

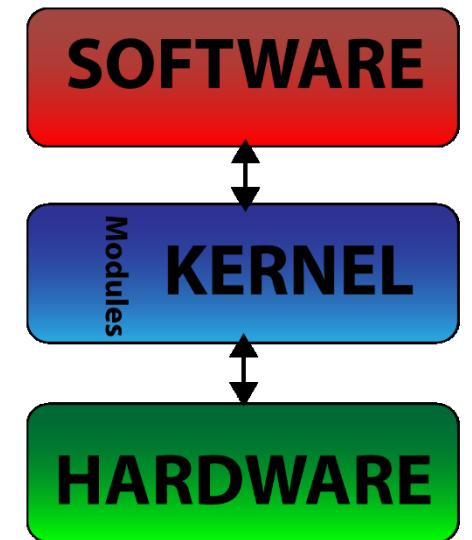
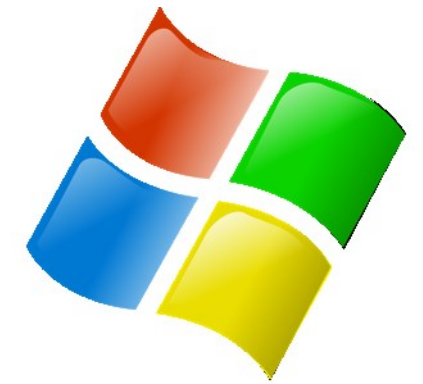


**UNIVERSITÀ DEGLI STUDI
DELLA BASILICATA**

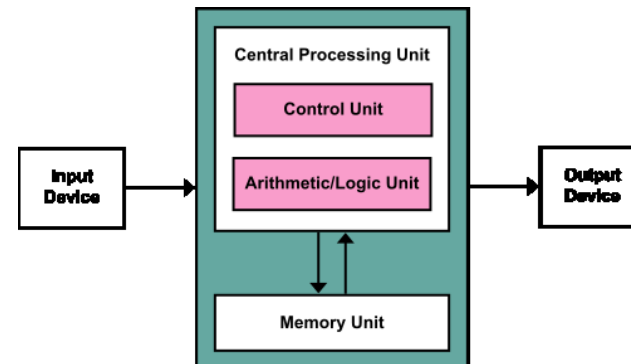
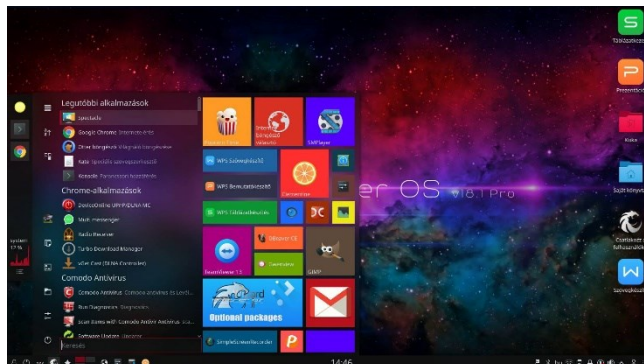
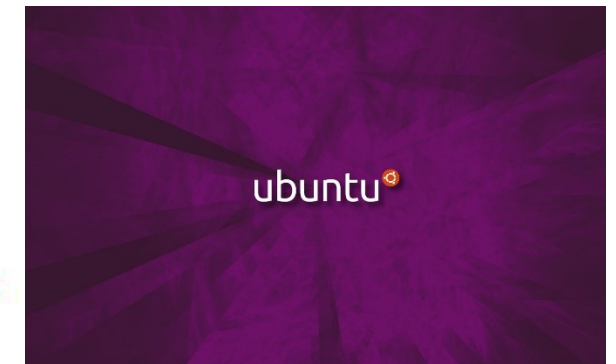
*Corso di Sistemi Operativi
A.A. 2019/20*

Esercitazione

Memoria centrale



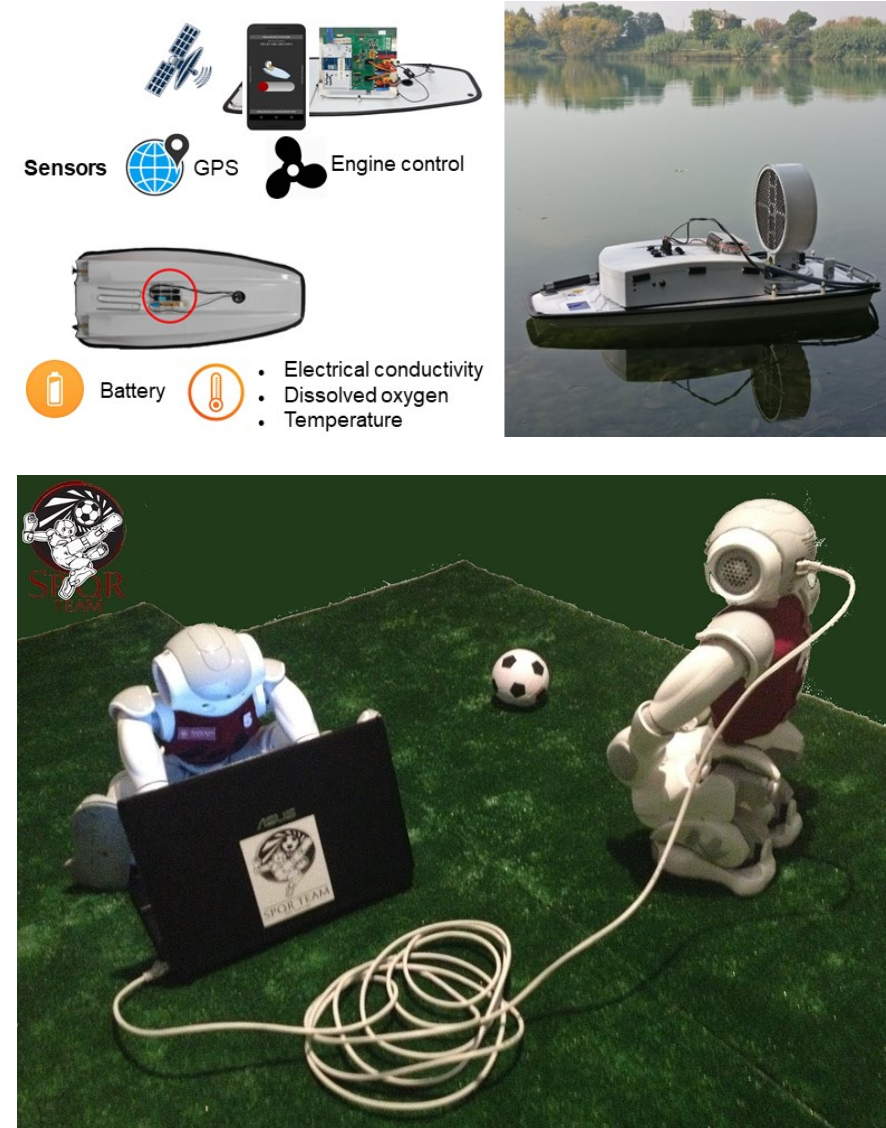
Docente:
**Domenico Daniele
Bloisi**



Novembre 2019

Domenico Daniele Bloisi

- Ricercatore RTD B
Dipartimento di Matematica, Informatica
ed Economia
Università degli studi della Basilicata
<http://web.unibas.it/bloisi>
- SPQR Robot Soccer Team
Dipartimento di Informatica, Automatica
e Gestionale Università degli studi di
Roma “La Sapienza”
<http://spqr.diag.uniroma1.it>



Ricevimento

- In aula, subito dopo le lezioni
- Martedì dalle 11:00 alle 13:00 presso:
Campus di Macchia Romana
[Edificio 3D](#) (Dipartimento di Matematica,
Informatica ed Economia)
[Il piano, stanza 15](#)

Email: domenico.bloisi@unibas.it



Domanda 1

Spiegare la differenza tra indirizzi logici e indirizzi fisici

Risposta Domanda 1

Gli indirizzi logici sono gli indirizzi generati dai programmi utenti relativi alla locazione 0 in memoria

Gli indirizzi fisici sono i reali indirizzi usati per recuperare e immagazzinare dati nella memoria del calcolatore

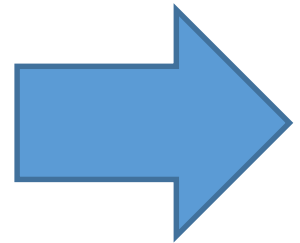
Domanda 2

Spiegare la differenza tra frammentazione interna e frammentazione esterna

Risposta Domanda 2

La frammentazione interna si verifica quando un processo p non occupa interamente la porzione di memoria che gli è stata allocata.

Tale spazio rimane inutilizzabile da altri processi fino al rilascio della memoria allocata a p



Risposta Domanda 2

La frammentazione esterna si verifica quando c'è una quantità totale di memoria libera sufficiente a soddisfare la richiesta di memoria da parte di un nuovo processo, tuttavia non sono disponibili porzioni di memoria contigue così grandi da contenere il nuovo processo.

Domanda 3

Fornire una descrizione dei seguenti algoritmi di allocazione:

- First-fit
- Best-fit
- Worst-fit

Risposta Domanda 3

First-fit: si scorre la lista delle locazioni di memoria disponibili fino a trovare il primo blocco che sia sufficientemente grande

Best-fit: si scandisce tutta la lista delle locazioni di memoria disponibili e si alloca il più piccolo blocco che sia sufficientemente grande

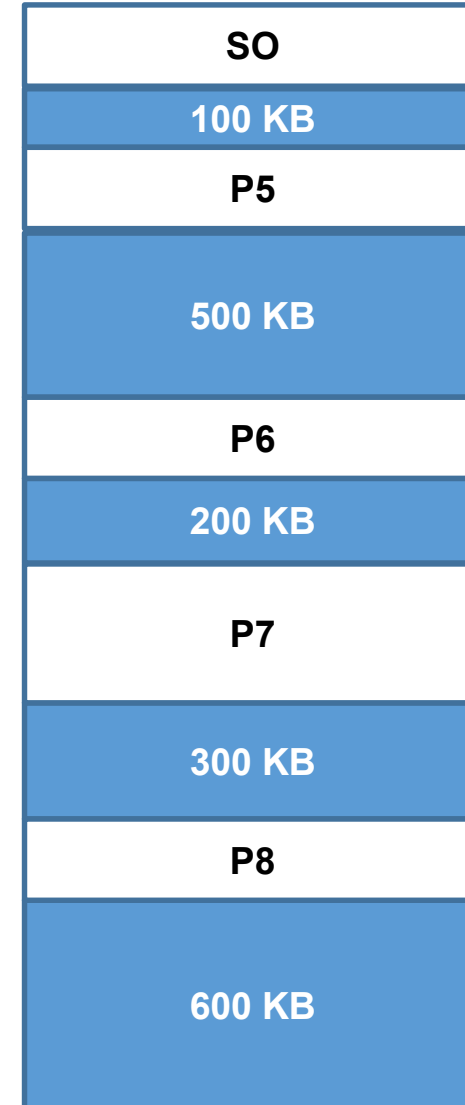
Worst-fit: si scandisce tutta la lista delle locazioni di memoria disponibili e si alloca il più piccolo blocco che sia sufficientemente grande. Questa politica produce le parti inutilizzate più grandi, le quali possono essere più utili rispetto alle piccole porzioni create da best-fit.

Esercizio 1

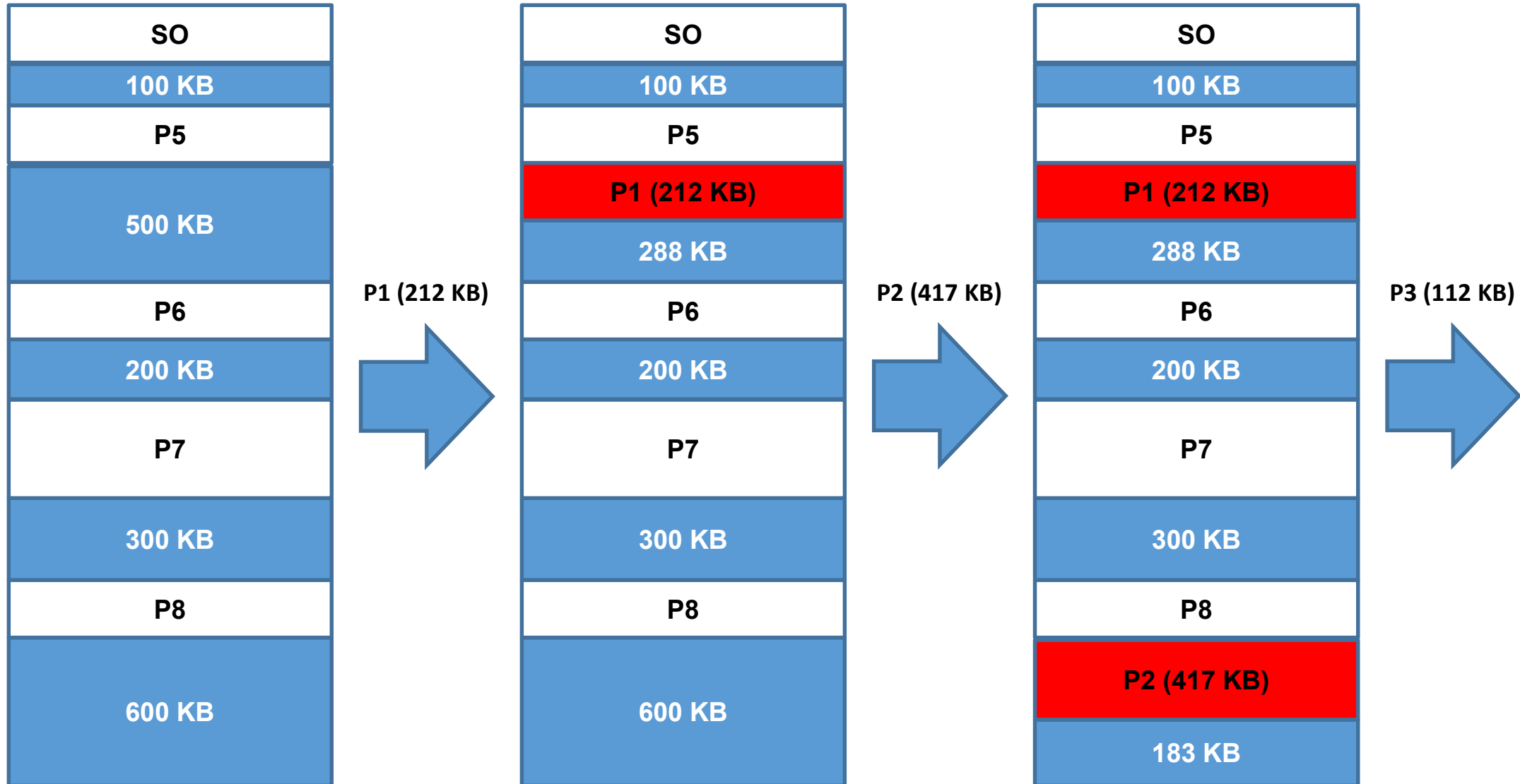
Si assuma di avere la memoria nella situazione illustrata a lato.

Utilizzando un approccio first-fit, come verranno allocati i processi seguenti?

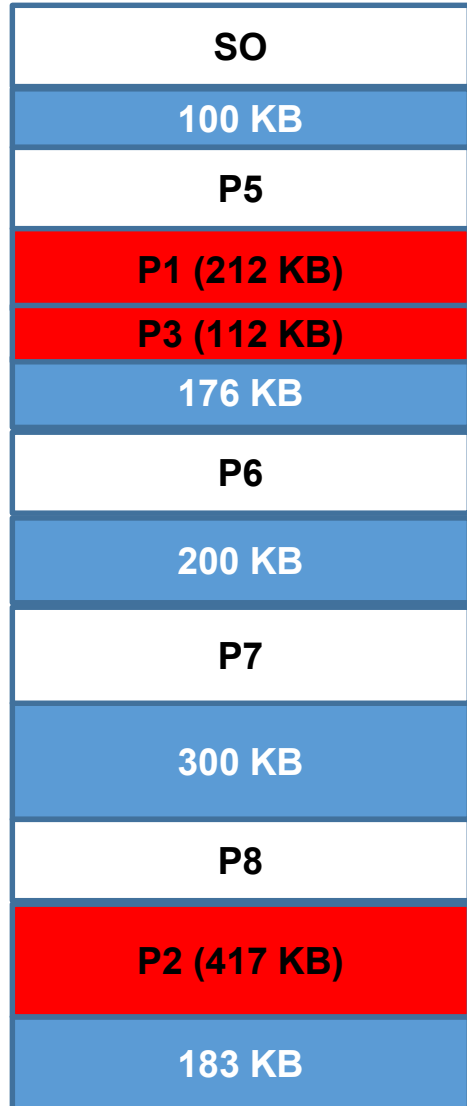
- P1 richiede 212 KB
- P2 richiede 417 KB
- P3 richiede 112 KB
- P4 richiede 426 KB



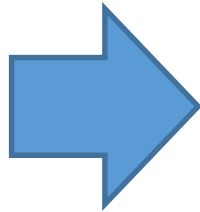
Soluzione Esercizio 1



Soluzione Esercizio 1



P4 (426 KB)



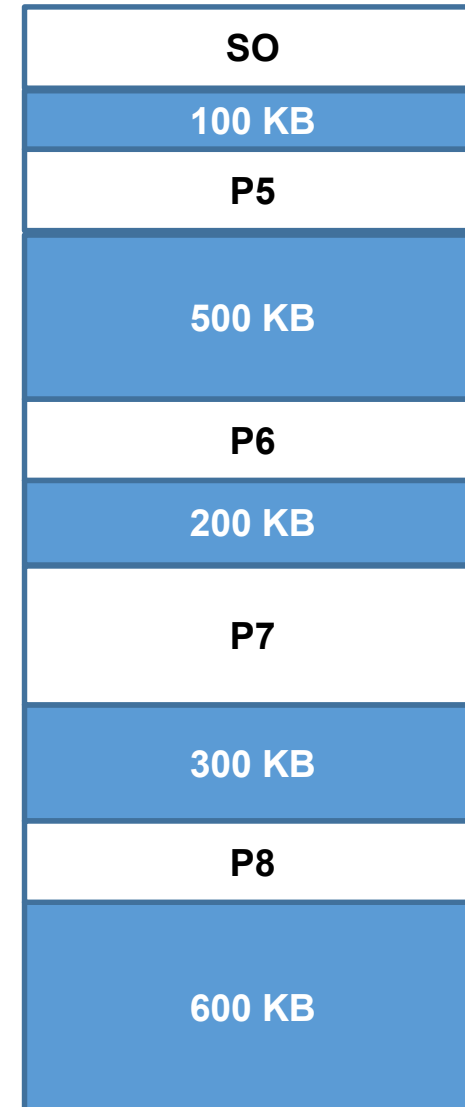
P4 (426KB) deve attendere

Esercizio 2

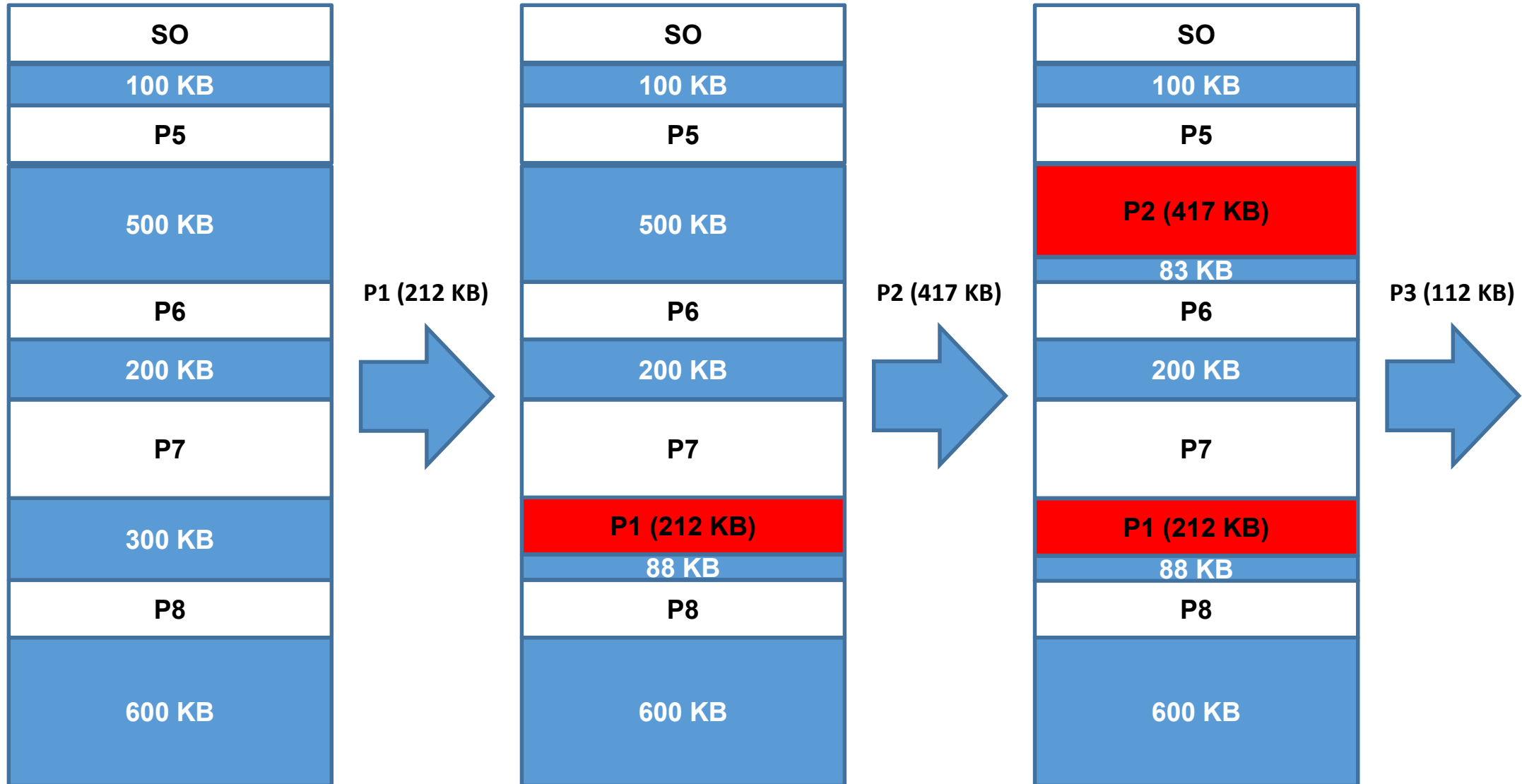
Si assuma di avere la memoria nella situazione illustrata a lato.

Utilizzando un approccio best-fit, come verranno allocati i processi seguenti?

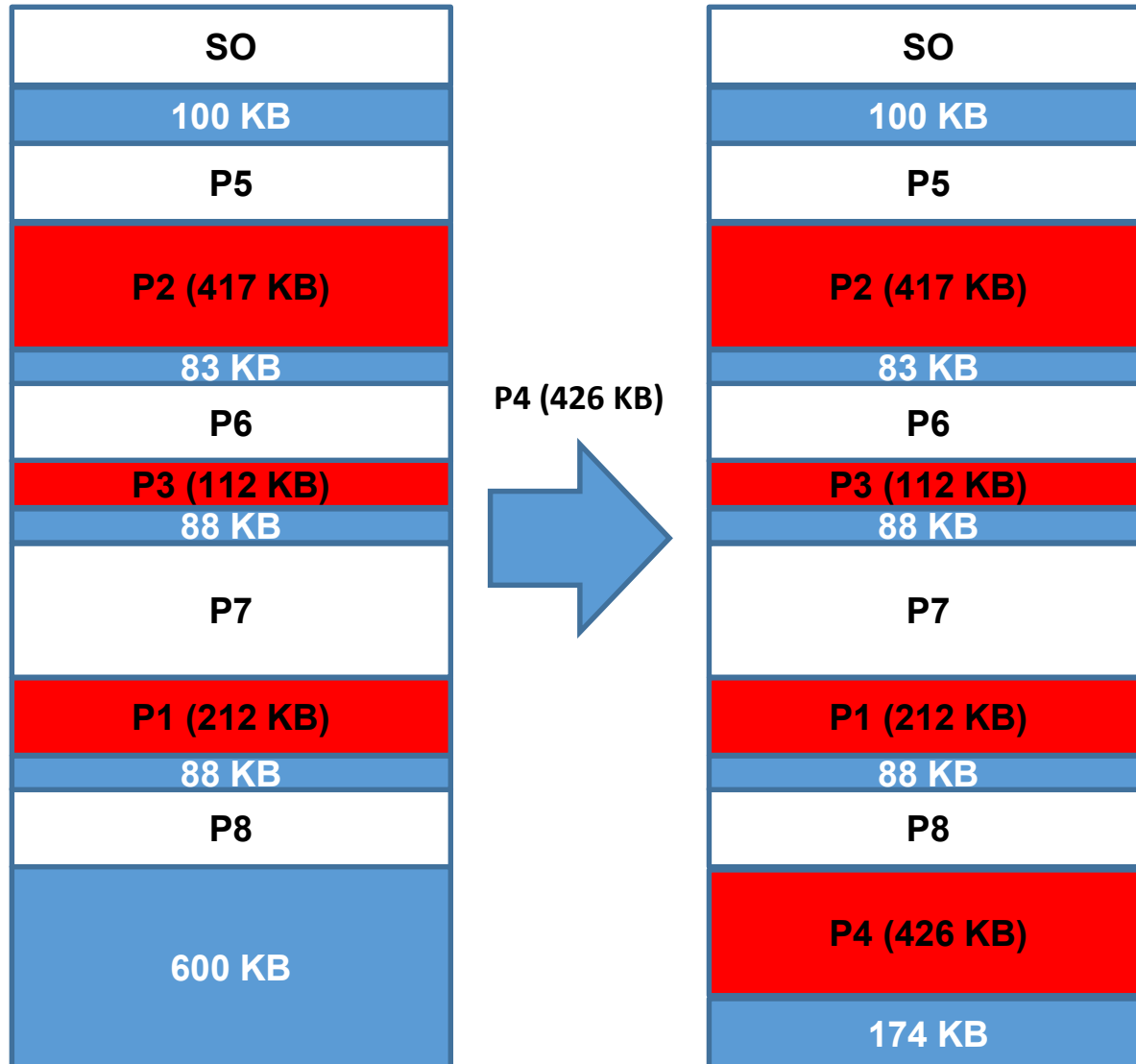
- P1 richiede 212 KB
- P2 richiede 417 KB
- P3 richiede 112 KB
- P4 richiede 426 KB



Soluzione Esercizio 2



Soluzione Esercizio 2

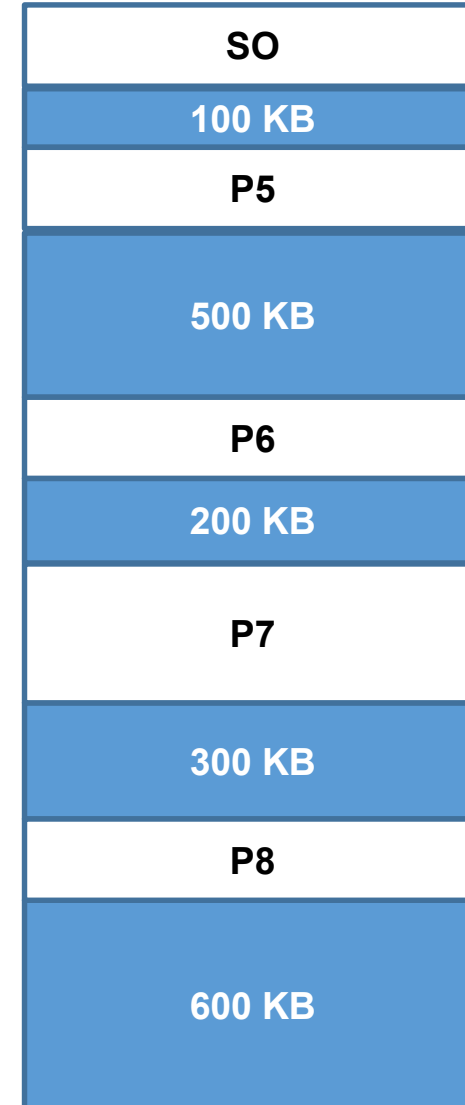


Esercizio 3

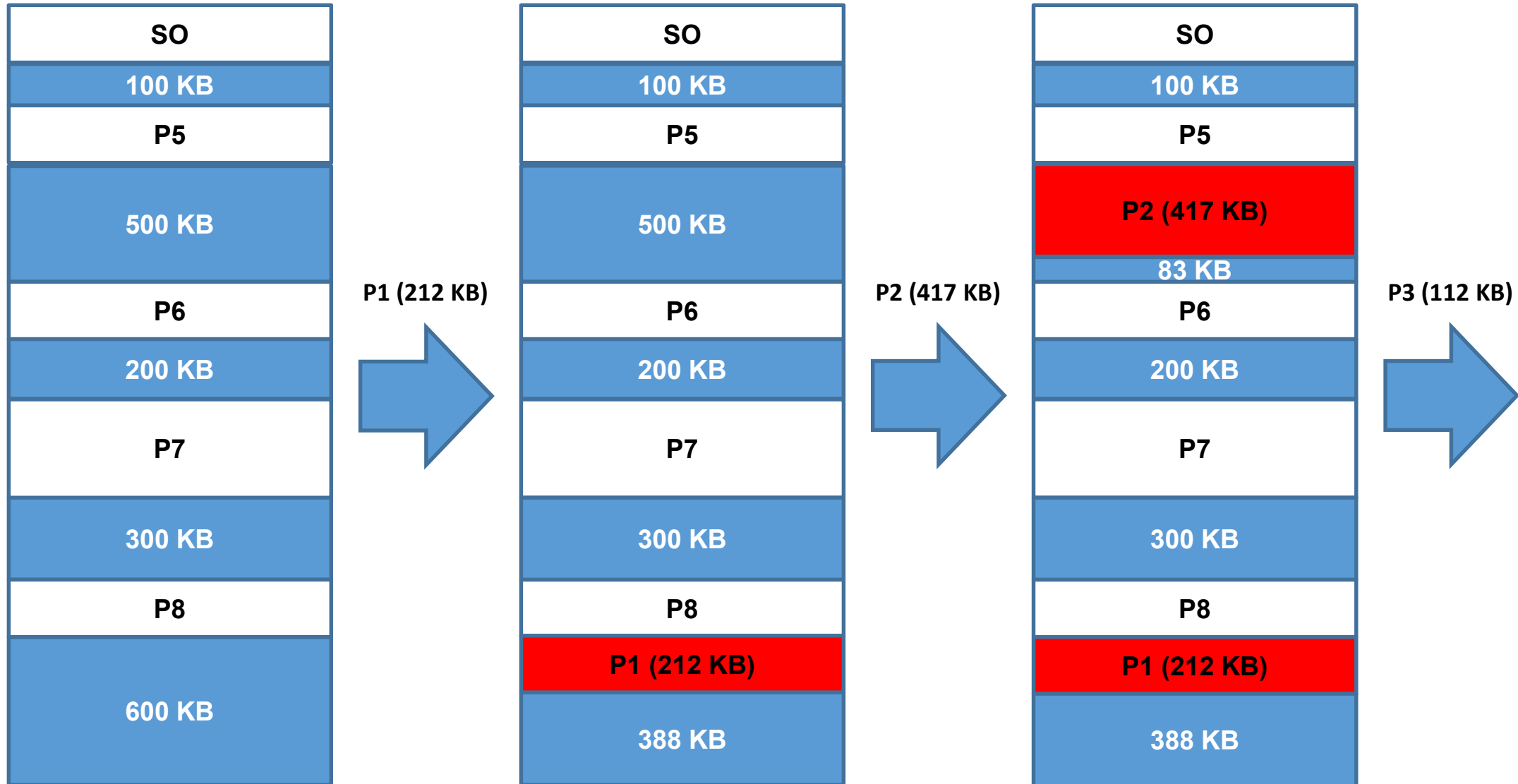
Si assuma di avere la memoria nella situazione illustrata a lato.

Utilizzando un approccio worst-fit, come verranno allocati i processi seguenti?

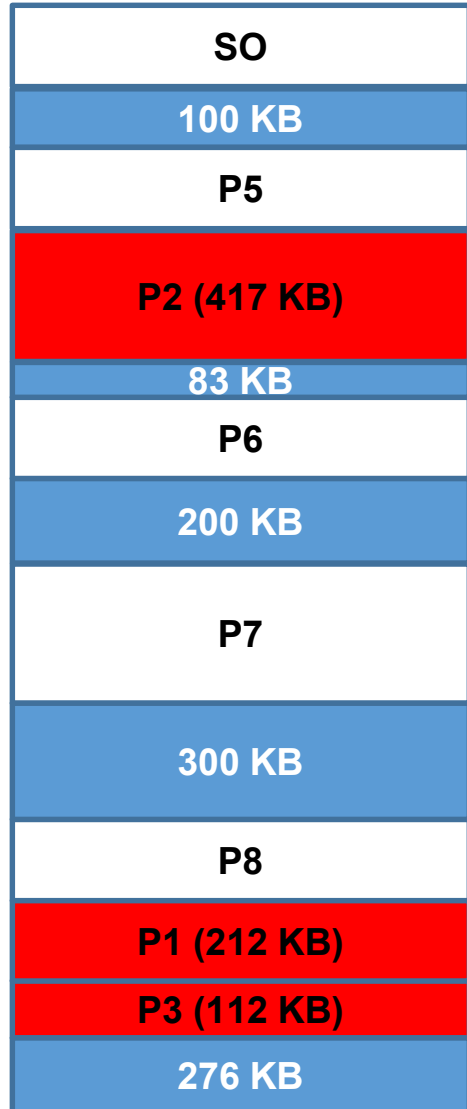
- P1 richiede 212 KB
- P2 richiede 417 KB
- P3 richiede 112 KB
- P4 richiede 426 KB



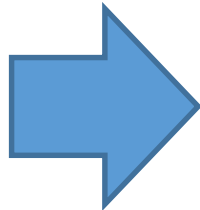
Soluzione Esercizio 3



Soluzione Esercizio 3



P4 (426 KB)



P4 (426KB) deve attendere

Esercizio 4

Si assuma di avere un sistema con

- una dimensione di pagina pari a 1KB
- uno spazio degli indirizzi logici di 15 bit

Di quante pagine può disporre il sistema?

Soluzione Esercizio 4

- Una dimensione di pagina pari a 1KB porta ad occupare 10 bit per l'indirizzamento poichè $1\text{KB} = 2^{10}\text{B}$
- Avendo uno spazio degli indirizzi logici di 15 bit, restano 5 bit da sfruttare.
- Quindi, il sistema potrà disporre di 2^5 pagine

Esercizio 5

Si assuma di avere un sistema con

- uno spazio degli indirizzi logici di 15 bit
- 8 pagine

Quanto sono grandi le pagine del sistema?

Soluzione Esercizio 5

- Il sistema dispone di 2^3 pagine
- Avendo uno spazio degli indirizzi logici di 15 bit, restano 12 bit da sfruttare.
- Una dimensione di pagina pari a 4KB porta ad occupare 12 bit per l'indirizzamento poichè $4KB = 2^{12}B$

In conclusione, il sistema avrà pagine da 4KB

Esercizio 6

Si consideri uno spazio di indirizzi logici di 8 pagine, ciascuna da 1KB, mappata su una memoria fisica da 32 frame.

Quanti bit servono per gli indirizzi logici?

Quanti bit servono per gli indirizzi fisici?

Soluzione Esercizio 6

- Il sistema dispone di 2^3 pagine
- Una dimensione di pagina pari a 1KB porta ad occupare 10 bit per l'indirizzamento poichè $1\text{KB} = 2^{10}\text{B}$
- Il sistema dispone di 2^5 frame

In conclusione, il sistema avrà uno spazio degli indirizzi logici da 13 bit e uno spazio degli indirizzi fisici da 15 bit.

Esercizio 7

Sia dato un sottosistema di memoria con paginazione, caratterizzato dalle seguenti dimensioni:

- frame 4 MB
- memoria fisica indirizzabile 128 GB

Si calcoli il numero di bit minimo per indicizzare tutte le pagine

Soluzione Esercizio 7

Data la dimensione di ogni pagina pari a 4 MB, saranno necessari 22 bit per indicizzare un elemento all'interno della stessa.

$$4 \text{ MB} \rightarrow 2^2 \times 2^{20} = 2^{22}$$

La memoria fisica, invece, necessita di almeno 37 bit.

$$128 \text{ GB} \rightarrow 2^7 \times 2^{30} = 2^{37}$$

Il numero di bit minimo per indicizzare tutte le pagine è quindi pari a $37 - 22 = 15$ bit

Esercizio 8

Sia dato un sottosistema di memoria con paginazione, caratterizzato dalle seguenti dimensioni:

- frame 16 MB
- memoria fisica indirizzabile 16 GB

Si calcoli il numero di bit minimo per indicizzare tutte le pagine

Soluzione Esercizio 8

Data la dimensione di ogni pagina pari a 16MB, saranno necessari 24 bit per indicizzare un elemento all'interno della stessa.

$$16 \text{ MB} \rightarrow 2^4 \times 2^{20} = 2^{24}$$

La memoria fisica, invece, necessita di almeno 34 bit.

$$16 \text{ GB} \rightarrow 2^4 \times 2^{30} = 2^{34}$$

Il numero di bit minimo per indicizzare tutte le pagine è quindi pari a $34 - 24 = 10$ bit

Esercizio 9

dimensione dello spazio degli indirizzi: 2^4

dimensione di una pagina: 4B

memoria fisica: 32 byte

Che indirizzo fisico corrisponde all'indirizzo logico 4?

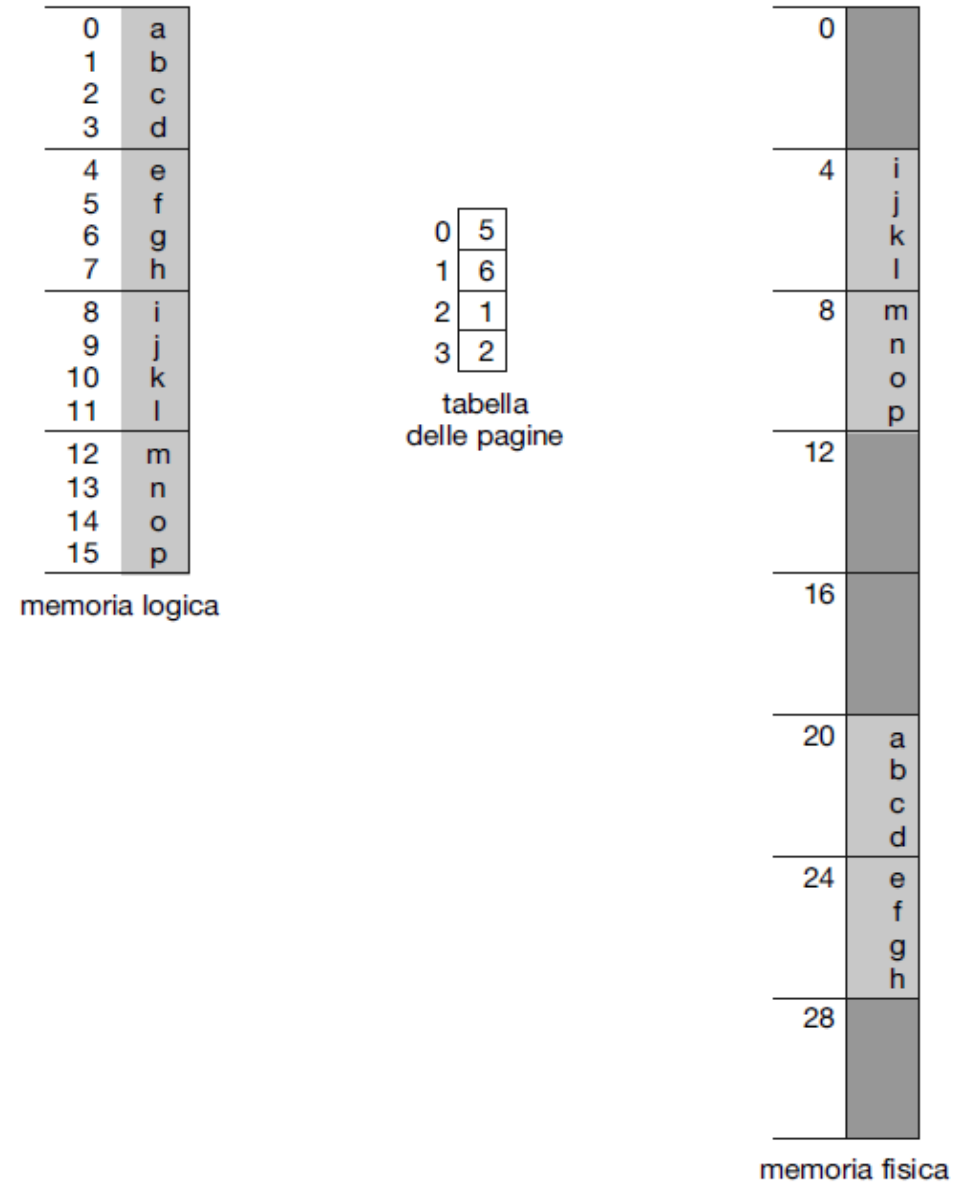


Figura 9.10 Esempio di paginazione per una memoria di 32 byte con pagine di 4 byte.

Soluzione Esercizio 9

Che indirizzo fisico corrisponde all'indirizzo logico 4?

indirizzo logico 4

pagina	offset
1	0

indirizzo fisico 24

frame	offset
6	0

$$(6 \times 4) + 0 = 24$$

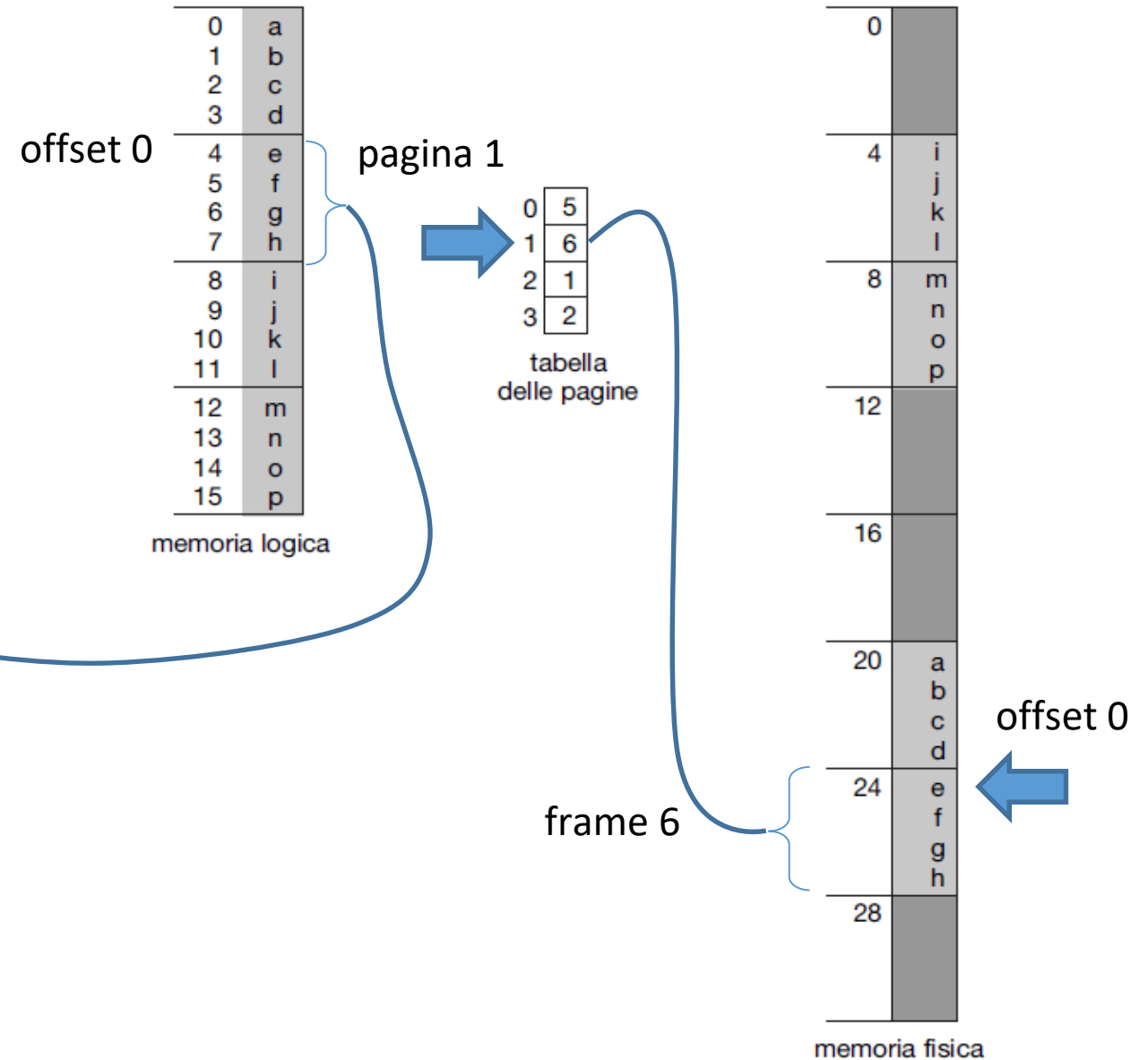


Figura 9.10 Esempio di paginazione per una memoria di 32 byte con pagine di 4 byte.

Esercizio 10

dimensione dello spazio degli indirizzi: 2^4

dimensione di una pagina: 4B

memoria fisica: 32 byte

Che indirizzo fisico corrisponde all'indirizzo logico 13?

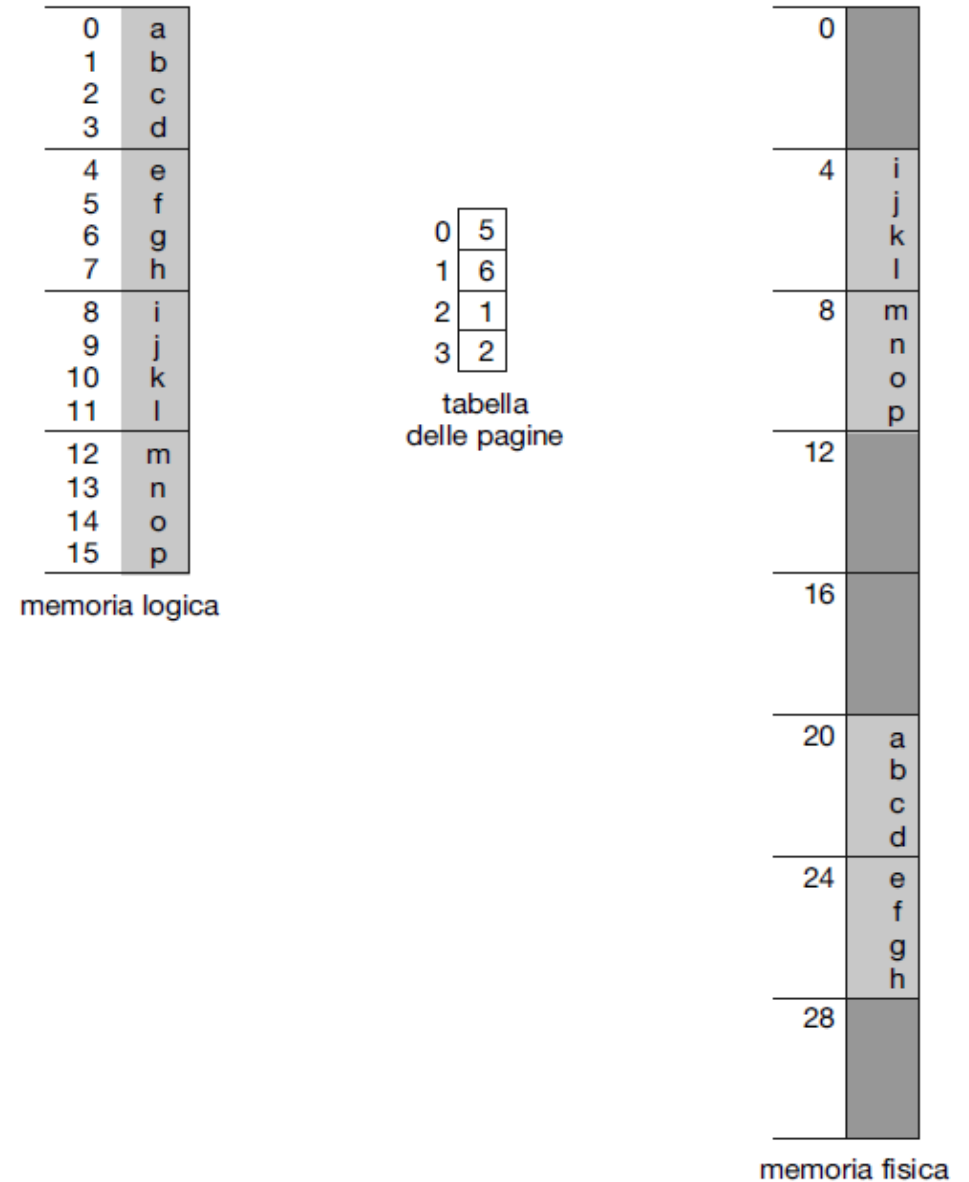
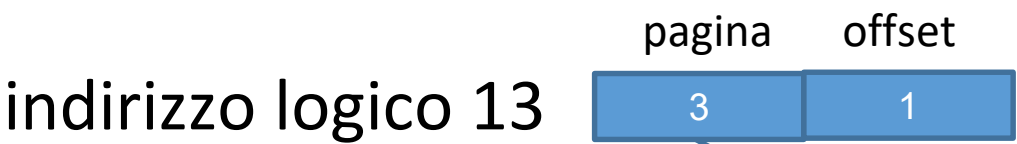


Figura 9.10 Esempio di paginazione per una memoria di 32 byte con pagine di 4 byte.

Soluzione Esercizio 10

Che indirizzo fisico corrisponde all'indirizzo logico 13?



$(2 \times 4) + 1 = 9$

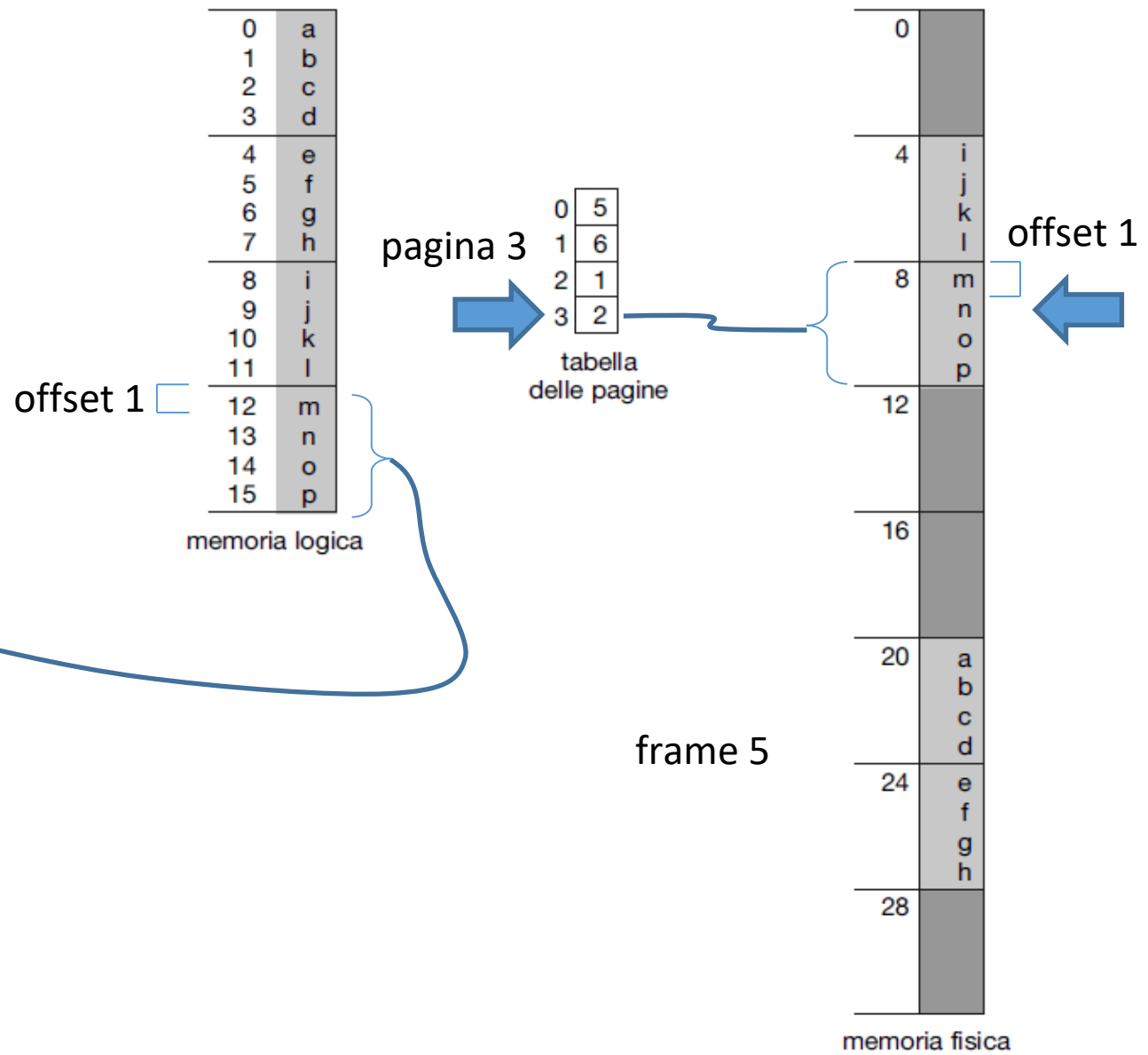


Figura 9.10 Esempio di paginazione per una memoria di 32 byte con pagine di 4 byte.

Esercizio 11

Sia dato un sistema di paginazione con una tabella delle pagine che risieda in memoria

Se per un accesso in memoria occorrono 200 nanosecondi, quanto tempo occorrerà per ottenere il dato relativo a un indirizzo logico?

Soluzione Esercizio 11

Tempo di accesso = $200 + 200 = 400$ ns

Per ottenere il numero del frame relativo ad un indirizzo logico occorreranno 400 nanosecondi, poichè occorrono 200 ns per accedere alla tabella delle pagine in RAM e 200 ns per accedere al dato in memoria

Esercizio 12

Sia dato un sistema di paginazione con una tabella delle pagine che risieda in memoria, avente un TLB con un hit ratio del 90%

Se per un accesso in memoria occorrono 200 nanosecondi, quanto tempo occorrerà per ottenere il dato relativo a un indirizzo logico?

Si supponga per semplicità che cercare una entry nella tabella delle pagine richieda tempo pari a 0 ns

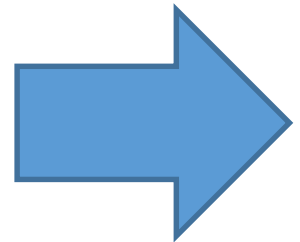
Soluzione Esercizio 12

Caso 1: il numero di pagina desiderato si trova nel TLB

Tempo di accesso = $0 + 200 = 200$ ns

Caso 2: il numero di pagina desiderato NON si trova nel TLB,
sono quindi necessari 2 accessi in memoria

Tempo di accesso = $200 + 200 = 400$ ns



Soluzione Esercizio 12

Il tempo effettivo di accesso alla memoria sarà quindi

tempo

$$\begin{array}{l} \text{effettivo} \\ \text{di accesso} \end{array} = 0.9 \times 200\text{ns} + (1-0.9) \times 400\text{ns} = 220\text{ns}$$

Memoria virtuale

La **memoria virtuale** è una tecnica che permette di eseguire processi che possono anche non essere completamente contenuti in memoria

La **memoria virtuale** facilita la programmazione, poiché il programmatore non deve preoccuparsi della quantità di memoria fisica disponibile, ma può concentrarsi sul problema da risolvere con il programma

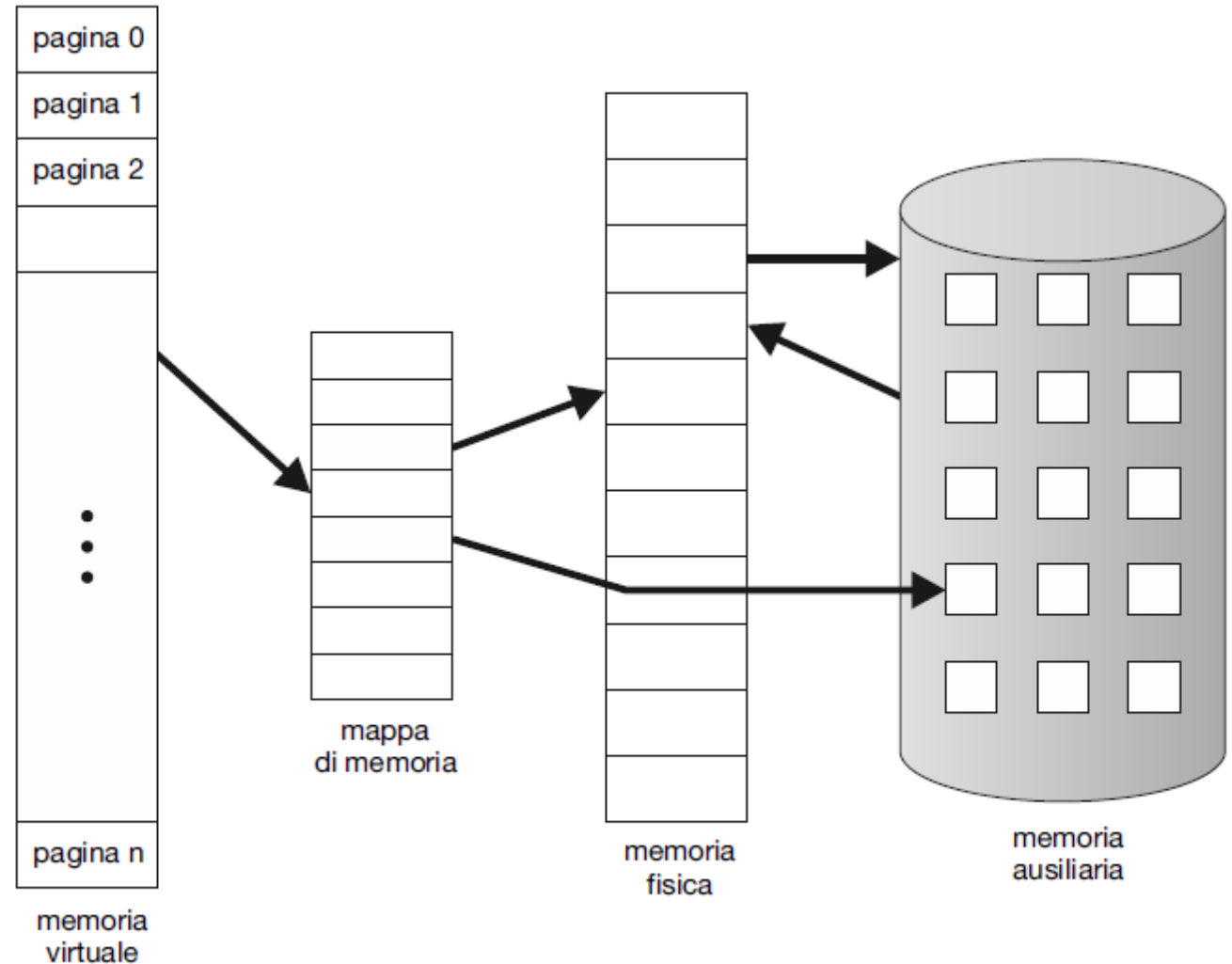


Figura 10.1 Schema che mostra una memoria virtuale più grande di quella fisica.

Esercizio 13

Si consideri un sistema a memoria virtuale con

- indirizzi virtuali a 32 bit
 - indirizzi fisici a 24 bit
 - pagine di 4K byte
-
- a) di quante pagine sono costituiti rispettivamente lo spazio di indirizzamento virtuale e quello fisico?
 - b) di quanti bit è costituito l'offset?
 - c) di quante righe è costituita la tavola delle pagine?
 - d) come è organizzata una tavola delle pagine a due livelli corrispondenti rispettivamente a gruppi di 12 e 8 bit?
 - e) quante pagine occupa un processo da 9734 byte?

Soluzione Esercizio 13

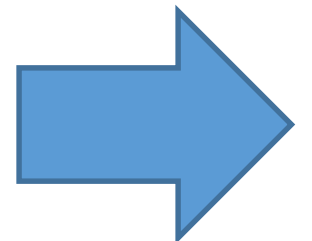
a) di quante pagine sono costituiti rispettivamente lo spazio di indirizzamento virtuale e quello fisico?

- Essendo gli indirizzi virtuali a 32 bit lo spazio virtuale è costituito da 2^{32} byte
- Le pagine sono da 4 K byte = 2^{12} byte, pertanto il numero di pagine virtuali è di:

$$2^{32}/2^{12} = 2^{20} = 1 \text{ M pagine}$$

- Essendo gli indirizzi fisici a 24 bit lo spazio fisico è costituito da 2^{24} byte
- Le pagine sono da 4 K byte = 2^{12} byte, pertanto il numero di pagine fisiche è di:

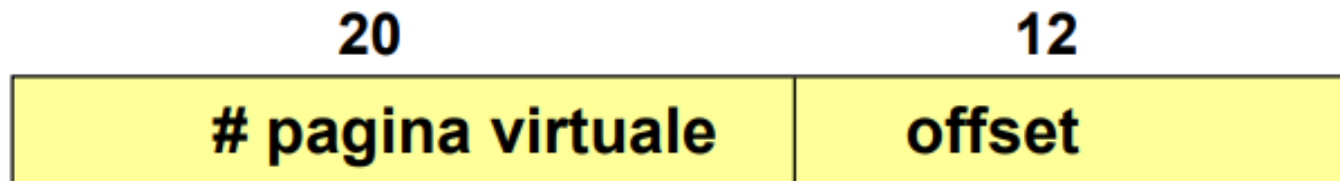
$$2^{24}/2^{12} = 2^{12} = 4 \text{ K pagine}$$



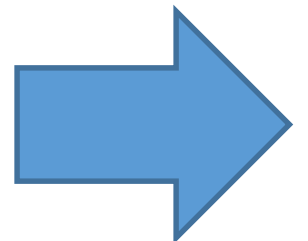
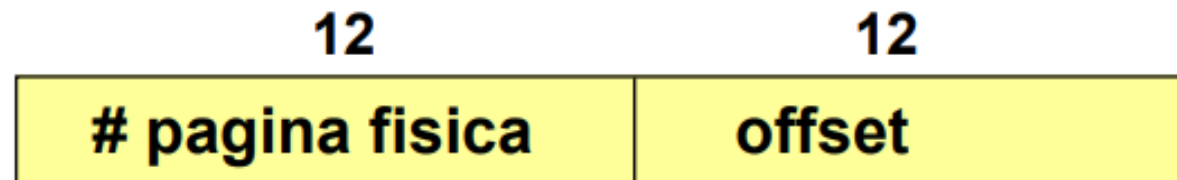
Soluzione Esercizio 13

b) di quanti bit è costituito l'offset?

- L'offset dipende solo dalla dimensione delle pagine
- Dato che le pagine sono da 4 K byte = 2^{12} byte, l'offset è di 12 bit
- Struttura dell'indirizzo virtuale:



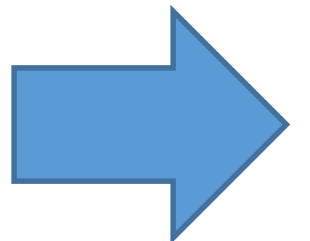
- Struttura dell'indirizzo fisico



Soluzione Esercizio 13

c) di quante righe è costituita la tavola delle pagine?

- La tavola delle pagine ha un elemento per ogni pagina dello spazio di indirizzamento virtuale. Le pagine virtuali sono 2^{20}



Soluzione Esercizio 13

d) Come è organizzata una tavola delle pagine a due livelli corrispondenti rispettivamente a gruppi di 12 e 8 bit?

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

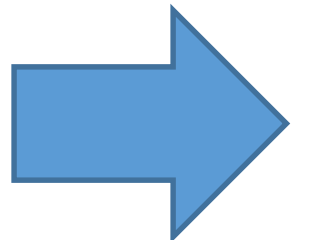
$$2^{12} \cdot 2^2 \text{ byte} = 2^{14} \text{ byte} = 16 \text{ Kbyte}$$

— Ciascuna delle *tavole di secondo livello* ha 2^8 elementi di 2^3 byte ciascuno, quindi occupa complessivamente:

$$2^8 \cdot 2^3 \text{ byte} = 2^{11} \text{ byte} = 2 \text{ Kbyte}$$

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

$$2^{12} \cdot 2^{11} \text{ byte} = 2^{23} \text{ byte} = 8 \text{ Mbyte}$$



Soluzione Esercizio 13

e) quante pagine occupa un processo da 9734 byte?

- Il processo viene allocato in pagine da 4 Kbyte = 4096 byte.
- Il numero di pagine occupate dal processo è dato da:

$$P = \lceil 9734/4096 \rceil = 3$$

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

$$S = 3 \cdot 4096 - 9734 = 2554 \quad (\text{pari al } 2554/4096 \cong 62\%)$$

- Questo spreco corrisponde al fenomeno della *frammentazione interna*

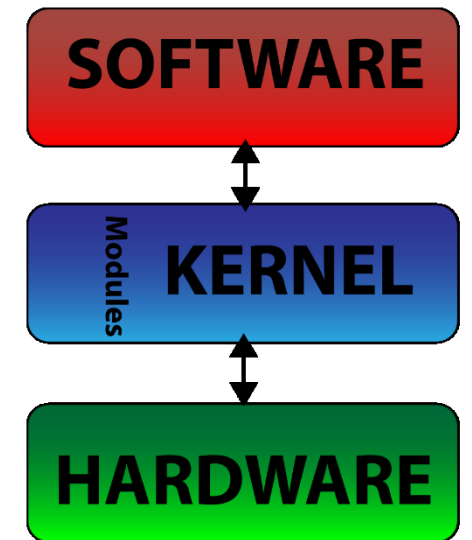
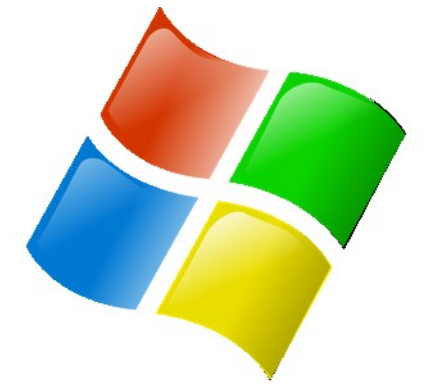


**UNIVERSITÀ DEGLI STUDI
DELLA BASILICATA**

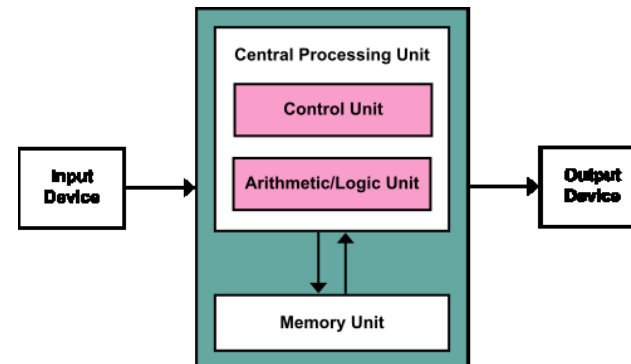
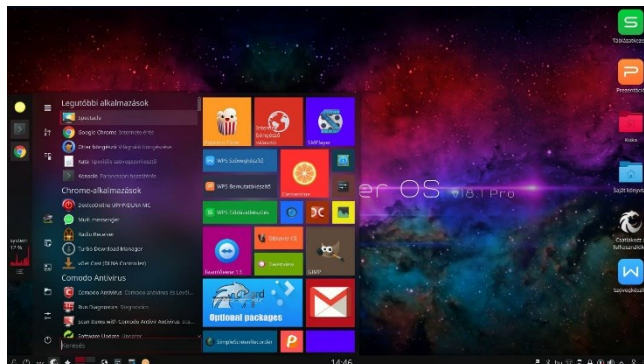
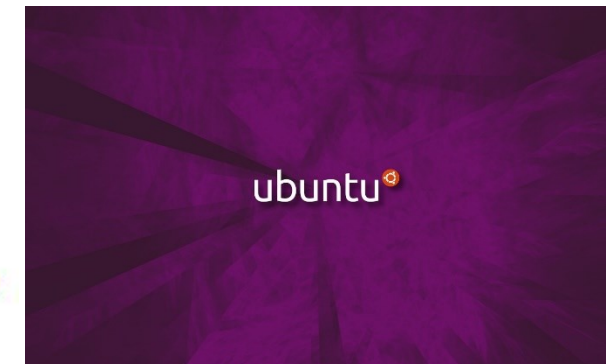
*Corso di Sistemi Operativi
A.A. 2019/20*

Esercitazione

Memoria centrale



Docente:
**Domenico Daniele
Bloisi**



Novembre 2019