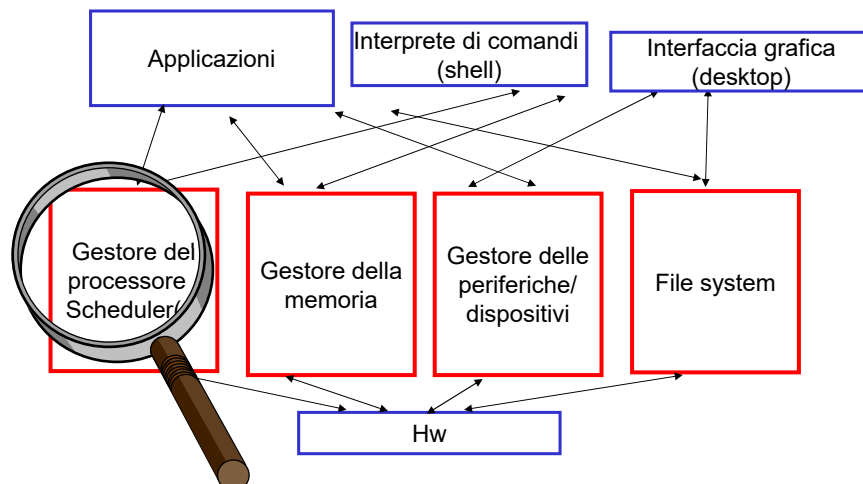


Organizzazione di un SO monolitico



1

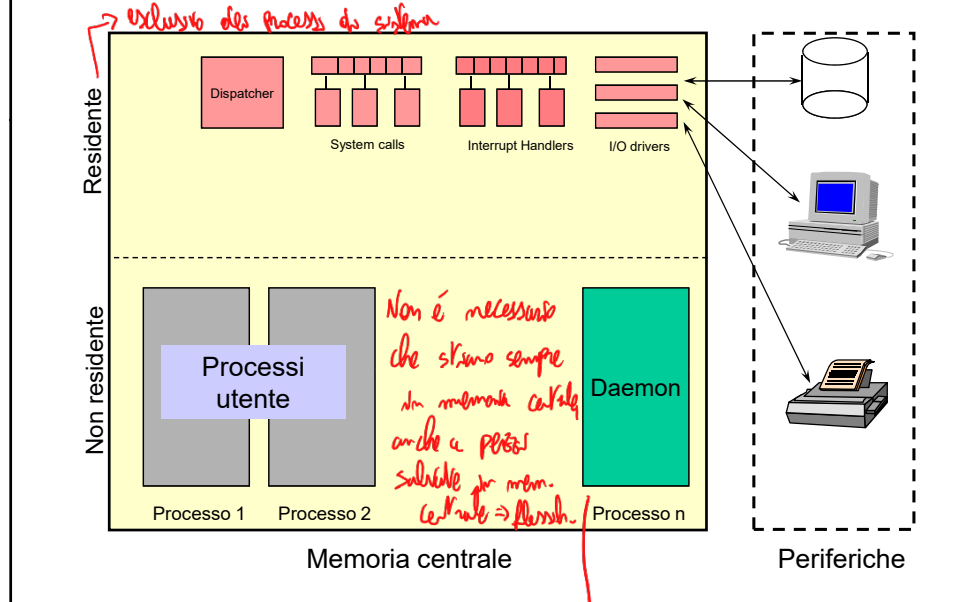
Processi

- Processo (Def.) :
programma in esecuzione completo del suo stato (spazio di indirizzamento, contenuto dei registri, file aperti...)
- Il concetto di processo è centrale nella organizzazione di ogni SO
- Tipicamente ad ogni istante ci sono molti processi attivi contemporaneamente (tutto o parte dello spazio di indirizzamento dei processi attivi è residente in memoria centrale)

2

Una serie di spazi occupati da processi utente e processi del sistema operativo

Modello (parziale) di riferimento di un S.O.



Processi sempre attivi che puntano all'arrivo dell'elab. e aspettano richieste e servizi

Dal programma al processo

Programma

Passo 1 : compilazione e creazione del file eseguibile

Compilatore

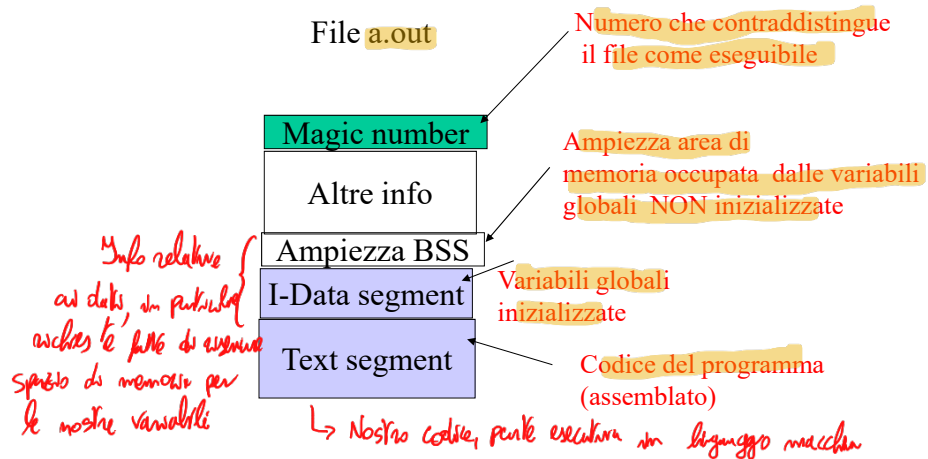
Eseguibile

È un file binario che

- * contiene tutte le informazioni necessarie all'esecuzione del programma da parte del processore
- * ha un formato che dipende dal SO che deve curarne l'esecuzione
- * è memorizzato su disco

Formato del file eseguibile

- Un esempio : il formato ELF di Linux



5

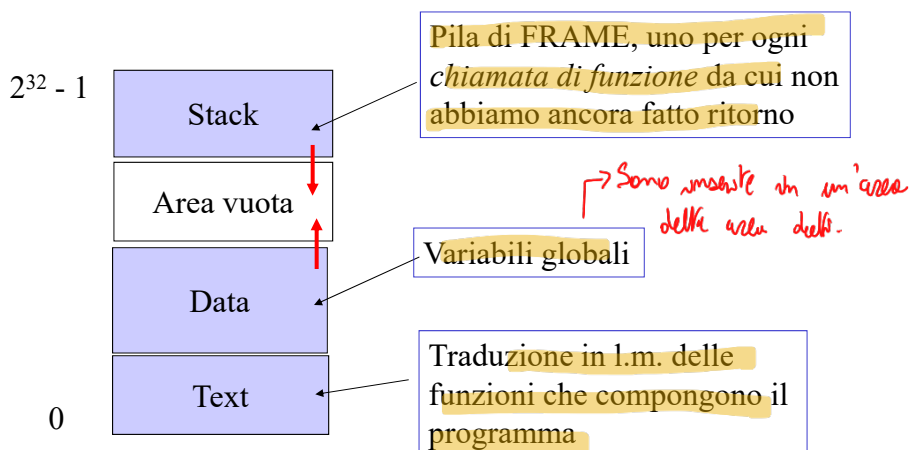
Caricamento di un programma



6

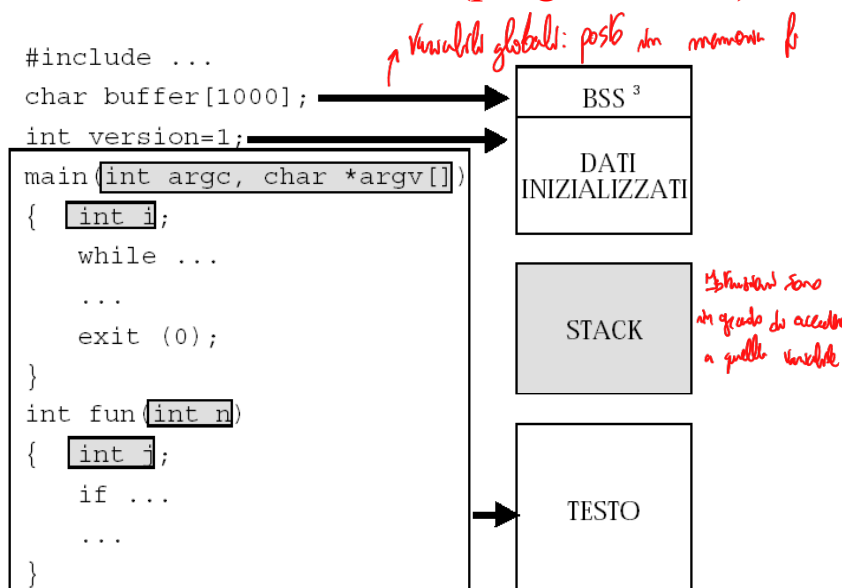
Spazio di indirizzamento

- Come è organizzata la memoria accessibile ad un programma in esecuzione ?



7

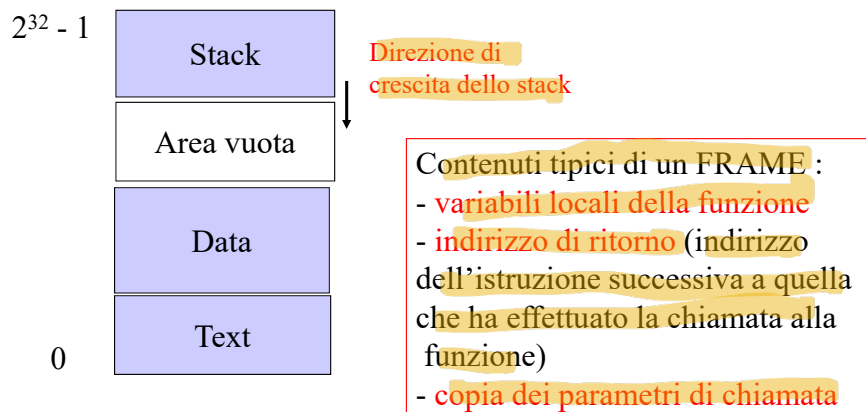
Spazio di Indirizzamento (programma C)



8

Spazio di indirizzamento (2)

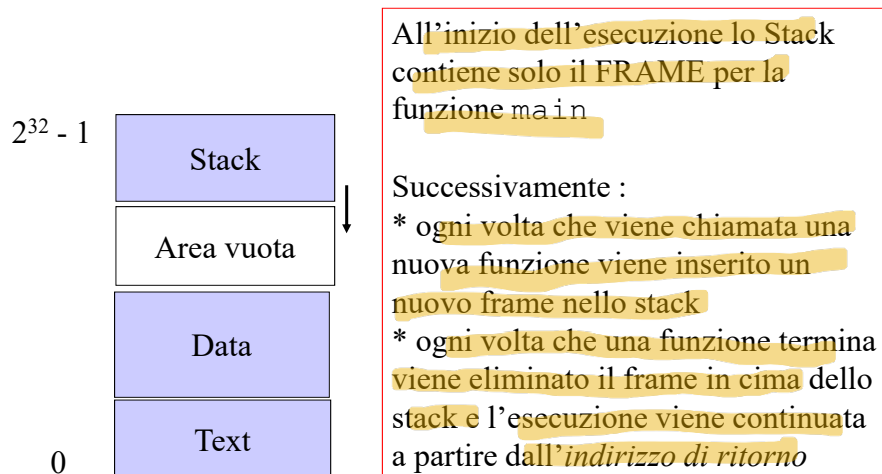
- Spazio di indirizzamento tipico (caso del linguaggio C)



9

Spazio di indirizzamento (3)

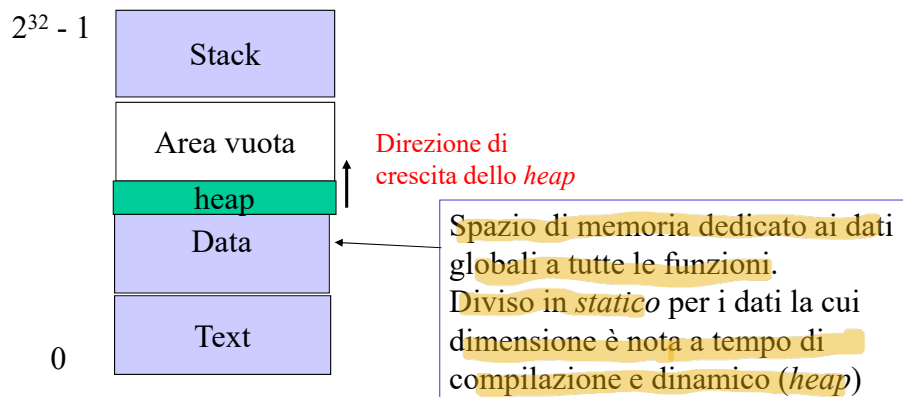
- Spazio di indirizzamento tipico (caso del linguaggio C)



10

Spazio di indirizzamento (4)

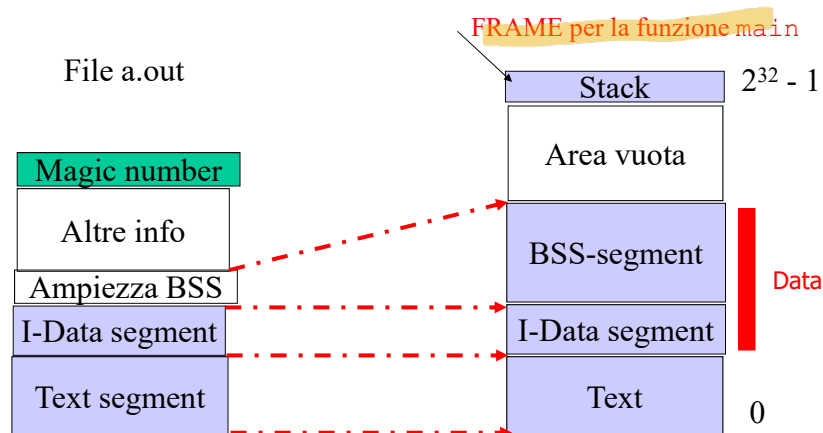
- Spazio di indirizzamento tipico (caso del linguaggio C)



11

Spazio di Indirizzamento (5)

- L'eseguibile contiene tutte le informazioni per creare la configurazione iniziale dello spazio di indirizzamento (*loading*)



12

Esecuzione di un programma

attivazione del programma, ovvero caricamento nel PC dell'indirizzo della prima istruzione da eseguire nell'area testo

A questo punto il programma ha il controllo del processore

Il Sistema Operativo potrà tornare in esecuzione solo se si verifica uno dei seguenti eventi :

- arrivo di una interruzione hw*
- terminazione del programma*
- invocazione esplicita di un servizio tramite una System Call*

13

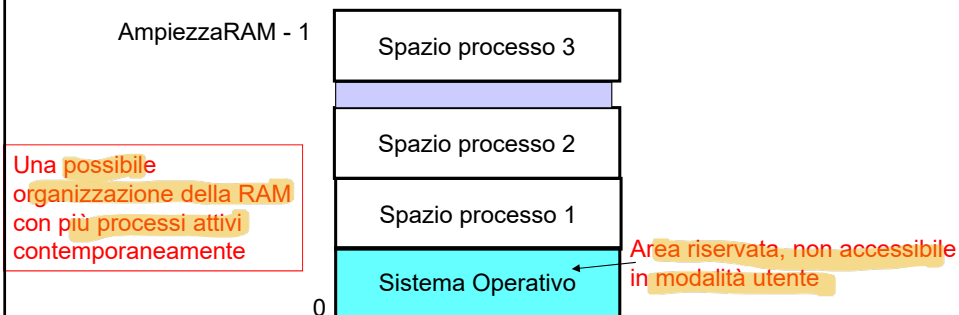
Il modello a processi sequenziali

- Nei computer attuali, ci sono molte attività attive contemporaneamente (sia di SO che di utente)
 - es : stampa in corso + Word processor + cd player ...
- Il modello a processi sequenziali : è un modello concettuale che permette di esprimere in modo semplice più attività contemporanee
 - ogni attività è incapsulata in un processo e viene programmata come se utilizzasse il calcolatore in modo esclusivo
 - l'interazione fra processi avviene solo con l'invocazione di un numero (limitato e fissato) di meccanismi di cooperazione (di IPC)
meccanismi ben precisi

14

Condivisione della RAM

- Tipicamente la RAM contiene lo spazio di indirizzamento di più processi :
 - es. organizzazione tipica della memoria negli SO degli anni '70 (es IBM 360)



Condivisione della RAM (problemi)

- Problemi legati alla condivisione della RAM :
 - (1) protezione dello spazio di indirizzamento di processi diversi

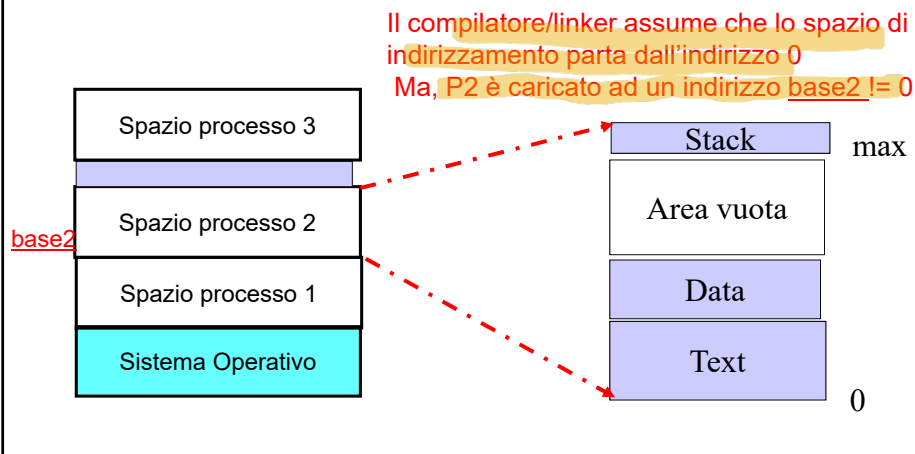


Indirizzi memorizzati all'interno del codice devono essere aggiornati sulla base di dove i dati sono organizzati in memoria

Condivisione della RAM (problemi)

- Problemi legati alla condivisione della RAM :

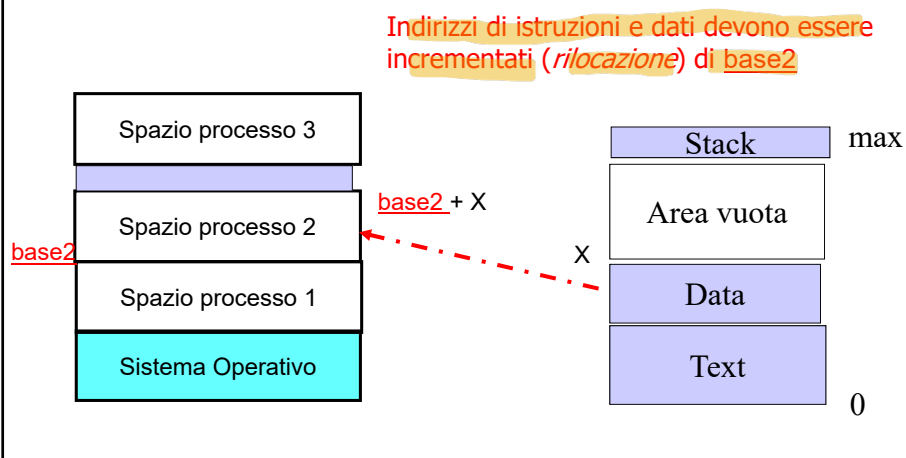
- (2) problema della rilocalizzazione



Spazio di indirizzamento è corretto solo se programma salvato a partire da 0. Allora spazio di indirizzamento viene aggiornato con un offset che viene sommato. Indirizzo sarà modo a meno di una costante iniziale

Condivisione della RAM (problemi)

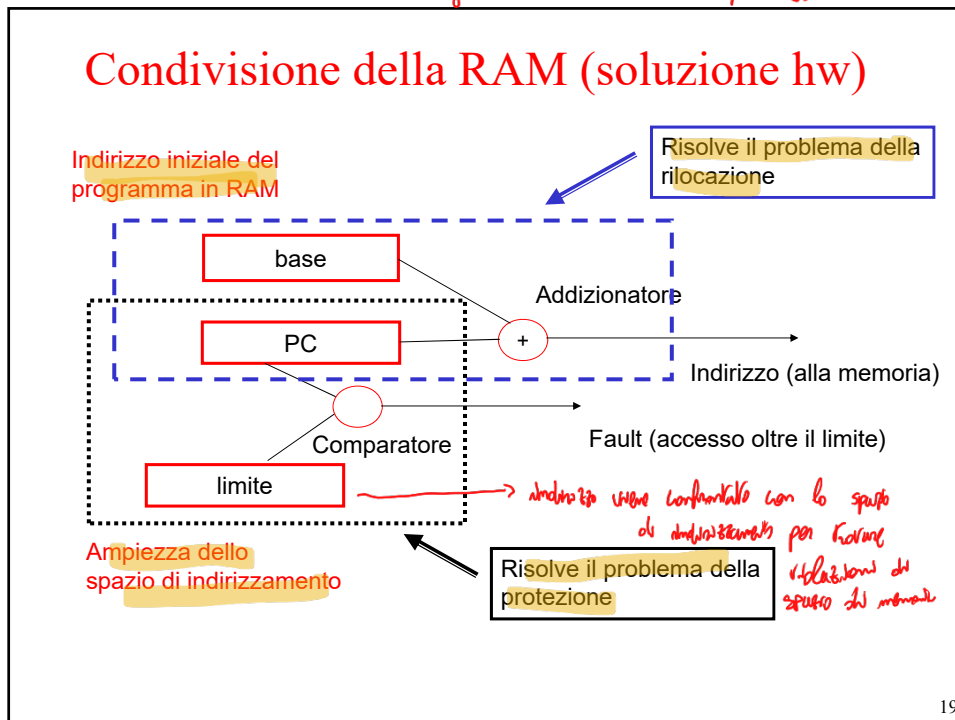
- Necessità di un meccanismo di rilocalizzazione :



Usiamo dei registri, ci basiamo sull'hardware

Ogni volta che un processo viene allocato a partire dall'indirizzo base2, basta che sommo base2 a ogni cosa. Quindi ora vanno sempre aggiunti.

Condivisione della RAM (soluzione hw)



Tutto così semplice se lo spazio è allocato in indirizzi consecutivi e sono tutti salvati completamente in memoria.
La rilocalizzazione non risolve il problema della gestione della memoria: se devono essere tutti consecutivi, avranno buchi nella memoria che non possono essere utilizzati: frammentazione della memoria.

Creazione di un processo

- Come si crea un processo?
 - Inizializzazione del sistema: Accensione PC
 - Richiesta da parte di un processo al SO per la creazione di un altro processo (attraverso chiamate di sistema)
 - es: fork - exec in Unix
 - es: CreateProcess in API Win32
 - L'invocazione da parte di un utente di un comando o di un'applicazione (il doppio click su un'icona, il lancio di un eseguibile ...)
 - L'inizio di un job batch (ad esempio job che vengono attivati ad una particolare ora del giorno)

→ Come faccio a far partire quello primordiale? Fase di inizializzazione si risolve con meccanismi hardware

↓
Alternativa: deframmentare e spostare per risolvere buchi.

Franco Sinatra



fallin

fall

I can't help
fallin' in love
with you