

SCHEDULING

a)

Tempo	Evento	Processo A	Processo B	Processo C	Processo D	Sem
T	Scade QdT	Stato: Bloccato Priorità: 6	Stato: In exec. Priorità: 8	Stato: Pronto Priorità: 2 tempo in pronto: 70 ms	Stato: Pronto Priorità: 7 tempo in pronto: 0ms	0, A
T=10	Scade QdT	Stato: Bloccato Priorità: 6	Stato: Pronto Priorità: 7 tempo in pronto: 0ms	Stato: Pronto Priorità: 2 tempo in pronto: 80 ms	Stato: In exec. Priorità: 7	0, A
T=20	Scade QdT	Stato: Bloccato Priorità: 6	Stato: In exec. Priorità: 7	Stato: Pronto Priorità: 2 tempo in pronto: 90 ms	Stato: pronto Priorità: 6 tempo in pronto: 0ms	0, A
T=25	B esegue wait(Sem)	Stato: Bloccato Priorità: 6	Stato: Bloccato Priorità: 7	Stato: Pronto Priorità: 2 tempo in pronto: 95 ms	Stato: In Esec. Priorità: 6 tempo in pronto: 0ms	0, A,B
T=28	D Termina	Stato: Bloccato Priorità: 6	Stato: Bloccato Priorità: 7	Stato: In Esec Priorità: 2	-	0, A,B
T=30	C esegue signal(Sem)	Stato: In Esec Priorità: 7	Stato: Bloccato Priorità: 7	Stato: pronto Priorità: 2 tempo in pronto: 0 ms	-	0, B
T=40	Scade QdT	Stato: In Esec Priorità: 6	Stato: Bloccato Priorità: 7	Stato: pronto Priorità: 2 tempo in pronto: 10 ms	-	0, B

b)

Tempo	Evento	Processo A	Processo B	Processo C	Processo D	Sem1
T	Scade QdT	Stato: Pronto Priorità: 6 tempo in pronto: 60 ms	Stato: In exec. Priorità: 8	Stato: Pronto Priorità: 2 tempo in pronto: 84 ms	Stato: Pronto Priorità: 7 tempo in pronto: 0 ms	1,-
T+10	Scade QdT	Stato: Pronto Priorità: 6 tempo in pronto: 70 ms	Stato: Pronto. Priorità: 7 tempo in pronto: 0 ms	Stato: Pronto Priorità: 2 tempo in pronto: 94 ms	Stato: In exec. Priorità: 7	1,-
T+16	C passa in esecuzione	Stato: Pronto Priorità: 6 tempo in pronto: 76 ms	Stato: Pronto. Priorità: 7 tempo in pronto: 6 ms	Stato: In exec Priorità: 9	Stato: Pronto Priorità: 7 tempo in pronto: 0 ms	1,-
T+25	C esegue wait(Sem)	Stato: Pronto Priorità: 6 tempo in pronto: 85 ms	Stato: Pronto. Priorità: 7 tempo in pronto: 15 ms	Stato: In exec Priorità: 9	Stato: Pronto Priorità: 7 tempo in pronto: 9 ms	0,-
T+26	Scade QdT	Stato: Pronto Priorità: 6 tempo in pronto: 86 ms	Stato: In exec. Priorità: 7	Stato: Pronto. Priorità: 5 tempo in pronto: 0 ms	Stato: Pronto Priorità: 7 tempo in pronto: 10 ms	0,-
T+28	B termina	Stato: Pronto Priorità: 6 tempo in pronto: 88 ms	-	Stato: Pronto. Priorità: 5 tempo in pronto: 2 ms	Stato: In exec. Priorità: 7	0,-
T+30	D esegue signal(Sem)	Stato: Pronto Priorità: 6 tempo in pronto: 90 ms	-	Stato: Pronto. Priorità: 5 tempo in pronto: 4 ms	Stato: In exec. Priorità: 7	1,-
T+38	Scade QdT	Stato: In exec. Priorità: 6	-	Stato: Pronto. Priorità: 5 tempo in pronto: 6 ms	Stato: Pronto Priorità: 6 tempo in pronto: 0 ms	1,-
T+40	-	Stato: In exec. Priorità: 6	-	Stato: Pronto. Priorità: 5 tempo in pronto:	Stato: Pronto Priorità: 6 tempo in pronto:	1,-

TABELLA DELLE PAGINE

1. lunghezza del campo offset : 10 bit
2. per la codifica degli indici di blocco sono disponibili 14 bit, di cui 6 sono indici della tabella di primo livello e i rimanenti 8 sono indici delle tabelle di secondo livello. Pertanto ogni tabella di secondo livello contiene 2^8 elementi
3. spazio occupato in memoria da ogni tabella di secondo livello : $2^8 * 3 = 768$ byte
4. gli indici dei blocchi di memoria fisica sono codificati con 12 bit, quindi la massima dimensione della memoria fisica è di 2^{12} blocchi $\rightarrow 2^{12} * 2^{10} = 2^{22}$ byte $\rightarrow 4$ MByte

Le tabelle di secondo livello da caricare in memoria sono le seguenti:

- a) per il riferimento alla pagina $8000 = 31 * 2^8 + 64$ si deve caricare la tabella di secondo livello di indice 31
- b) per il riferimento alla pagina $10000 = 39 * 2^8 + 16$ si deve caricare la tabella di secondo livello di indice 39
- c) per il riferimento alla pagina $600 = 2 * 2^8 + 88$ si deve caricare la tabella di secondo livello di indice 2.

WORKING SET

1) Configurazione della *CoreMap* e delle tabelle delle pagine al termine del punto 1:

B,0		B,1	C,4	C,8	C,2	A,4	B,2	A,2		B,5	A,1	A,6		B,6	B,7	A,5	C,7	B,9	
10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29

Pagina	Blocco	TempoRif
0		
1	21	4
2	18	8
3		
4	16	10
5	26	1
6	22	5
7		
8		
9		
Processo A		

Pagina	Blocco	TempoRif
0	10	13
1	12	12
2	17	11
3		
4		
5	20	9
6	24	3
7	25	6
8		
9	28	14
Processo B		

Pagina	Blocco	TempoRif
0		
1		
2	15	8
3		
4	13	5
5		
6		
7	27	4
8	14	9
9		
Processo C		

Non è stato eseguito il *Working Set Manager*

2) Configurazione della *CoreMap* e delle tabelle delle pagine al termine del punto 2:

B,0	A,9	B,1	C,4	C,8	C,2	A,4	B,2	A,2	A,8	B,5	A,1	A,6		B,6	B,7	A,5	C,7	B,9	
10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29

Pagina	Blocco	TempoRif
0		
1	21	4
2	18	8
3		
4	16	14
5	26	12
6	22	13
7		
8	19	15
9	11	11
Processo A		

Pagina	Blocco	TempoRif
0	10	13
1	12	12
2	17	11
3		
4		
5	20	9
6	24	3
7	25	6
8		
9	28	14
Processo B		

Pagina	Blocco	TempoRif
0		
1		
2	15	8
3		
4	13	5
5		
6		
7	27	4
8	14	9
9		
Processo C		

Viene quindi eseguito il *Working Set Manager* che rimuove 6 pagine dei soli processi A e B.

Il tempo attuale del processo A è: 15

Il tempo attuale del processo B è: 14

Vengono quindi rimosse le pagine: A1, B6, B7, A2, B5, A9

B,0		B,1	C,4	C,8	C,2	A,4	B,2		A,8			A,6				A,5	C,7	B,9	
10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29

Pagina	Blocco	TempoRif
0		
1		
2		
3		
4	16	14
5	26	12
6	22	13
7		
8	19	15
9		
Processo A		

Pagina	Blocco	TempoRif
0	10	13
1	12	12
2	17	11
3		
4		
5		
6		
7		
8		
9	28	14
Processo B		

Pagina	Blocco	TempoRif
0		
1		
2	15	8
3		
4	13	5
5		
6		
7	27	4
8	14	9
9		
Processo C		

3) Configurazione della *CoreMap* e delle tabelle delle pagine al termine del punto 3:

B.0	B.7	B.1	C.4	C.8	C.2	A.4	B.2	B.5	A.8	B.8		A.6				A.5	C.7	B.9	
10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29

Pagina	Blocco	TempoRif
0		
1		
2		
3		
4	16	14
5	26	12
6	22	13
7		
8	19	15
9		
Processo A		

Pagina	Blocco	TempoRif
0	10	13
1	12	12
2	17	11
3		
4		
5	18	16
6		
7	11	15
8	20	17
9	28	14
Processo B		

Pagina	Blocco	TempoRif
0		
1		
2	15	8
3		
4	13	5
5		
6		
7	27	4
8	14	9
9		
Processo C		

Non è stato eseguito il *Working Set Manager*

PAGE DAEMON

↓

Proc	A			C			B	C			B	D	B	B	C	B	B	A	D	A	D	
Pag	11			9			1	3			0	6	2	6	7	3	7	9	2	1	7	
Rif	0			0			0	0			0	0	0	1	0	1	0	0	0	0	0	
Blocco	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21

Core Map al tempo t+1

Scaricate le pagine: D8, C0, A8, D1

↓

Proc	A	A	A	C	B	C	B	C	C		B	D	B	B	C	B	B	A	D	A	D	
Pag	11	8	5	9	9	0	1	3	2		0	6	2	6	7	3	7	9	2	1	7	
Rif	0	1	1	0	1	1	0	1	1		0	0	0	1	0	1	0	1	0	1	0	
Blocco	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21

Core Map al tempo t+10

Caricate le pagine: A8, A5, B9, C0, C2

↓

Proc				C	B	C	B	C	C		B	D	B	B	C	B	B		D		D	
Pag				9	9	0	1	3	2		0	6	2	6	7	3	7		2		7	
Rif				0	1	1	0	1	1		0	0	0	1	0	1	0		0		0	
Blocco	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21

Core Map al tempo t+11

Swap out del processo A

↓

Proc	B	B	C	C	B	C	B	C	C		B	D	B	B	C	B	B		D		D	
Pag	4	5	8	9	9	0	1	3	2		0	6	2	6	7	3	7		2		7	
Rif	1	1	1	1	1	1	0	1	1		1	1	0	1	1	1	1		1		1	
Blocco	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21

Core Map al tempo t+20

Caricate le pagine: B4, B5, C8

↓

Proc	B	B	C	C	B	C			C		B	D		B	C	B	B		D		D	
Pag	4	5	8	9	9	0			2		0	6		6	7	3	7		2		7	
Rif	0	0	0	0	0	0			0		0	0		0	0	0	0		0		0	
Blocco	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21

Core Map al tempo t+21

Scaricate le pagine: B2, B1, C3

GESTIONE DISCO

op. su cilindro:	670	settore:	55				
inizio:	0	seek:	3,3	rotazione:	20	percorrenza:	0,025 fine: 23,325
op. su cilindro:	32	settore:	100				
inizio:	23,325	seek:	6,38	rotazione:	20	percorrenza:	0,025 fine: 49,73
op. su cilindro:	300	settore:	61				
inizio:	49,73	seek:	2,68	rotazione:	20	percorrenza:	0,025 fine: 72,435
op. su cilindro:	400	settore:	88				
inizio:	72,435	seek:	1	rotazione:	20	percorrenza:	0,025 fine: 93,46
op. su cilindro:	400	settore:	199				
inizio:	93,46	seek:	0	rotazione:	2,75	percorrenza:	0,025 fine: 96,235
op. su cilindro:	1300	settore:	750				
inizio:	96,235	seek:	9	rotazione:	20	percorrenza:	0,025 fine: 125,26
op. su cilindro:	1200	settore:	662				
inizio:	125,26	seek:	1	rotazione:	20	percorrenza:	0,025 fine: 146,285
op. su cilindro:	320	settore:	50				
inizio:	146,285	seek:	8,8	rotazione:	20	percorrenza:	0,025 fine: 175,11

FAT

1. Capacità del disco: $512 * 2 * 1024 = 2^{20} = 1\text{M}$ blocchi $\rightarrow 2^{30} = 1\text{ Gbyte}$
2. La lunghezza degli elementi della FAT, necessaria per indirizzare tutti i blocchi del disco, è di 32 bit $\rightarrow 4$ byte;
3. La FAT occupa $2^{20} * 4$ byte $\rightarrow 2^{22} / 2^{10}$ blocchi $\rightarrow 4096$ blocchi;
ciascun blocco contiene 256 elementi della FAT
4. Il primo blocco del disco disponibile come blocco dati ha indice 4098
5. Il numero di blocchi dati nel File System è di $2^{20} - 4098$ blocchi
6. Il primo *run* è distribuito nei blocchi 15600 ... 15612; il secondo *run* è distribuito nei blocchi 9900 ... 9919.
Il carattere 12.000 (primo carattere da leggere) occupa la posizione $12.000 \bmod 1024 = 736$ nel blocco logico $12.000 \div 1024 = 11$, mappato nel blocco dati $15600 + 11 = 15611$, che è il penultimo del primo run.
Nel primo run quindi si leggono $1024 - 736 = 288$ byte dal blocco 15611 e 1024 byte dal blocco 15612.
Nel secondo run si leggono $3000 - 1024 - 288 = 1688$ bytes. Quindi si leggono 1024 bytes dal blocco 9900 e $1688 - 1024 = 664$ bytes dal blocco 9901.
7. Ricordando che ogni blocco del disco contiene 256 elementi della FAT e che l'elemento i della FAT è associato al blocco dati i del file, contiene il puntatore al blocco dati $i+1$ ed è contenuto nel blocco $i \div 256$, gli elementi della FAT ai quali si deve accedere per eseguire la lettura sono quelli di indice:
 - 15600-15612 che sono contenuti nel blocco 60 della FAT, che è il numero 62 del disco. L'elemento 15612 contiene il puntatore al blocco dati 9900, che è il primo blocco del secondo run.
 - 9900, contenuto nel blocco 38 della FAT che è il numero 40 del disco. Contiene il puntatore al blocco dati 9901.In conclusione si devono leggere dal disco i blocchi 62 e 38.
8. Il massimo numero di errori di pagina è 2, per la lettura dei blocchi 62 e 38 nell'ipotesi che non siano già caricati in memoria.

FFS

1. Un indirizzo occupa 16 bit=2 bytes. Pertanto, il numero di puntatori che possono essere contenuti in un blocco indiretto è dato da $2^{10}/2= 512$;
2. il primo e l'ultimo blocco indirizzabili con puntatori diretti hanno rispettivamente indici logici 0 e 9;
3. il primo e l'ultimo blocco indirizzabili con indirizzamento indiretto semplice hanno rispettivamente indici logici 10 e $(10+ 512) - 1= 521$;
4. l'ultimo carattere del file è contenuto nel blocco $(498.600 - 1) \text{ div } 2^{10}= 486$; quindi il file è composto da 487 blocchi;

il primo carattere da leggere è il 13.400, sul quale è posizionato il puntatore di lettura, è contenuto nel blocco di indice logico: $13.400 \text{ div } 2^{10}= 13$. L'ultimo carattere da leggere è il numero $13.400+2.000-1 = 15.399$, che è contenuto nel blocco di indice logico: $15.399 \text{ div } 2^{10}= 15$. Quindi i blocchi logici da leggere sono il 13, 14 e 15, che rientrano nell'indirizzamento indiretto singolo. Nella sequenza dei blocchi indirizzati dall'indirizzamento indiretto singolo, il puntatore corrispondente al blocco logico 13 occupa la posizione $13-10=3$ del blocco fisico 800.

Quindi il file system deve leggere il blocco indice di primo livello 800. All'interno di questo blocco individua gli elementi 3, 4 e 5 che contengono i puntatori ai blocchi 677, 678 e 705 che devono essere letti.

I blocchi da leggere sono quindi **800, 677, 678, 705**

NTFS

numero run	<i>(inizio, lunghezza)</i>
0	600,6
1	21000,2
2	21003,1
3	1890,3

numero di blocchi da leggere: **1**

indirizzo del primo blocco del run: **21003**

cilindro del primo blocco del run: **52**

tempo di seek: $(88-52) * 1 \text{ msec} = 36 \text{ msec}$

tempo rotazionale: **10 msec**

tempo di lettura dei blocchi: **0,1 msec**

tempo totale per leggere tutto il 3° run: **46,1 msec**