

esercizi

SCHEDULING DEI PROCESSI

6-bis. Esercizi sullo scheduling

1

marco lapeagna

riepilogo

- **Algoritmo First Come First Served (FCFS)** : i processi vengono eseguiti nell'ordine di arrivo
- **Algoritmo Shortest Job First (SJF)** : i processi vengono eseguiti da quello con il minor al maggior tempo di esecuzione rimanente
- **Algoritmo con priorita'** : i processi vengono eseguiti secondo un ordine definito da una priorita'
- **Algoritmo Round Robin (RR)** : viene assegnato un quanto di tempo ad ogni processo, scaduto il quale la cpu viene assegnata al successivo processo

6-bis. Esercizi sullo scheduling

2

marco lapeagna

riepilogo

- Algoritmo First Come First Served (FCFS) : **e' senza prelazione**
- Algoritmi (SJF) e con priorita' possono essere
 - **Con prelazione**
 - **Senza prelazione**
- Algoritmo Round Robin (RR) : **e' sempre con prelazione**

6-bis. Esercizi sullo scheduling

3

marco lapeagna

Esercizio 1:

Siano i seguenti processi nella ready queue, arrivati nell'ordine specificato e tutti presenti ad un dato istante 0

Proc.	Esecuz.	prior.
P ₁	10	3
P ₂	1	1
P ₃	2	3
P ₄	1	4
P ₅	5	2

- si considerino i seguenti algoritmi di scheduling:
- **FCFS**
 - **SJF senza prelazione**
 - **priorita' senza prelazione (p=0 pr. Max)**
 - **Round robin con q=1**

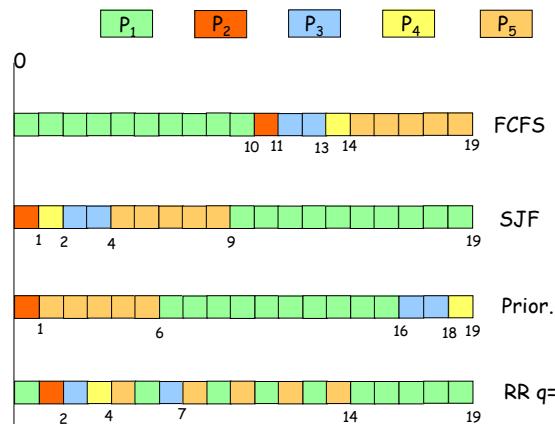
1. Si disegnino i **4 diagrammi** di Gantt
2. Calcolare i **tempi di completamento** nei 4 casi
3. Calcolare il **tempo medio di completamento** nei 4 casi
4. Calcolare la deviazione standard nei casi SJF e RR

6-bis. Esercizi sullo scheduling

4

marco lapeagna

Diagrammi di Gantt



6-bis. Esercizi sullo scheduling

5

marco lapeagna

Tempi di completamento e medi

- FCFS**
 - $P_1 \rightarrow 10 \quad P_2 \rightarrow 11 \quad P_3 \rightarrow 13 \quad P_4 \rightarrow 14 \quad P_5 \rightarrow 19$
 - $T_m = 67/5 = 13.4$
- SJF**
 - $P_1 \rightarrow 19 \quad P_2 \rightarrow 1 \quad P_3 \rightarrow 4 \quad P_4 \rightarrow 2 \quad P_5 \rightarrow 9$
 - $T_m = 35/5 = 7 \quad s = 6.6$
- Priorita'**
 - $P_1 \rightarrow 16 \quad P_2 \rightarrow 1 \quad P_3 \rightarrow 18 \quad P_4 \rightarrow 19 \quad P_5 \rightarrow 6$
 - $T_m = 60/5 = 12$
- RR q=1**
 - $P_1 \rightarrow 19 \quad P_2 \rightarrow 2 \quad P_3 \rightarrow 7 \quad P_4 \rightarrow 4 \quad P_5 \rightarrow 14$
 - $T_m = 46/5 = 9.2 \quad s = 6.3$

6-bis. Esercizi sullo scheduling

6

marco lapeagna

Esercizio 2:

Siano i seguenti processi nella ready queue, arrivati nei tempi specificati

Proc.	arrivo	esecuz.	si considerino i seguenti algoritmi di scheduling:
P_1	0.0	8	▪ FCFS senza prelazione
P_2	0.5	4	▪ SJF senza prelazione
P_3	1.0	1	

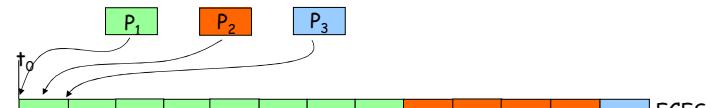
1. Si disegnino i **diagrammi di Gantt**
2. Calcolare i **tempi di completamento** nei 2 casi
3. Calcolare il **tempo medio di completamento** nei 2 casi

6-bis. Esercizi sullo scheduling

7

marco lapeagna

soluzione



- FCFS**
 - $P_1 \rightarrow 8 \quad P_2 \rightarrow 11.5 \quad P_3 \rightarrow 12$
 - $T_m = 31.5/3 = 10.5$
- SJF**
 - $P_1 \rightarrow 8 \quad P_2 \rightarrow 12.5 \quad P_3 \rightarrow 13$
 - $T_m = 28.5/3 = 9.5$

6-bis. Esercizi sullo scheduling

8

marco lapeagna

Esercizio 3:

Siano i seguenti processi nella ready queue, arrivati nell'ordine specificato e tutti presenti ad un dato istante 0

Proc.	Esecuz.	.	si considerino l'algoritmo RR con i seguenti quanti di tempo:
P ₁	10		▪ q=1
P ₂	1		▪ q=2
P ₃	2		▪ q=5
P ₄	1		▪ q=10
P ₅	5		

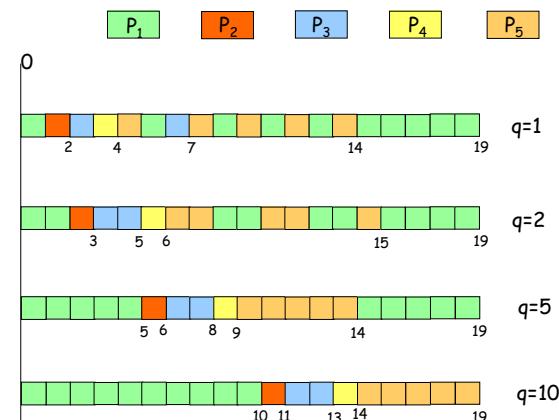
1. Si disegnino i **4 diagrammi di Gantt**
2. Calcolare i **tempi di completamento** nei 4 casi
3. Calcolare il **tempo medio di completamento** nei 4 casi

6-bis. Esercizi sullo scheduling

9

marco lapeagna

Diagrammi di Gantt



6-bis. Esercizi sullo scheduling

10

marco lapeagna

Tempi di completamento e medi

- **q=1**
 - P₁ → 19 P₂ → 2 P₃ → 7 P₄ → 4 P₅ → 14
 - T_m = 46/5 = **9.2**
- **q=2**
 - P₁ → 19 P₂ → 3 P₃ → 5 P₄ → 6 P₅ → 15
 - T_m = 48/5 = **9.6**
- **q=5**
 - P₁ → 19 P₂ → 6 P₃ → 8 P₄ → 9 P₅ → 14
 - T_m = 56/5 = **11.2**
- **q=10**
 - P₁ → 10 P₂ → 11 P₃ → 13 P₄ → 14 P₅ → 19
 - T_m = 67/5 = **13.4**

6-bis. Esercizi sullo scheduling

11

marco lapeagna

quindi

q piccolo → tempi medi di esecuzione piccoli
q grande → tempi medi di esecuzione grandi

MA

Non si e' considerato il tempo di cambio di contesto



Si ripeta l'esercizio considerando il tempo di cambio di contesto **τ=1**

6-bis. Esercizi sullo scheduling

12

marco lapeagna

Tempi medi di completamento con c.s.

$q=1$

- P1 soffre di 14 cs, P2 di 1, P3 di 6, P4 di 3 e P5 di 13 cs
- Tempo medio di compl. $(46+37)/5 = 83/5 = 16.6$

$q=2$

- P1 soffre di 9 cs, P2 di 1, P3 di 2, P4 di 3 e P5 di 8 cs
- Tempo medio di compl. $(48+23)/5 = 71/5 = 14.2$

$q=5$

- P1 soffre di 5 cs, P2 di 1, P3 di 2, P4 di 3 e P5 di 4 cs
- Tempo medio di compl. $(56+15)/5 = 71/5 = 14.2$

$q=10$

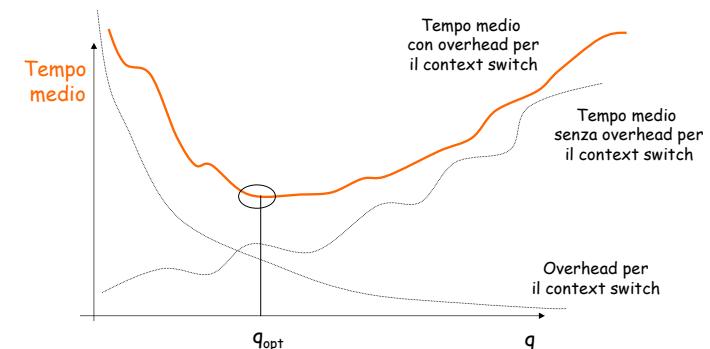
- P1 soffre di 0 cs, P2 di 1, P3 di 2, P4 di 3 e P5 di 4 cs
- Tempo medio di compl. $(67+10)/5 = 77/5 = 15.4$

6-bis. Esercizi sullo scheduling

13

marco lapeagna

In generale



6-bis. Esercizi sullo scheduling

14

marco lapeagna

Esercizio 4:

Siano i seguenti processi nella ready queue, arrivati nell'ordine specificato

Proc. arrivo. esec.

P_1	0	7	si consideri l'algoritmo SJF
P_2	2	4	▪ senza prelazione
P_3	3	4	▪ con prelazione
P_4	5	2	
P_5	7	3	
P_6	10	2	

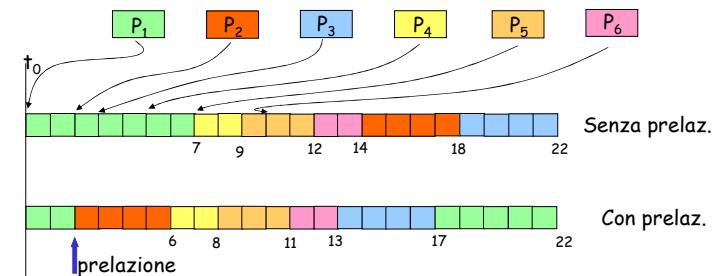
- Si disegnino i diagrammi di Gantt
- Calcolare i tempi di completamento nei 2 casi
- Calcolare il tempo medio di completamento nei 2 casi

6-bis. Esercizi sullo scheduling

15

marco lapeagna

soluzione



Senza prelazione

- $P_1 \rightarrow 7 \quad P_2 \rightarrow 16 \quad P_3 \rightarrow 19 \quad P_4 \rightarrow 4 \quad P_5 \rightarrow 5 \quad P_6 \rightarrow 4$
- $T_m = 55/6 = 9.16$

Con prelazione

- $P_1 \rightarrow 22 \quad P_2 \rightarrow 4 \quad P_3 \rightarrow 14 \quad P_4 \rightarrow 3 \quad P_5 \rightarrow 4 \quad P_6 \rightarrow 3$
- $T_m = 50/6 = 8.33$

6-bis. Esercizi sullo scheduling

16

marco lapeagna

Esercizio 5:

Siano i seguenti processi presenti al tempo 0

Proc. esec.

P_1	10	Si usi scheduling a code multiple
P_2	50	▪ coda Q1 $\rightarrow P_1, P_2, P_3$ con scheduling SJF s.p.
P_3	40	▪ coda Q2 $\rightarrow P_4, P_5$ con scheduling RR ($q_1=20$)
P_4	50	▪ scheduling tra le code RR ($q_2=30$)
P_5	30	

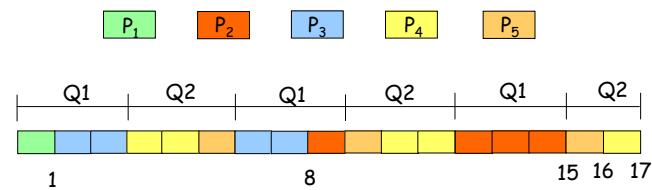
1. Si disegni il diagramma di Gantt
2. Calcolare i tempi di completamento
3. Calcolare il tempo medio di completamento

6-bis. Esercizi sullo scheduling

17

marco lapeagna

soluzione



- $P_1 \rightarrow 1$ $P_2 \rightarrow 15$ $P_3 \rightarrow 8$ $P_4 \rightarrow 17$ $P_5 \rightarrow 16$
- $T_m = 57/5 = 11.4$

6-bis. Esercizi sullo scheduling

18

marco lapeagna

Esercizio 6:

Siano i seguenti processi

Proc. esec. arrivo

P_1	9	0	Si usi scheduling
P_2	6	2	▪ FCFS
P_3	5	4	▪ SJF con prelazione
P_4	2	5	
P_5	1	8	Fornire i relativi diagrammi di Gantt

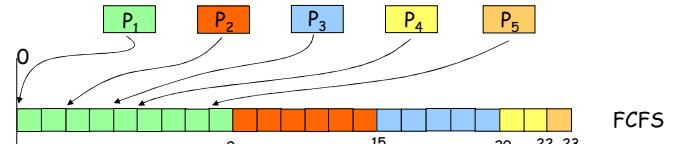
Nel caso di scheduling RR con $q=10$ ms, quanto deve durare il context switch per far terminare tutti i processi in 45 ms?

6-bis. Esercizi sullo scheduling

19

marco lapeagna

soluzione



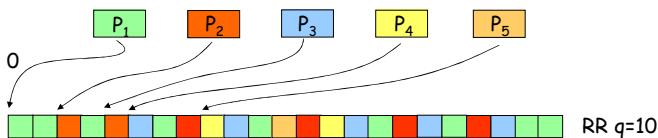
- FCFS
 - $P_1 \rightarrow 9$ $P_2 \rightarrow 13$ $P_3 \rightarrow 16$ $P_4 \rightarrow 16$ $P_5 \rightarrow 15$
 - $T_m = 69/5 = 13.8$
- SJF Con prelazione
 - $P_1 \rightarrow 23$ $P_2 \rightarrow 9$ $P_3 \rightarrow 12$ $P_4 \rightarrow 2$ $P_5 \rightarrow 1$
 - $T_m = 47/5 = 9.4$

6-bis. Esercizi sullo scheduling

20

marco lapeagna

soluzione



Sia x il tempo del context switch

20 context switches \rightarrow tempo = $20x$ ms
Tempo esec dei processi = 23 ms

Tempo totale di esecuzione = $23 + 20x$

$$23 + 20x < 45 \rightarrow x < 22/20 = 1.1$$

note

▪ SJF con prelazione

- T=2 arriva P2 con 6 u.t. mentre a P1 restano 7 u.t. \rightarrow Prelazione
- T=4 arriva P3 con 5 u.t. mentre a P2 restano 4 u.t. \rightarrow No prelazione
- T=5 arriva P4 con 2 u.t. mentre a P2 restano 3 u.t. \rightarrow Prelazione
- T=7 finisce P4 va in esecuzione P2 con 3 u.t.
- T=8 arriva P5 con 1 u.t. mentre a P2 restano 2 u.t. \rightarrow Prelazione
- T=9 finisce P5 e vanno in esecuzione nell'ordine P2, P3 e P1 fino al loro completamento

▪ RR con q=10

- T=2 supponiamo che P2 arriva un attimo prima della scadenza del quanto ed entra nella coda
- T=4 supponiamo che P3 arriva un attimo prima della scadenza del quanto ed entra nella coda. Allo scadere del quanto, P2 va in esecuzione ed esce dalla coda mentre entra P1

Scheduler di Linux



- Vengono schedulati i **threads**
- Tre classi: **real time FCFS**, **real time Round Robin**, **Time sharing**
- Ogni thread ha:
 - Priorità variabile p nell'intervallo $[0, 40]$, dove $p=0$ è la priorità minima. Generalmente all'inizio $p=20$
 - Un credito c (quanti di 10 ms da spendere)
- Lo scheduler calcola la **goodness (gdn)** di ogni thread come
 - Se (class == realtime) $gdn=1000 + p$
 - Se (class == timesharing & $c > 0$) $gdn = c+p$
 - Se (class == timesharing & $c == 0$) $gdn = 0$

Scheduler di Linux

- Ogni volta viene selezionato il **thread con gdn maggiore**
- Ogni 10 ms c viene decrementato
- Il **thread viene prelazionato** quando:
 - c diventa 0
 - Il thread si blocca
 - Nella ready queue entra un thread con gdn maggiore
- Quando $c = 0$, per tutti i processi nella ready queue viene ricalcolato c come segue:
 - $c = c/2 + p$

Esercizio

- Sia un [sistema Linux](#) dove nella ready queue sono presenti i seguenti threads:
 - A: $c=5$, $p=8$, class = timesharing
 - B: $c=15$, $p=4$, class = timesharing
 - C: $c=9$, $p=X$, class= Y
 - D: $c=7$, $p=15$, class = timesharing

Se B e' stato l'ultimo thread ad andare in esecuzione, si discuta in funzione di X e Y quale thread sara' il prossimo ad andare in esecuzione

Soluzione

- Se $Y==\text{realtime}$ allora
 - A: $gdn = 13$
 - B: $gdn = 19$
 - C: $gdn = 1000 + X$ ← goodness maggiore qualunque sia X
 - D: $gdn = 22$
- Se $Y==\text{timesharing}$ allora
 - A: $gdn = 13$
 - B: $gdn = 19$
 - C: $gdn = 9 + X$
 - D: $gdn = 22$