**
  - Segmentazione
- La sfida è **minimizzare i tempi di accesso** al disco secondario, cercando di trovare la migliore politica di accesso alle pagine/segmenti.

# Paginazione e *page fault*

- La paginazione suddivide la memoria primaria in unità a dimensione fissa dette **pagine**.
- Queste pagine vengono caricate in RAM quando richieste e restano fino a quando non vengono rimpiazzate.
- Quando una pagina è assente quando riferita (cioè non è presente nella RAM), si genera un ***page fault***.
- Quando si ha un ***page fault***, il sistema operativo deve rimpiazzare una pagina, salvando su disco la pagina rimossa e inserendo quella nuova.
- Il rimpiazzo ottimale non è possibile, perché il sistema operativo non sa a priori quali pagine il processo richiederà.

# Politiche di rimpiazzo

- **Not Recently Used (NRU)**
  - 1 bit **R**: pagina **R**iferita durante l'esecuzione (1) o no (0)
  - 1 bit **M**: pagina **M**odificata durante l'esecuzione (1) o no (0)
  - I bit vengono aggiornati periodicamente
  - L'ordinamento delle pagine avviene nel seguente modo:
    - Classe 0: non riferita, non modificata
    - Classe 1: non riferita, modificata
    - Classe 2: riferita, non modificata
    - Classe 3: riferita, modificata
  - Viene scelta una pagina a caso nella classe con indice minore
- **First In First Out (FIFO)**
  - Rimuove la pagina più vecchia in RAM (pagine ordinate per tempo di arrivo in RAM)
- **Second chance (SC)**
  - FIFO ma rimpiazza solo le pagine con bit  $R = 0$
- **Clock (C)**
  - Come SC ma i page frame sono inseriti in una lista circolare

# Politiche di rimpiazzo

- Least Recently Used (LRU)
  - Tiene traccia delle pagine più usate dai processi
  - Basato sulla probabilità che pagine molto usate, verranno usate ancora.
  - Oneroso perché richiede un riordinamento ad ogni riferimento.
- Not Frequently Used (NFU)
  - Versione approssimata di LRU
  - Ogni pagina ha un contatore che viene incrementato se la pagina è utilizzata
  - La pagina che ha il contatore con valore minore viene rimpiazzata.
  - Tiene conto della frequenza di utilizzo, ma non del tempo di permanenza nella RAM.
- Aging (A)
  - NFU modificato
  - Tiene conto del tempo di permanenza nella RAM.
  - Ad ogni scadenza del clock si effettua uno shift a destra del contatore e si accosta a sinistra il bit R.
  - Con N bit, perde memoria dopo N aggiornamenti.

# Esempio

Consideriamo un processo con tempo di esecuzione 20 unità e 7 pagine in memoria. I dati relativi all'elaboratore sono:

Dimensione RAM	4 KB
Dimensione Area di swap	8
Tempo di context switch	1
Dimensione pagina	1
Tempo di accesso al disco	1
Banda del bus di dati	1
Tecnica di gestione della memoria	pagine

Le pagine richieste nei vari istanti sono indicati in tabella. In rosso le pagine modificate.

Istante	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
Richiesta	0	1	2	3	0	3	1	4	3	0	5	4	3	0	6	0	4	3	4	0

Utilizzando una politica FIFO, trovare il numero di page fault totali. E con LRU?

# Esercizio

Consideriamo 3 processi con tempo di esecuzione 6 unità di tempo ciascuno.

I dati dell'elaboratore e dei processi sono riportati in tabella (11 pagine)

Processo	T. arrivo	T. esec.	Richiesta al tempo 0	Richiesta al tempo 1	Richiesta al tempo 2	Richiesta al tempo 3	Richiesta al tempo 4	Richiesta al tempo 5
P1	0	6	0, 4, 5	0, 1, 3, 9	1, 2, 6	0, 1, 5	0, 7, 8	0, 1
P2	1	6	0, 2, 7	0, 1, 4	1, 8, 5	0, 6	1, 3	3, 8
P3	0	6	1, 2, 3	1, 4, 5	0, 6, 7	1, 8, 9, 10	8, 10	1

Dimensione RAM	64 KB
Dimensione Area di swap	64 KB
Dimensione pagina	4 KB
Tempo di context switch	1
Tempo di accesso al disco	1
Banda del bus di dati	1
Tecnica di gestione della memoria	pagine

Utilizzando la politica RR per i processi (time slice = 2) calcolare il numero di page fault utilizzando:

- FIFO
- LRU
- NFU

Qual'è il migliore?

**Per domande e dubbi:**

**e-mail**

**[dronzani@math.unipd.it](mailto:dronzani@math.unipd.it)**