

# **UD9 Sistemi**

Gestione della Memoria

Prof. Viglietti Francesco

IV B Info

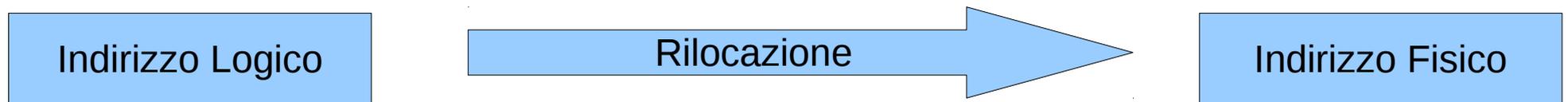
A.s. 2010-11

# Spazio logico e spazio fisico



Ogni programma eseguibile occupa uno spazio logico di memoria

Il programma eseguibile caricato anche parzialmente in memoria occupa uno spazio fisico.



Caricamento del programma in memoria può essere:

- Statico → il programma viene caricato per intero fino a terminazione
- Dinamico → vengono caricate e scaricate continuamente parti di programma

# Memory manager

Il gestore della memoria si occupa della gestione delle informazioni, delle politiche di assegnazione, controlla i passaggi di stato per i programmi che attendono assegnazioni e caricamento da periferiche.

Esso gestisce la memoria in modo da:

- Non limitare la creazione di altri processi
- Consentire l'avanzamento dei processi
- Ridurre numero e durata delle sospensioni per attese di caricamento di codice e dati

# Caricamento Statico – Partizioni fisse

Lo spazio di memoria viene suddiviso, dall'amministratore del sistema in fase di boot, in partizioni di dimensioni diverse. Ogni programma è caricato in una partizione fino alla terminazione. La dimensione della partizione dev'essere superiore al programma, ma se molto superiore si ha un eccessivo frammentazione. Per la gestione serve una tabella che abbia numero, dimensione, indirizzo, stato



# Esempio - Partizioni fisse

Supponendo che nella partizione 1 sia contenuto un programma di 5K, nella 2 uno di 28K, nella 5 uno di 390K, Determinare la dimensione della memoria e la distribuzione. Trovare inoltre la percentuale di spreco di memoria dovuta alla frammentazione.

Numero	Dimensione	Indirizzo	Stato
1	8K	312K	uso
2	32K	320K	uso
3	32K	352K	libera
4	120K	384K	libera
5	520K	504K	uso

# Caricamento Statico – Partizioni variabili

A differenza dal modello precedente le partizioni vengono create dal OS. Inizialmente vengono associate partizioni della dimensione del processo, quanto però i processi terminano il OS deve gestire l'allocazione e la redistribuzione della frammentazione creatasi dai vari processi. Servono due tabelle per la gestione, una tabella delle partizioni occupate e una delle partizioni libere.



# Esempio - Partizioni variabili

Un sistema a partizioni variabili si trova nello schema indicato sotto, si verificano le seguenti richieste di carico e rilascio di memoria: carico Pa da 200K, rilascio del P1, rilascio di P3, carico di Pb da 80K, rilascio di P2. Determinare le prestazioni di FF e BF

Numero	Dimensione	Indirizzo	Stato
1	8K	312K	uso
2	32K	320K	uso
3	24K	352K	uso
4	120K	384K	uso
5			vuoto

Numero	Dimensione	Indirizzo	Stato
1	8K	376K	libero
2	520K	504K	libero
3			vuoto
4			vuoto
5			vuoto

# Caricamento Statico – Partizioni variabili, strategie di allocazione

Scelta della partizione idonea ad ospitare il programma. I metodi più comuni sono:

First Fit → Il programma viene inserito nella prima partizione sufficiente a contenerlo

Best Fit → Il programma viene inserito nella partizione la cui dimensione sia più vicina a quella richiesta

Worst Fit → Il programma viene inserito nella partizione la cui dimensione sia più grande possibile, in cui il frammento abbia la dimensione massima

# Partizionamento rilocabile

È il modo per risolvere il fenomeno della frammentazione. Consiste nel ricompattare tutte le aree libere in un'unica area contigua di memoria posta in un'estremità dello spazio totale, così da ottenere nuove aree libere riutilizzabili.

# Caricamento Statico – Segmentazione

Consiste nella suddivisione del programma in segmenti, di dimensione scelta dal programmatore. Viene usato il metodo delle partizioni variabili. Questo metodo tende a diminuire la frammentazione. Oltre alla tabella delle partizioni servirà una tabella dei descrittori per ogni processo...

# Caricamento Statico – Paginazione