

---

# **Esercitazione E6**

## **Compito di autovalutazione**

# Memoria virtuale

---

Si consideri un sistema a memoria virtuale con indirizzi a 32 bit, indirizzi fisici a 20 bit, pagine da 512 byte, e descrittori di pagina di 16 byte, calcolare:

- a) la struttura dell'indirizzo virtuale e dell'indirizzo fisico;
- b) la dimensione in pagine ed in byte degli spazi di indirizzamento virtuale e fisico;
- c) la struttura e le dimensioni (complessive e delle singole tavole) in un'organizzazione con tavola delle pagine a due livelli in cui i 12 bit più significativi dell'indirizzo corrispondono al livello più alto;
- d) quante pagine occupa un programma di 2100 byte.

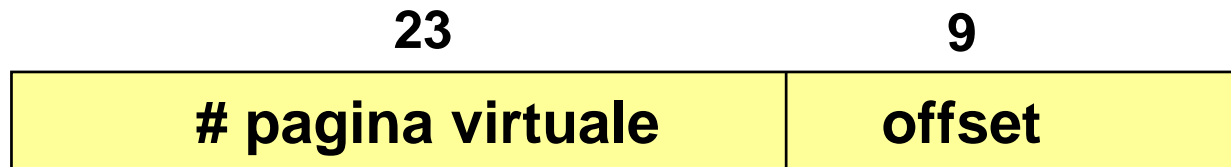
# Memoria virtuale

---

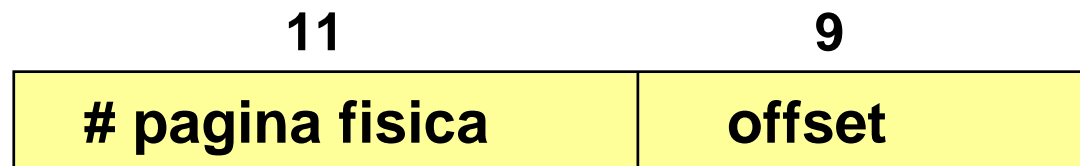
a)

## a) La struttura dell'indirizzo virtuale e dell'indirizzo fisico

- L'offset dipende solo dalla dimensione delle pagine
- Dato che le pagine sono da 512 byte =  $2^9$  byte, l'offset è di 9 bit
- Struttura dell'indirizzo virtuale:



- Struttura dell'indirizzo fisico



# Memoria virtuale

---

b)

b) la dimensione in pagine ed in byte degli spazi di indirizzamento virtuale e fisico

- Essendo gli indirizzi virtuali a 32 bit o spazio virtuale è costituito da  $2^{32}$  byte
- Le pagine sono da 512 byte =  $2^9$  byte, pertanto il numero di pagine virtuali è di:

$$2^{32}/2^9 = 2^{23} = 8 \text{ M pagine}$$

- Essendo gli indirizzi fisici a 20 bit o spazio fisico è costituito da  $2^{20}$  byte
- Le pagine sono da 512 byte =  $2^9$  byte, pertanto il numero di pagine fisico è di:

$$2^{20}/2^9 = 2^{11} = 2 \text{ K pagine}$$

- c) la struttura e le dimensioni (complessive e delle singole tavole) in un'organizzazione con tavola delle pagine a due livelli in cui i 12 bit più significativi dell'indirizzo corrispondono al livello più alto;
- Una tavola delle pagine a due livelli è costituita da:
    - Una *tavola di primo livello* in cui viene usato come indice il primo gruppo di bit (**12** bit). Pertanto la tavola ha  **$2^{12}$**  elementi.
    - Una *tavola di secondo livello* in corrispondenza a ciascun elemento della tavola di primo livello (nell'esempio  **$2^{12}$**  tavole), in ciascuna delle quali viene usato come indice il secondo gruppo di bit (**11** bit).
  - Ciascun elemento della tavola di primo livello è un puntatore alla corrispondente tavola di secondo livello, ed ha le dimensioni di un indirizzo virtuale; nell'esempio **32** bit cioè **4** byte =  **$2^2$**  byte.
  - Ciascun elemento della tavola di secondo livello è un descrittore, nell'esempio **16** byte =  **$2^4$**  byte.

c) la struttura e le dimensioni (complessive e delle singole tavole) in un'organizzazione con tavola delle pagine a due livelli in cui i 12 bit più significativi dell'indirizzo corrispondono al livello più alto;

— La *tavola di primo livello* ha  $2^{12}$  elementi di  $2^2$  byte ciascuno, quindi occupa complessivamente:

$$2^{12} \cdot 2^2 \text{ byte} = 2^{14} \text{ byte} = 16 \text{ Kbyte}$$

— Ciascuna delle *tavole di secondo livello* ha  $2^{11}$  elementi di  $2^4$  byte ciascuno, quindi occupa complessivamente:

$$2^{11} \cdot 2^4 \text{ byte} = 2^{15} \text{ byte} = 32 \text{ Kbyte}$$

— Dato che le tavole di secondo livello sono  $2^{12}$ , esse occupano complessivamente:

$$2^{12} \cdot 2^{15} \text{ byte} = 2^{27} \text{ byte} = 128 \text{ Mbyte}$$

# Memoria virtuale

---

d)

d) quante pagine occupa una programma di 2100 byte.

- La procedura viene allocata in pagine da **512** byte
- Il numero di pagine occupate dalla procedura è dato da:

$$P = \lceil 2100 / 512 \rceil = 5$$

- L'ultima pagina è solo parzialmente occupata. C'è uno sfrido pari a:

$$S = 5 \cdot 512 - 2100 = 2560 - 2100 = 460 \text{ byte}$$

- Questo spreco corrisponde al fenomeno della *frammentazione interna*

# FAT 16 e FAT 32

---

Si consideri un file system FAT 32 su un disco da 1 GB con cluster da 2 KB. Il disco è pieno complessivamente al 10%, e la metà dello spazio è occupato da file di dimensione inferiore a 2 KB. Tutto il contenuto del disco viene copiato su un altro disco della stessa dimensione, ma con file system FAT 16 e dimensione dei cluster minima. Si calcoli:

- a) la dimensione dei cluster nel secondo disco;
- b) il numero dei file di dimensione inferiore a 2 KB;
- c) una stima dell'occupazione dei file di dimensione inferiore a 2k sul secondo disco;
- d) una stima dello spazio sprecato nella nuova organizzazione a causa dei file di dimensione inferiore a 2 KB.

# FAT 16 e FAT 32

---

a)

## a) Calcolare la dimensione dei cluster nel secondo disco

- La dimensione *minima* del cluster dipende dalla dimensione della partizione, e dal file system
- Nella FAT 16 ci possono essere al massimo  $2^{16}$  cluster
- Pertanto, dato che il disco è da 1 GB =  $2^{30}$  byte:

$$\text{dim. minima cluster} = 2^{30} / 2^{16} = 2^{14} = 16 \text{ KB}$$

- I cluster nella disco in questione sono pertanto da 16 KB

## b) Stimare il numero dei file di dimensione inferiore a 2 KB

- Dall'enunciato si sa che, nel primo disco:
  - è occupato il 10% del disco cioè circa **100 MB**
  - i file di dimensione inferiore a **2 KB** occupano la metà dello spazio, cioè circa **50 MB**.
- Un file, anche se più piccolo di **2 KB**, occupa comunque un intero cluster , cioè **2 KB**
- Quindi, una stima del il numero dei file 'piccoli' è:

$$N_{\text{small}} = 50 \text{ MB} / 2 \text{ KB} = 25.000$$

- Sugli altri file non è possibile dedurre altro, se non il fatto che occupano, nella prima organizzazione, circa **50 MB**

c) una stima dell'occupazione dei file di dimensione inferiore a 2k sul secondo disco;

- Ciascun file ‘piccolo’ occupa un cluster
- I cluster sul secondo disco sono da **16 KB**
- La occupazione complessiva dei file piccoli sul secondo disco è quindi:

$$N_{\text{small}} \cdot 16 \text{ KB} = 25.000 \cdot 16 \text{ KB} \approx 400 \text{ MB}$$

- Quindi lo spazio sprecato attribuibile ai file piccoli è:

$$S_{\text{FAT16}} - S_{\text{min}} = 400 \text{ MB} - 50 \text{ MB} \approx 350 \text{ MB}$$

- Per gli altri file non è possibile alcuna stima, essenzialmente perché non si sa né quanti sono né di che dimensione

# FAT 16 e FAT 32: e i file 'grandi'?

---

- Si può calcolare una stima per eccesso dello spazio occupato dai file 'grandi' nella nuova organizzazione?
- La stima conservativa consiste nel caso peggiore, cioè che siano tutti di dimensione uguale a **2 K**
- Quindi, considerando che sul primo disco occupano metà dello spazio occupato, saranno esattamente nello stesso numero dei file piccoli
- Pertanto sul secondo disco occuperanno anch'essi circa **400 MB**.
- Si può quindi affermare che, nel caso peggiore, il nuovo disco i file occuperanno complessivamente **800 MB**, cioè esso sarà pieno al **80%**
- Questa è la migliore stima per eccesso che possiamo fare.

# Gestione di processi e di thread

---

- V** - lo scheduler determina il passaggio di un thread dallo stato ready allo stato running;
- F** - thread diversi dello stesso processo condividono lo stesso stack;
- V** - nella soluzione basata sull'alternanza stretta un processo fuori della sezione critica può bloccare un'altro processo;
- V** - un processo che esegue  $\text{DOWN}(S)$  quando  $S=0$  entra in stato di blocco;
- F** - tutte le discipline di scheduling tendono a massimizzare l'utilizzazione della CPU;
- F** - tutte le discipline di scheduling tendono a minimizzare il tempo di risposta;
- V** - lo scheduling FCFS penalizza i processi I/O bound;

# Gestione della memoria virtuale

---

- V** - le istruzioni macchina fanno riferimento ad indirizzi virtuali;
- V** - la dimensione dello spazio di indirizzamento virtuale è una caratteristica della piattaforma hardware;
- V** - in un sistema a paginazione la traduzione degli indirizzi mantiene l'offset;
- V** - il TLB consente di ridurre gli accessi alla tavola delle pagine;
- F** - località spaziale significa forte probabilità di fare riferimento allo stesso indirizzo in tempi vicini;
- F** - l'algoritmo di rimpiazzamento ottimo si basa sulla conoscenza dettagliata dei riferimenti passati;
- F** - la segmentazione risolve il problema della frammentazione esterna.

# Sistemi multimedia

---

- F** - JPEG realizza una compressione lossless delle immagini;
- V** - MPEG sfrutta la ridondanza spaziale;
- F** - dopo la compressione MPEG tutti i fotogrammi hanno la stessa dimensione;
- F** - lo scheduling RMS della CPU da sempre risultati migliori rispetto allo EDF;
- F** - il supporto delle funzioni VCR non comporta il ricorso a stream privati;
- V** - la distribuzione di Zipf delle frequenze di accesso ai film dipende esclusivamente dal numero di film gestiti;
- F** - in MPEG per decomprimere un frame di tipo I occorre conoscere i frame precedenti.

# Gestione di dischi e file system

---

- F** - nel file system di Unix tutti gli utenti del gruppo dell'owner hanno gli stessi diritti dell'owner del file;
- V** - a parità di dimensione del blocco la dimensione massima di un file in un sistema Unix dipende da quella degli indirizzi su disco;
- F** - usando la FAT 16 si ha una migliore utilizzazione dello spazio su disco rispetto alla FAT 32;
- F** - in una partizione Unix gli i-node vengono creati ed allocati man mano che i file vengono creati;
- F** - la MFT è una struttura più compatta della FAT;
- F** - per deframmentare un disco occorre prima riformattarlo;
- V** - lo scheduling dei dischi sfrutta la conoscenza della coda di richieste di accesso.