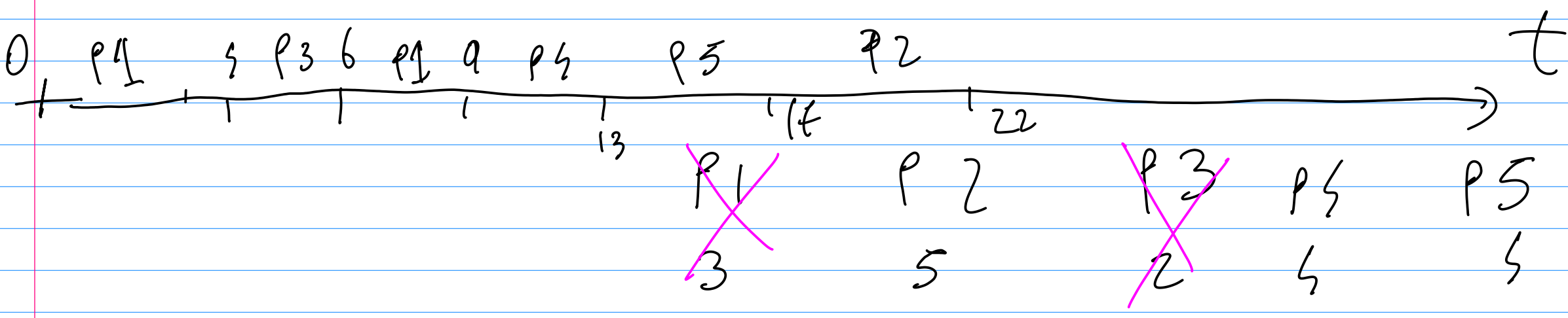


2. Supponiamo di avere cinque processi: P1, P2, P3, P4 e P5 con rispettive durate: 7, 5, 2, 4, 4 secondi. Supponiamo anche che i tempi di arrivo nella coda di scheduling dei vari processi sia, rispettivamente: 0, 3, 4, 6 e 9 secondi (ovvero, ad esempio, P4 arriva a tempo 6). Assumendo di utilizzare l'algoritmo di scheduling Shortest Remaining Time Next, quale è il processo che sarà schedulato per ultimo? Riportare solo il nome del processo.

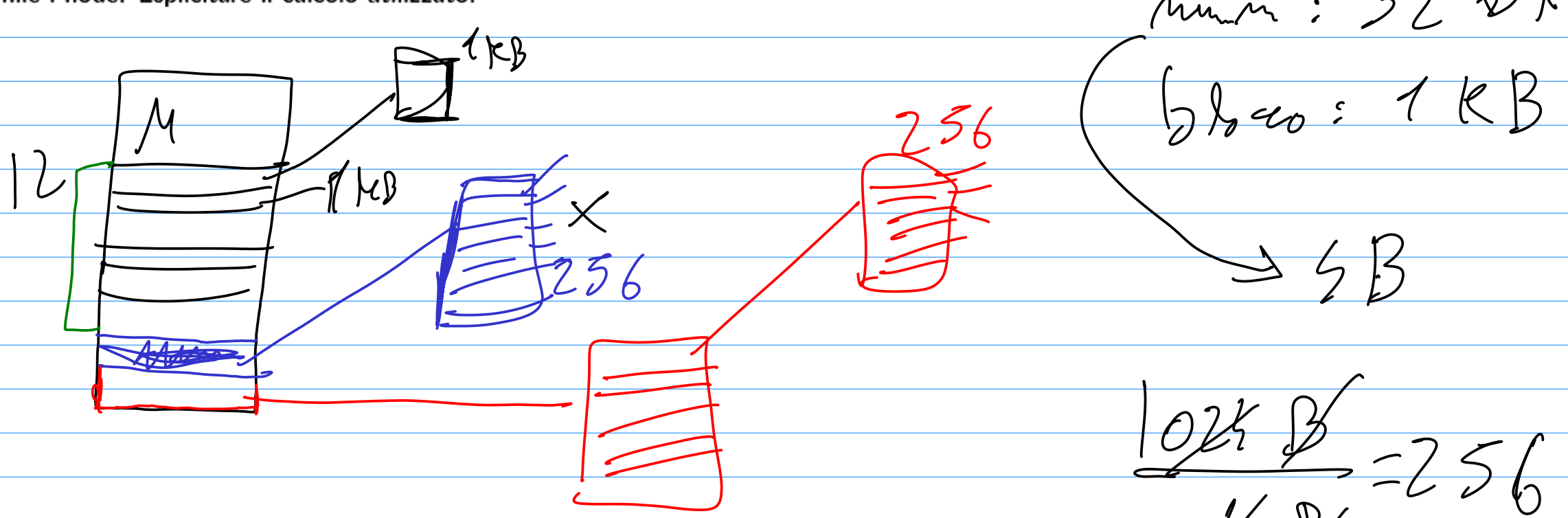
P1	7	0
P2	5	3
P3	2	4
P4	4	6
P5	4	9

Shortest remaining time next (greediziana)

SJF nel tempo rimanente



7. Supponiamo di utilizzare un file-system UNIX basato su i-node particolari che contengono i seguenti campi: 12 indirizzi diretti a blocchi di dati, 1 indirizzo ad un blocco indiretto singolo e 1 indirizzo ad un blocco indiretto doppio. Supponendo di avere numeri di blocchi a 32 bit e blocchi su disco da 1 kb, indicare esattamente la dimensione massima (in kb) supportata da un simile i-node. Esplicitare il calcolo utilizzato.



$$\frac{1024 \text{ B}}{4 \text{ B}} = 256$$

$$(12 + x + x^2) \cdot 1 \text{ KB}$$

$$(12 + 256 + 256^2) \cdot 1 \text{ KB} =$$

$$= (12 + 2^8 + (2^8)^2) \cdot 1 \text{ KB} =$$

$$= (12 + 2^8 + 2^{16}) \cdot 1 \text{ KB} = 65804 \text{ KB}$$

9. Tra gli algoritmi di sostituzione delle pagine abbiamo visto l'algoritmo di Aging. Tale algoritmo mantiene per ogni pagina un contatore binario che viene aggiornato periodicamente tenendo conto del bit di riferimento delle varie pagine. Supponiamo di avere 5 pagine in memoria e che a tempo t lo stato dei contatori binari sia il seguente:

pagina	contatore
A	00101000
B	10000001
C	01100000
D	00101011
E	00000010

In seguito, a tempo $t+1$, tali contatori sono aggiornati tenendo conto dello stato dei seguenti bit di riferimento (A:1, B:0, C:1, D:0, E:1) ed ancora, a tempo $t+2$, vengono nuovamente aggiornati con i bit di riferimento (A:0, B:1, C:0, D:0, E:1). Riportare nella risposta lo stato dei contatori dopo aver applicato questi due aggiornamenti ed indicare quale sarebbe la pagina destinata ad essere rimossa secondo l'algoritmo Aging se fosse necessario a tempo $t+3$.

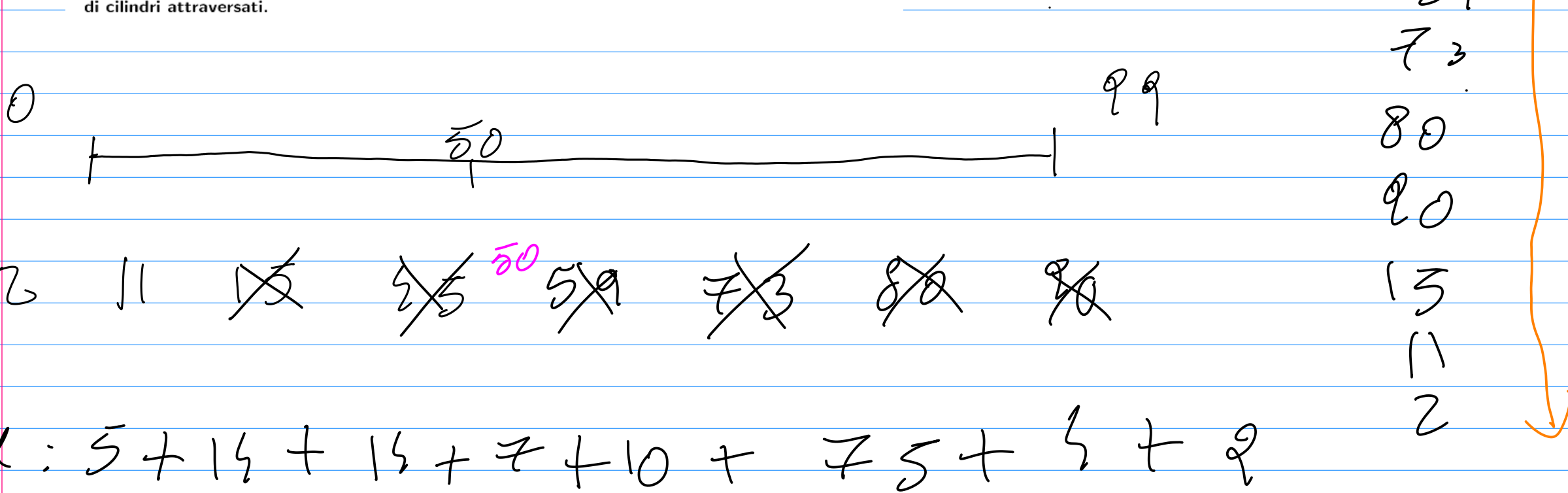
$t+1$: A 1 0010100 D 0 0010101
 B 0 1000000 E 1 0000001
 C 1 0110000

$t+2$: A 0 1001010 D 0 0001010
 B 1 0100000 E 1 1000000
 C 0 1011000

$t+3$: sostituire pagina D! D (count + bias)

5. Abbiamo visto vari algoritmi di scheduling dei movimenti della testina di un disco rotazionale che permettono di ottimizzare vari parametri. In particolare ci siamo soffermati sul "seek time". Supponiamo di avere una determinata sequenza di richieste di I/O e che i numeri di cilindro associati a tale richieste siano, in ordine di arrivo, i seguenti: 73, 15, 11, 59, 80, 45, 90, 2. Assumiamo che: la testina, all'arrivo della prima richiesta (ovvero quella per il cilindro 73), sia già posizionata sul cilindro 50; che i cilindri siano in totale 100 e che siano numerati da 0 a 99. Supponendo di utilizzare l'algoritmo di scheduling "shortest seek time first", indicare la sequenza esatta che si andrebbe ad utilizzare e la distanza totale coperta dalla testina in termini di numero di cilindri attraversati.

SSTF



$$SK: 5 + 15 + 15 + 7 + 10 + 75 + 3 + 9 = 138$$

7. Supponiamo di utilizzare un algoritmo di scheduling preemptive come il Round-Robin: assumendo di avere un context switch effettuato in 5 ms e di usare quanti di tempo lunghi 50 ms, a quanto ammonta la percentuale di overhead per la gestione dell'interlacciamento dei processi? Riportare per esteso il calcolo della percentuale richiesta.

Context switch $\rightarrow 5 \text{ ms}$
 $q \rightarrow 50 \text{ ms}$
 Overhead %:

$$\frac{5 \text{ ms}}{(50 + 5) \text{ ms}} = 0,09$$

 $100\% \rightarrow 9\%$

9. Supponiamo di avere 3 processi che condividono una variabile x e che i loro pseudo-codici siano i seguenti:

P1: wait(S) $x=x-1$ signal(T)	P2: wait(R) $x=x+2$ signal(T) wait(R) $x=x+1$ signal(T)	P3: wait(T) if ($x>0$) signal(R) else signal(S) wait(T) print(x)
--	---	--

Determinare l'output del processo P3 assumendo che il valore iniziale di x è -2 e che i 3 semafori abbiano i seguenti valori iniziali: S=0, R=1, T=0.

$$x = -2 \quad S = 0 \quad R = 1 \quad T = 0$$

P1 blocco(S)
 P2 $\rightarrow R=0$
 $x=0 \quad T=1$
 P2 blocco(R)
 P3 $\rightarrow T=0$
 $S=1$
 P3 blocco(T)
 P1 $S=0$
 $x=-1$
 $T=1$
 P3 zblocco
 print(x) $\rightarrow -1$

mutex_unlock:
 MOVE mutex, #0
 RET

mutex_lock:
 TSL reg, mutex
 CMP reg, 0
 JEQ OK
 CALL thread_yield
 JMP mutex_lock
 OK: RET

4. Consideriamo un sistema che fa uso di memoria virtuale con le seguenti caratteristiche: uno spazio di indirizzamento virtuale da 1 Gb, un numero di pagina virtuale a 22 bit e un indirizzo fisico a 20 bit. Determinare esattamente quanti frame fisici ci sono in memoria.

$$VM = 1 \text{ GB} \rightarrow 2^{30} \quad m = 20$$

$$\#PV: 22 \text{ bit}$$

$$\#i.p. 20 \text{ bit}$$

$$i.v. = \boxed{\#PV \mid offset}$$

$$\#frame = 2^{12}$$

$$\begin{aligned} \#ILO &: 1025 \quad 2^{10} \\ \#EVA &: 2^{20} \\ \#IGA &: 2^{30} \\ \#GR &: 2^{16} \end{aligned}$$

$$i.p. = \boxed{\#frame \mid off}$$

