

I Prova in itinere – 5 Novembre 2018  
**(teoria)**

Università di Salerno

1. **Codice comportamentale.** Durante questo esame si deve lavorare da soli. Non si può consultare materiale di nessun tipo. Non si può chiedere o dare aiuto ad altri studenti.
2. **Istruzioni.** Rispondere alle domande. Per la brutta usare i fogli posti alla fine del plico (NON si possono usare fogli aggiuntivi); le risposte verranno corrette solo se inserite nello spazio ad esse riservate oppure viene indicata con chiarezza la posizione alternativa.  
Per essere accettata per la correzione la risposta deve essere ordinata e di facile lettura.  
TUTTE le risposte vanno GIUSTIFICATE. Ciascuna risposta non giustificata vale ZERO.

Nome e Cognome: \_\_\_\_\_

Matricola: \_\_\_\_\_

Firma \_\_\_\_\_

Spazio riservato alla correzione: non scrivere in questa tabella.

1	2	Tot	bonus
/30	/20	/50	/10

**1. 30 punti**

Un hard disk ha la capienza di  $2^{42}$  byte ed è formattato in blocchi da 1Kb.

Si assuma che un file **pluto** la cui taglia è 300Kb sia allocato su tale hard disk, che il suo FCB sia già presente in memoria principale e che  $b$  sia il numero del primo blocco di **pluto**.

Giustificando le risposte, rispondere ai quesiti seguenti.

1) Assumendo che:

•• lo spazio libero sia gestito attraverso una lista linkata, dove  $c$  è il numero del primo blocco della lista, e che

•• sia adottata allocazione contigua per i file  
dire

1a) "quanti accessi a disco" sono necessari e

1b) "come viene eventualmente modificata la lista linkata dei blocchi liberi" e

1c) come vengono modificate le informazioni di **pluto** nel FCB (relativamente al recupero dei suoi blocchi),

nel caso si voglia **cancellare il primo blocco di pluto** ed **modificare il contenuto dell'ultimo blocco di pluto**

2) Assumendo che:

• lo spazio libero sia gestito attraverso una bitmap (vettore di bit), già presente in memoria principale, e che

• sia adottata allocazione linkata per i file  
dire

2a) "quanti accessi a disco" sono necessari e

2b) "come viene eventualmente modificata la bitmap" e

2c) come vengono modificate le informazioni di pluto nel FCB (relativamente al recupero dei suoi blocchi),

nel caso si voglia **cancellare l'ultimo blocco di pluto** ed **modificare il contenuto del primo blocco di pluto**

3) Assumendo che:

•• si adotti una organizzazione del filesystem simile a Unix, dove il FCB sia del tipo seguente:  
attributi

ind. blocco 0

ind. blocco 1

ind. blocco 2

ind. blocco indirizzi indirezione singola

ind. blocco indirizzi indirezione singola

(3.1) dire "quanti blocchi" sono necessari per memorizzare **pluto** (compresi eventuali blocchi indice)

(3.2) "quanti accessi a disco" sono necessari per modificare l'ultimo blocco di **pluto**

(3.3) "quanti blocchi" liberi devo recuperare se volessi che **pluto** occupi lo spazio massimo per tale tipo di FCB.

4) Assumendo che:

- una FAT sia usata sia per l'allocazione dei file di tale sistema che per la gestione dei blocchi liberi, e che
- $c$  sia il primo blocco libero

Dato il seguente frammento di FAT,

Entry	Contenuto
$b - 3$	5
$b - 2$	2
$b - 1$	$b - 3$
$b$	$b + 5$
$b + 1$	$b + 7$
$b + 2$	8
$b + 3$	7
$b + 4$	$b - 2$
$b + 5$	$b - 1$
$c$	$c + 2$
$c + 1$	$c + 4$
$c + 2$	10

dire

- 1a) "quanti accessi a disco" sono necessari e
- 1b) "come viene eventualmente modificata la FAT" e
- 1c) come vengono modificate le informazioni di pluto nel FCB (relativamente al recupero dei suoi blocchi),  
nel caso si voglia **modificare il contenuto del secondo blocco di pluto e cancellare il primo blocco di pluto.**

NOTA: Si assume che un blocco da aggiungere ai blocchi liberi viene aggiunto all'inizio.

## 2. 20 punti

In un sistema lo scheduling della CPU è gestito mediante 2 code multiple denominate A, B, di cui la coda A ha priorità maggiore rispetto alla coda B, cioè se ci sono processi nella coda a priorità maggiore non possono essere eseguiti processi presenti in una coda a priorità inferiore.

Inoltre, gli algoritmi di scheduling adottati all'interno di ciascuna coda sono i seguenti:

- la coda A adotta lo SJF con prelazione;
- la coda B adotta il RR con quanto di tempo di 3msec.

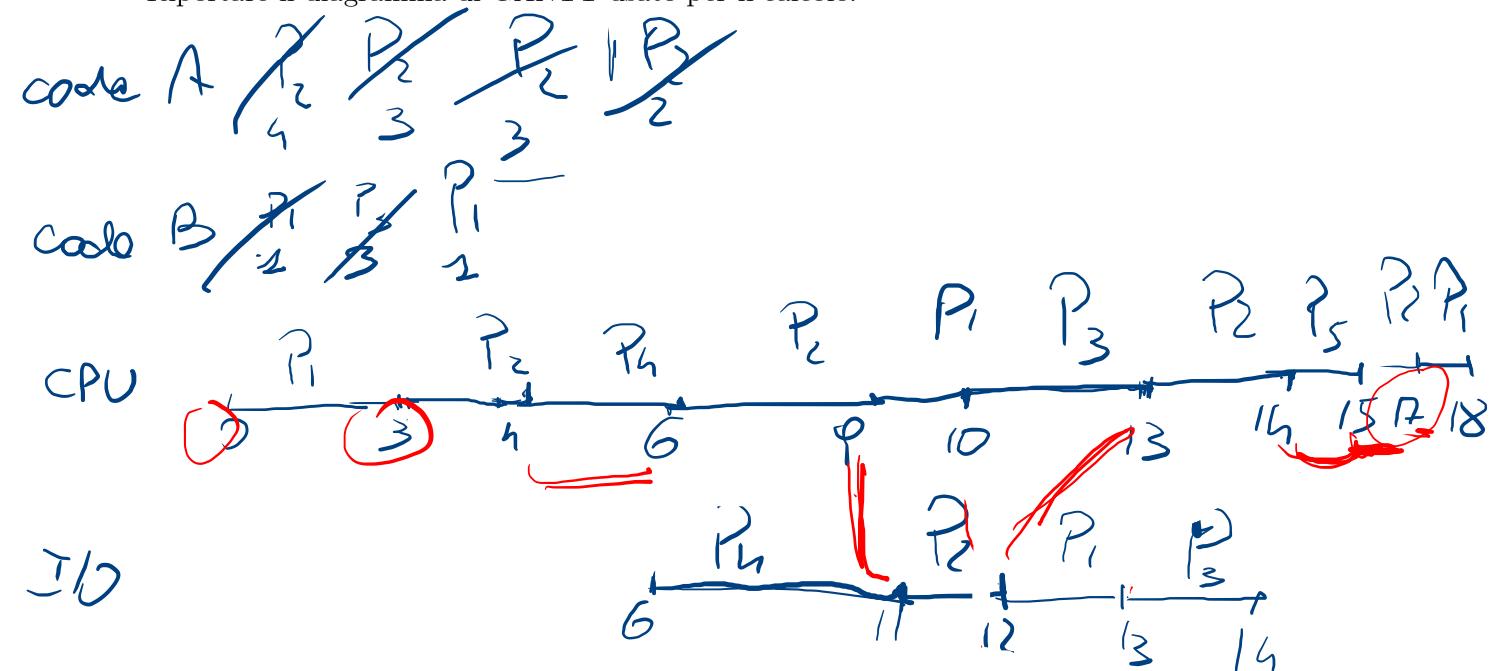
Si tenga conto, inoltre, che le operazioni di I/O avvengono tutte su "uno stesso dispositivo" il cui scheduling è gestito attraverso un algoritmo FIFO.

Nella tabella sottostante sono elencati i processi che arrivano nel sistema. Per ciascun processo sono riportati: il tempo di arrivo, la priorità (espressa con la lettera della coda corrispondente), l'entità dei burst che richiede.

Processo	T. di Arrivo	priorità	1° CPU burst	I/O burst	2° CPU burst
$P_1$	0	B	4	1	1
$P_2$	2	A	4	1	3
$P_3$	4	B	3	1	-
$P_4$	4	A	2	5	1
$P_5$	14	A	1	-	-

Calcolare il turnaround ed il waiting time di ogni processo.

Riportare il diagramma di GANTT usato per il calcolo.



$$WT \ P_1 \ 6 + 2 + 4 = 12$$

$$Turn \ P_1 \ 18$$

$$WT \ P_2 \ 1 + 2 + 2 + 1 + 1 = 7$$

$$Turn \ P_2 \ 15$$

## 3. (bonus) 10 punti

Si assuma che un processo  $P$  abbia già consumato 5 CPU burst la cui durata  $t_i$  per  $i = 1, \dots, 5$ , è mostrata nella tabella sottostante

$t_1$	$t_2$	$t_3$	$t_4$	$t_5$
1	5	3	7	2

- 1) Determinare la **stima del sesto CPU burst**  $\tau_6$  secondo un parametro  $\alpha$  con  $0 \leq \alpha \leq 1$ .
- 2) Discutere che il valore di  $\tau_6$  nel caso di  $\alpha = 0$  e  $\alpha = 1$ .

$$\begin{aligned}
 \tau_6 &= \alpha t_5 + (1-\alpha) \overbrace{\tau_5}^{\text{me}} \quad \text{me } \tau_5 = \alpha t_4 + (1-\alpha) \tau_4 \\
 &= \alpha t_5 + (1-\alpha)(\alpha t_4 + (1-\alpha) \tau_4) = \\
 &= \alpha t_5 + (1-\alpha)\alpha t_4 + (1-\alpha)^2 \tau_4 \quad \text{me } \tau_4 = \alpha t_3 + (1-\alpha) \tau_3 \\
 &= \alpha t_5 + (1-\alpha)\alpha t_4 + (1-\alpha)^2 (\alpha t_3 + (1-\alpha) \tau_3) = \\
 &= \alpha t_5 + (1-\alpha)\alpha t_4 + (1-\alpha)^2 \alpha t_3 + (1-\alpha)^3 \tau_3 \quad \text{me } \tau_3 = \alpha t_2 + (1-\alpha) \tau_2 \\
 &= \alpha t_5 + (1-\alpha)\alpha t_4 + (1-\alpha)^2 \alpha t_3 + (1-\alpha)^3 (\alpha t_2 + (1-\alpha) \tau_2) = \\
 &= \alpha t_5 + (1-\alpha)\alpha t_4 + (1-\alpha)^2 \alpha t_3 + (1-\alpha)^3 \alpha t_2 + (1-\alpha)^4 \tau_2 \\
 &= \alpha t_5 + (1-\alpha)\alpha t_4 + (1-\alpha)^2 \alpha t_3 + (1-\alpha)^3 \alpha t_2 + (1-\alpha)^4 (\alpha t_1 + (1-\alpha) \tau_1) \\
 &= \alpha t_5 + (1-\alpha)\alpha t_4 + (1-\alpha)^2 \alpha t_3 + (1-\alpha)^3 \alpha t_2 + (1-\alpha)^4 \alpha t_1 + (1-\alpha)^5 \tau_1 \\
 &\uparrow \quad \underline{\alpha = 0} \quad \tau_6 = \tau_1 \\
 &\downarrow \quad \alpha = 1 \quad \tau_6 = t_5
 \end{aligned}$$

## 4. (bonus) 10 punti

Si consideri un disco dotato di una sola testina e 100 tracce. Si consideri inoltre che lo spostamento da una traccia alla adiacente richieda  $1ms$  e che *per effettuare il rewind la testina impiega  $1ms$* . Si supponga che al tempo  $0ms$  mentre la testina si trova sulla traccia 18 e si sta muovendo verso la traccia 0, le richieste in sospeso siano (i tempi indicati sono in  $ms$ ):

traccia	25	6	10	66	51	97
tempo di arrivo	0	4	12	26	70	67

- a) Determinare come vengono servite le richieste seguendo la strategia **C-SCAN**
- b) Valutare, i tempi di attesa di ogni richiesta. [Si ricordi che il tempo di attesa di una richiesta è dato dal tempo intercorso tra l'arrivo della richiesta e il servizio della stessa.]

FOGLIO DA UTILIZZARE PER LA BRUTTA

FOGLIO DA UTILIZZARE PER LA BRUTTA