

INDIRIZZI GENERATI DALLA CPU

SPAZIO DEGLI INDIRIZZI LOGICI DEL PROCESSO → BINDING → SPAZIO DEGLI INDIRIZZI FISICI DEL PROCESSO

MMU = memory management unit

- Elabora l'indirizzo logico in un indirizzo fisico

LINKING & LOADING

- Vengono commesse porzioni di codice presenti su più file
- Carica il codice nella RAM perché possa essere eseguito

Sono detti **STATICI** quando le due operazioni vengono eseguite prima dell'esecuzione, **DINAMICI** quando avvengono dopo/durante l'esecuzione

Con il linking/loading DINAMICI non tutte le librerie vengono caricate

- prima ma solo quando utilizzate

posizione di codice

STUB → identifica la collocazione delle procedure e associa il suo codice all'invocazione

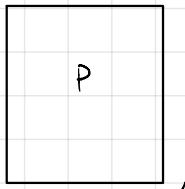
Il loader dinamico recupera il codice delle procedure

se successo, sostituisce se stesso con l'indirizzo della procedura (completa il binding)
se FALLIMENTO, causa il cincimento (LOADER) del codice necessario

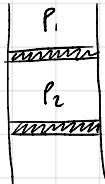
APPROCCI DELL'ALLOCAZIONE DELLA RAM AI PROCESSI

- ① ALLOCAZIONE CONTIGUA
- ② PAGINAZIONE
- ③ SEGMENTAZIONE

1-



Partizione contigua



Ogni processo può accedere a una
regione di memoria intera ma
queste sono separate da sezioni di
RAM

So: PAGINAZIONE + TLB

HIT RATIO : 32% conta 1 tempo di accesso al TLB + 1 tempo di accesso alla RAM
TLB MISS : 8% conta 2 tempi di accesso alla RAM + 1 tempo di accesso al TLB

TEMPO MEDIO DI ACCESSO ALLE PAGINE fatti:

$$t_{TLB} = 20$$

$$t_{RAM} = 100$$

$$t_{MEDIO} = 120 \cdot 0,32 + 120 \cdot 0,08 = 110,4 + 17,6 \approx 128$$

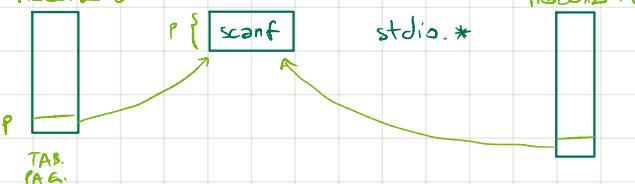
$t_{TLB} + t_{RAM}$ $2t_{RAM} + t_{TLB}$

CONDIVISIONE DI PAGINE

- Il contesto di paginazione semplifica il meccanismo di linking/loading dinamici

esempio

Processo 0

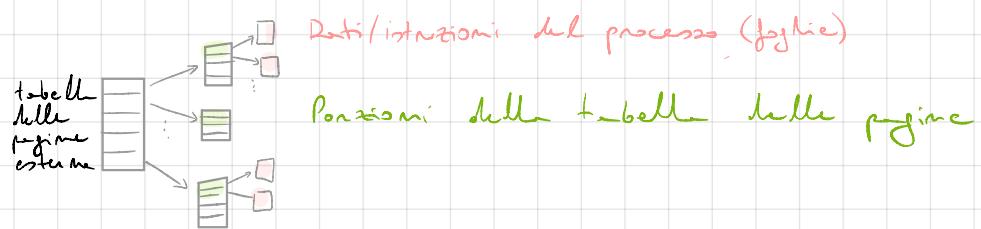


Si possono utilizzare i meccanismi visti precedentemente per l'utilizzo di pagine in lettura e scrittura da parte di più processi

IMPLEMENTAZIONE DELLA TABELLA DELLE PAGINE

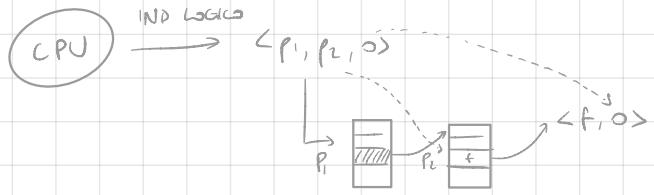
PAGINAZIONE MULTILIVELLO

Utilizzo un albero a 2 livelli

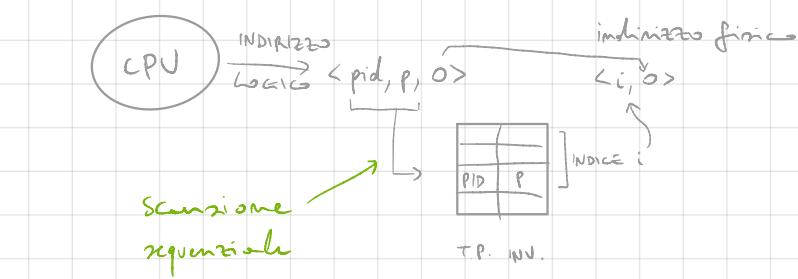
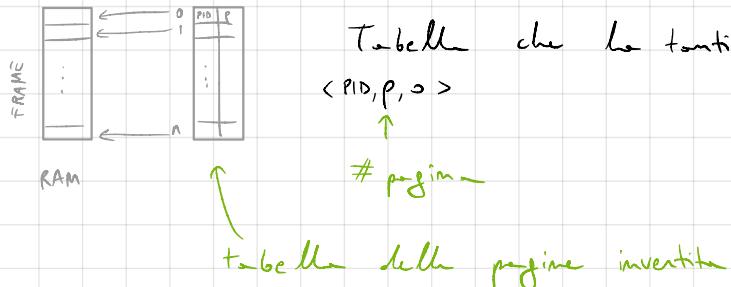


Utilizzo la tripleta $\langle p_1, p_2, o \rangle$ dove:

- p_1 sarà l'indice della tabella delle pagine esterna
- p_2 è un riferimento al frame della pagina più interna
- o è l'offset

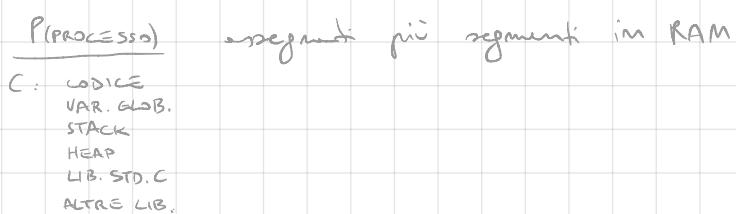


- HASH TABLE
- "PUNTO DI VISTA DELLA RAM"

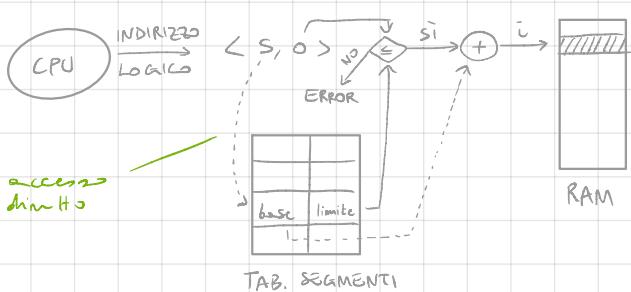
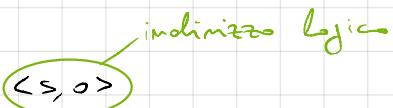


- SEGMENTAZIONE

Rilascia il vincolo di contiguità come nella paginazione ma posso solo avere segmenti omogenei (senza elementi misti)



Ottieniamo una tabella dei segmenti



Torna il problema della FRAMMENTAZIONE

• SEGMENTAZIONE PAGINATA

Ogni frammento è multiplo della dimensione di un frame riducendo il problema della frammentazione

