

Macchine Astratte



Definizione e tipi di
implementazione

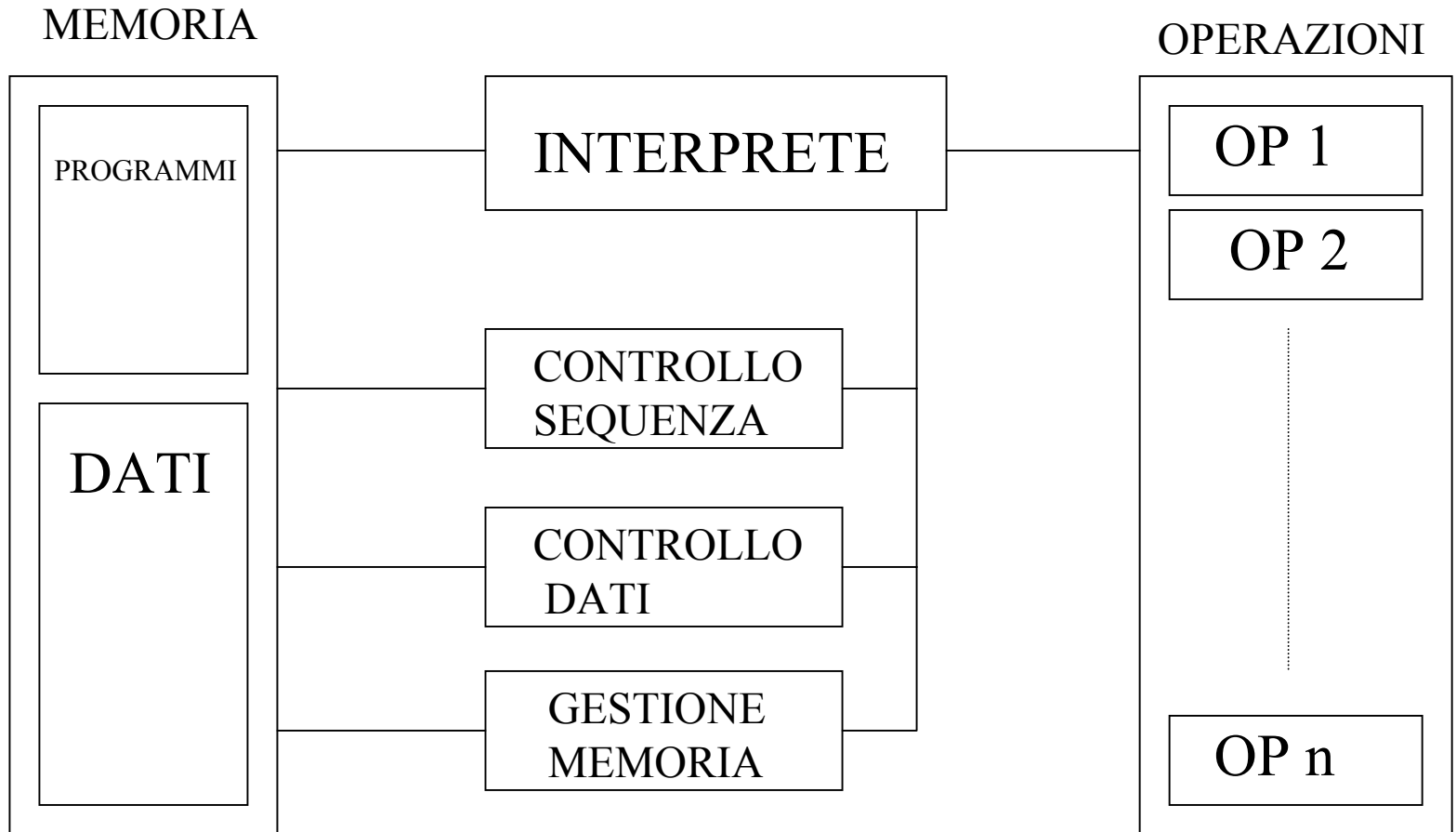


Macchine astratte

- Sono un insieme di concetti che sintetizzano le caratteristiche base di ogni linguaggio di programmazione
- Sono un modello di riferimento per la definizione e l'implementazione dei linguaggi
- Costituiscono un'ontologia del nucleo del “mondo” dei linguaggi di programmazione
- Spiegano in maniera precisa i concetti di compilazione, interpretazione e soluzioni implementative miste

Macchina Astratta

un insieme di **strutture dati** ed **algoritmi** in grado di **memorizzare** ed **eseguire** programmi





Es: Una macchina fisica come macchina astratta

- Operazioni Primitive
 - Operazioni aritmetico-logiche
 - Operazioni di manipolazione di stringhe di bit
 - Lettura/Scrittura di celle di memoria e registri
 - Input/output
- Controllo di Sequenza (salti, condizionali, chiamate e ritorni dai sottoprogrammi)
 - Registro contatore istruzioni (PC)
 - Strutture dati che contengono i punti di ritorno dei sottoprogrammi



Es: Una macchina fisica come macchina astratta

■ Controllo dati

- Acquisizione operandi
- Memorizzazione risultato
- Architettura a registri:
 - Registri indice
 - Indirizzamento indiretto
- Architettura a Pila:
 - Gestione della Pila



Es: Una macchina fisica come macchina astratta

■ Gestione della memoria

– Architettura a registri:

- Nessuna poiché la memorizzazione è statica

– Architettura a Pila:

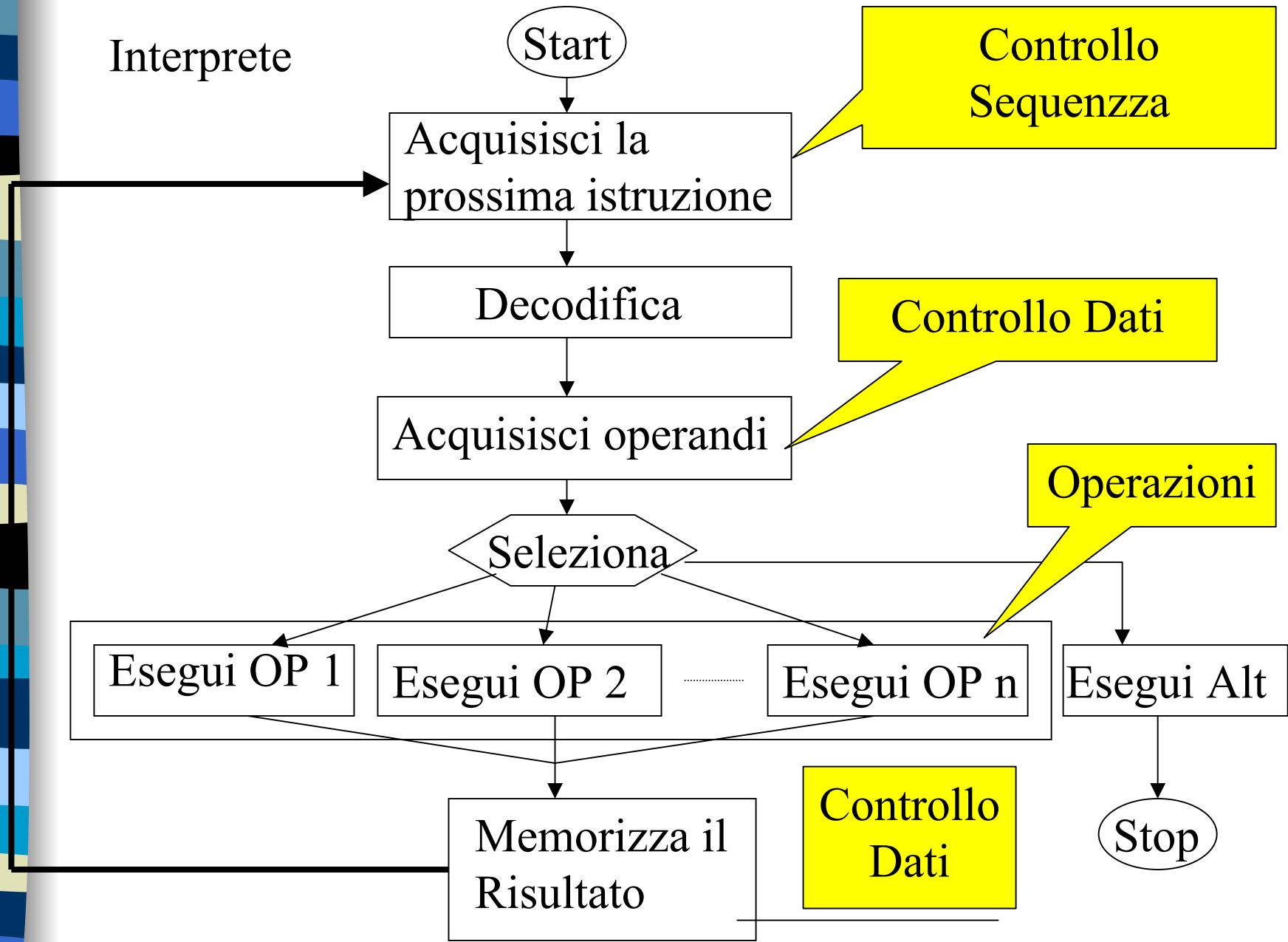
- Allocazione e recupero dei dati sulla Pila



L'interprete

- La struttura dell'interprete è sempre la stessa per una qualunque macchina astratta
- Quello che cambia sono le altre componenti

Interprete



Controllo Sequenza

Controllo Dati

Operazioni

Controllo Dati

Stop



Il linguaggio di una macchina astratta

- M macchina astratta
- L_M linguaggio “macchina” di M: è il linguaggio in cui si esprimono tutti i programmi interpretati dall’interprete di M
- I programmi sono particolari dati primitivi su cui opera l’interprete



Rappresentazione dei programmi

- Interna: strutture dati in memoria
- Esterna: stringhe di caratteri (scritte dal programmatore)
- La conversione fra forma esterna e forma interna è realizzata dal **caricatore** (loader)
- L_M denota uno qualunque dei due



Macchine astratte

- Ai componenti di M corrispondono i componenti di L_M
 - Tipi di dato primitivi
 - Meccanismi per il controllo della sequenza
 - Meccanismi per il controllo del trasferimento dei dati
 - Meccanismi per la gestione della memoria



Realizzazione di Macchine Astratte

- Una macchina astratta è una collezione di strutture dati ed algoritmi
- Può essere realizzata combinando 3 tecniche
 1. Realizzazione in hardware
 2. Emulazione o simulazione via firmware
 3. Simulazione software



Realizzazione in hardware

- In linea di principio è sempre possibile
- E' una soluzione che soffre di scarsa flessibilità
- Il costo di realizzazione è molto alto per macchine astratte complesse
- E' realmente utile solo per macchine astratte (e linguaggi) di basso livello



Realizzazione in firmware

- Si realizzano le strutture dati e gli algoritmi mediante **microprogrammi** su una macchina (ospite) microprogrammabile
- I microprogrammi risiedono in una speciale memoria ROM e vengono eseguiti ad altissima velocità



Realizzazione in software

- Concettualmente identica all'emulazione
- Non richiede macchine ospiti particolari
- Chiaramente meno efficiente



Realizzazione dell'interprete

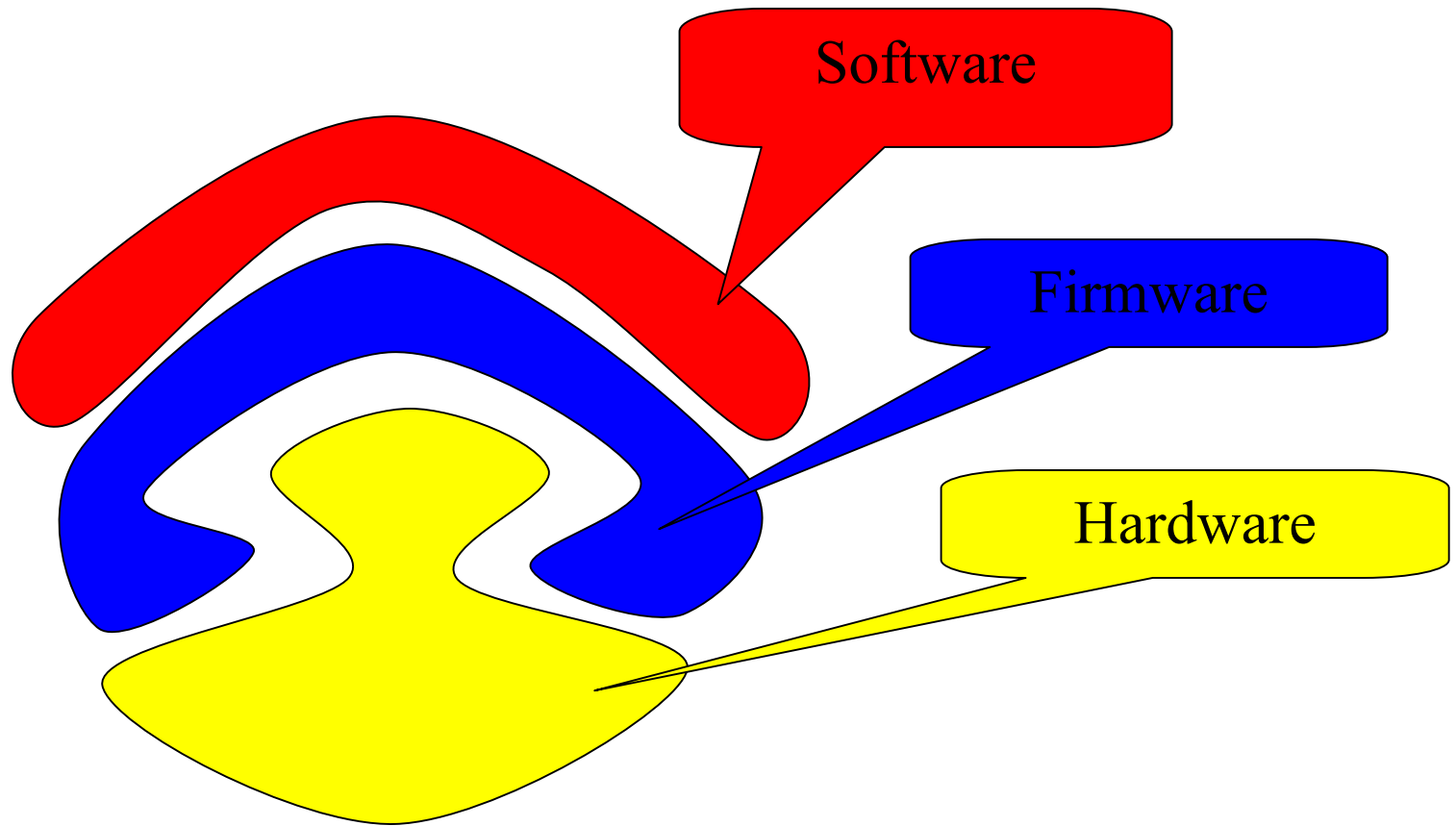
- L'interprete di una macchina astratta M può essere:
 - Lo stesso della macchina ospite M_0
 - M è un'estensione di M_0 di cui condivide almeno l'interprete
 - Simulato
 - M ed M_0 sono “diversi” perché hanno un diverso cervello, cioè diversi interpreti, anche se possono coincidere su altre componenti
- Distinzione importante per i concetti di interpretazione e compilazione



Realizzazione software su una macchina ospite

- M macchina astratta
- M_0 macchina ospite (hardware, eventualmente microprogrammabile)
- Alcune componenti di M possono coincidere con quelle di M_0
- Alcune componenti (simulate) di M possono essere realizzate in firmware

La realizzazione alla fine è una
combinazione delle tre





Dai linguaggi alle macchine astratte

- $M \rightarrow L_M$
- $L \rightarrow M_L$ è la macchina astratta che ha L come linguaggio macchina
- Se L è un linguaggio ad alto livello, M_L può essere molto complessa
- Implementare L vuol dire realizzare M_L
- Come?



Implementazione di M_L

- Generalmente mediante simulazione (software od eventualmente firmware) su una macchina ospite M_0
- Se l'interprete di M_L è simulato, l'implementazione si chiama **interpretativa**
- Esiste un'alternativa basata su tecniche di traduzione (soluzione **compilativa**)