

01_Rielaborazione

Questo file è la rielaborazione delle slide [01_Architettura.pdf](#)

Un'**algoritmo** è un insieme finito di istruzioni usate per la risoluzione di un certo lavoro.

Quest'ultimo molte volte viene definito come una particolare funzione parziale di una [macchina di Turing](#) (Alan Turing, matematico britannico) oppure come un programma di una [macchina di Von Neumann](#) (matematico statunitense).

Le funzioni di base di un'elaboratore possono essere riassunte in 4 macro aree:

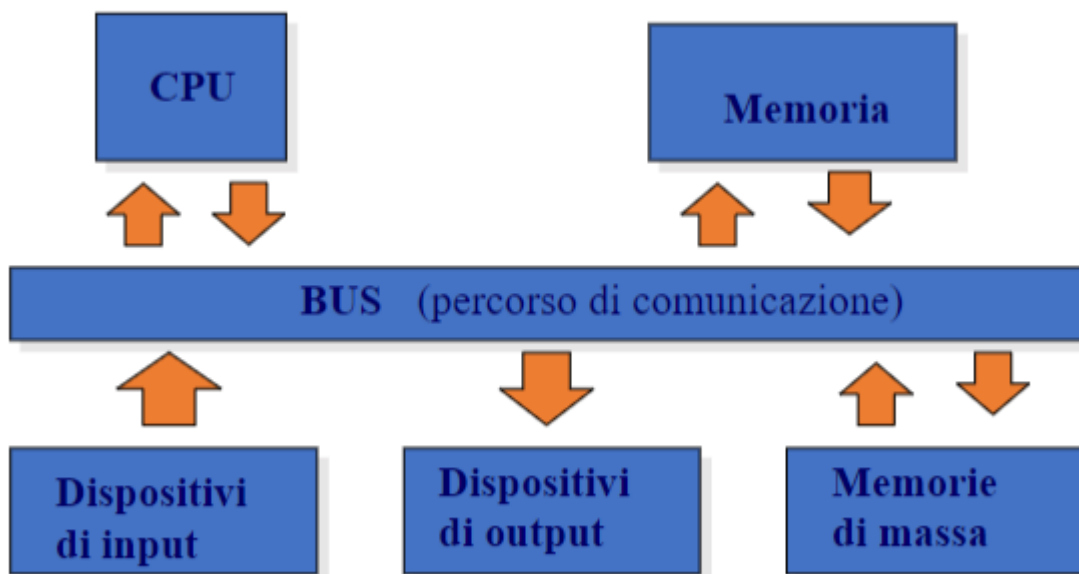
- memorizzazione dei dati
- elaborazione dei dati
- trasferimento dei dati
- controllo.

Un'esempio di elaboratore è il **computer**, il computer è una macchina che computa ovvero che esegue un certo algoritmo (ovviamente scritto in modo che la macchina stessa lo possa interpretare). Esistono vari tipi di computer:

- Laptop
- Server
- Desktop
- Tablet/Smartphone

Un qualsiasi computer moderno segue la seguente architettura:

macchina di Von Neumann



Questo tipo di architettura viene chiamata Macchina di Von Neuman (questa è l'architettura alla quale fa riferimento la definizione di algoritmo sopracitata). Di seguito una descrizione di tutti i componenti:

- La **CPU**, o **Central Processing Unit**, è l'unità centrale di elaborazione di un computer. È il componente principale che esegue le istruzioni dei programmi, gestisce le operazioni logiche e aritmetiche, e coordina il funzionamento delle altre parti del sistema. Come memoria di lavoro usa i registri e la cache. La sua velocità si misura in numero di cicli al secondo (MHz o GHz). La CPU è formata da 4 parti fondamentali:
 - il **program counter**: una locazione di memoria che contiene l'indirizzo dell'istruzione da eseguire
 - il **registro delle istruzioni**: una locazione di memoria contenente l'istruzione da eseguire
 - **ALU (Arithmetic logic unit)**: un sistema che esegue le operazioni aritmetiche e logiche
 - **CU (Control Unit)**: Il sistema di controllo gestisce il flusso di esecuzione delle istruzioni. Fa sì che ogni parte del processore faccia il suo lavoro nel momento giusto, attraverso una serie di cambiamenti di stato.
- La **memoria**, è un deposito di dati e di istruzioni da eseguire, ne esistono di 3 tipi:
 - **ROM, o Read Only Memory**, è una memoria di sola lettura non volatile dove si trovano tutte varie informazioni come le istruzioni usate per l'avvio del pc, o dei parametri necessari per il corretto funzionamento del dispositivo
 - **RAM, o Random Access Memory**, è una memoria volatile utilizzata per immagazzinare tutte quelle istruzioni che poi vengono eseguite dal processore

- La **cache** è una memoria ad alta velocità che si trova all'interno o vicino alla CPU e viene utilizzata per immagazzinare temporaneamente i dati e le istruzioni più frequentemente utilizzate. La sua funzione principale è quella di ridurre i tempi di accesso alla RAM migliorando così le prestazioni del sistema e quindi riducendo il **Bottleneck** tra CPU e RAM.
- **Dispositivi di input**: come un tastiera ed un mouse
- **Dispositivi di output**: come un monitor o una stampante
- **Memoria di massa**: un tipo di memoria non volatile che viene usata per immagazzinare grandi file, e ovviamente molto più lenta di qualsiasi altro tipo di memoria sopracitata. La memoria di massa è fondamentale per garantire l'archiviazione stabile di software, documenti, immagini e altri dati digitali, rendendoli accessibili nel tempo. Degli esempi sono:
 - **Hard disk drive (HDD)**: basato su dischi magnetici rotanti.
 - **Solid State Drive (SSD)**: usa chip di memoria flash, più veloce e resistente rispetto agli HDD.
- **BUS**, tutte queste componenti comunicano tra di loro attraverso i bus. I principali tipi di bus vengono usati per inviare dati o segnali di controllo

La macchina di von Neumann viene definita logicamente come una terna, ovvero un'insieme di 3 elementi:

- **N** = {0,1,2,3} ovvero l'insieme dei numeri naturali (l'alfabeto della macchina)
- **IS** = {ZERO, INC, SOM, SOT, MOL, DIV, UGUALE, MINORE, SALCOND, ALT} è l'Instruction Set ovvero un'insieme di istruzioni che la macchina può usare
- **P** = {I0, I1, I2, I3, ... , |P| - 1} è una sequenza finita di istruzioni prese dall'insieme IS, questo insieme si chiama programma

Come possiamo ben notare un programma eseguibile dalla macchina von Neumann consiste in una lista di istruzioni che devono essere eseguite dal processore. Ogni istruzione viene sottoposta al ciclo macchina ovvero una serie di passaggi impiegati per l'esecuzione dell'istruzione stessa, di seguito i passaggi appena citati:

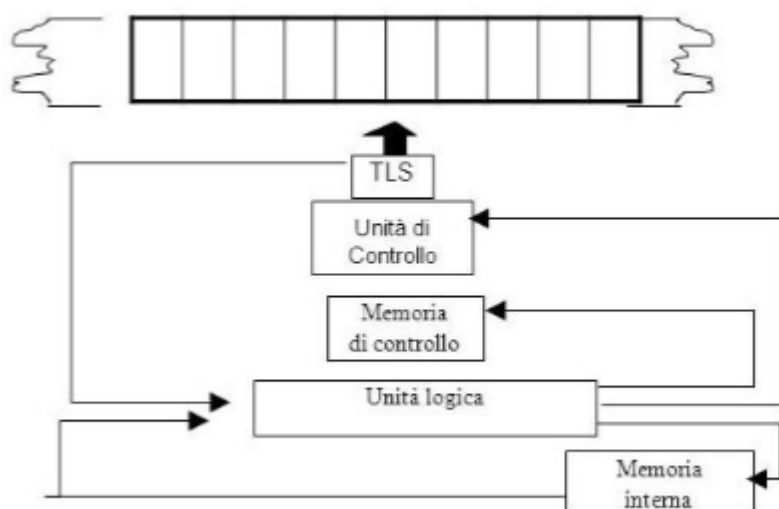
1. **Legge il contenuto del program counter**: ovvero l'indirizzo della prossima istruzione da eseguire.
2. **Caricamento nel registro delle istruzioni (fetch)**: Il processore va a recuperare l'istruzione dalla memoria, utilizzando l'indirizzo letto da program counter. Questa istruzione viene quindi inserita nel **registro delle istruzioni**.
3. **Decodifica dell'istruzione**: Una volta che l'istruzione è stata caricata, il processore la decodifica, per capire di che tipo di istruzione si tratta

4. **Invio all'ALU**: Se l'istruzione richiede un'operazione aritmetica o logica, la **unità logico-aritmetica** (ALU) riceve l'istruzione e i dati necessari per eseguire l'operazione.
5. **Accesso ai dati**: Se l'istruzione da eseguire richiede dei dati la control unit li recupera dalla memoria. Se l'istruzione è del tipo SOM(M1, M2) ad esempio, significa che M1 e M2 sono indirizzi in memoria da cui vengono presi i dati.
6. **Esecuzione**: L'ALU effettua l'operazione richiesta (ad esempio, somma o confronto) utilizzando i dati forniti.
7. **Memorizzazione del risultato**: Una volta ottenuto il risultato, viene registrato nella **locazione di memoria** specificata dall'istruzione.
8. **Aggiornamento del contatore**: Il program counter viene incrementato per puntare all'istruzione successiva.
9. **Ripetizione del ciclo**: Questo ciclo continua fino a quando non viene incontrata un'istruzione speciale che ferma o altera l'esecuzione, come un'istruzione ALT o un salto condizionato che modifica il flusso del programma.

Un altro modello di elaboratore dalla quale Von Neumann prese spunto fu:

macchina di Turing

Una **macchina di Turing (o MdT)** è stata inventata da Alan Turing nel 1936. Questo modello è fondamentale nella teoria della computabilità e fornisce una rappresentazione astratta di come funzionano i calcolatori. Formalmente viene definita in questo modo:



Di seguito una descrizione di tutti i componenti:

- ****Nastro****: Un'unità di memoria esterna infinita, suddivisa in celle, ogni cella contiene un simbolo oppure è vuota.

- ****Testina di lettura/scrittura (TLS)** : ** Un dispositivo che interagisce direttamente con il nastro.
 - ****Unità di memoria interna** : ** Una struttura che memorizza lo stato interno della macchina.
 - ****Unità di calcolo** : ** Un componente che esegue le operazioni di base.
 - ****Unità di controllo** : ** Il "cervello" della macchina, che coordina le altre unità.
 - ****Unità di logica** : ** Un componente che si occupa delle operazioni logiche.
- Il comportamento di una MdT può essere programmata definendo un'insieme di regole, o quintuple di questo tipo:
- (stato-interno-corrente, simbolo-letto, prossimo-stato-interno, simbolo-scritto, direzione)
- di seguito degli esempi:
- (0, A, 1, B, -) se la macchina si trova nello stato 0 e legge il simbolo A passa allo stato 1 e scrive sul nastro B e sta ferma
 - (1, B, 0, A, >) se si trova nello stato 1 e legge il simbolo B passa allo stato 0 e scrive sul nastro A e si muove di una posizione a destra

È importante sottolineare come l'attenzione di Turing sia rivolta al processo di calcolo, **indipendentemente da come esso avviene fisicamente**. Una M.d.T è un dispositivo ideale, cioè indipendente da ogni sua possibile realizzazione fisica. Una funzione si dice **Turing-computabile** se almeno una MdT è in grado di computarla con un numero finito di passi.

Condizioni di finitezza

Una MdT per essere tale deve rispettare le condizioni di finitezza che sono:

- il numero di simboli che usa deve essere fissato e finito
- il numero di caselle del nastro osservabili in una volta è finito
- è possibile ricordare solo un numero finito di stadi precedenti
- le operazioni che può compiere sono:
 1. Cambiare il contenuto di alcune caselle osservate
 2. Cambiare le caselle osservate
 3. Cambiare il proprio stato
 4. Osservare nuove caselle che si trovano ad una distanza prefissata dalla casella osservata

Macchina di Turing universale

Se supponiamo di avere una macchina di Turing senza limiti di spazio, di tempo e che non possa commettere errori quest'ultima sarà in grado di calcolare tutte le funzioni calcolabili in ogni singola macchina di Turing, questa macchina la chiamiamo **Macchina di Turing Universale (MdTU)**, inoltre deve rispettare sia la condizione di finitezza sopracitata ma anche la condizione di determinatezza spiegata di seguito

Condizione di determinatezza

Le azioni di una MdTU devono dipendere solo dai simboli contenuti nella casella osservata in quell'istante e dallo "Stato mentale" corrente, cioè da quello che ricorda dei calcoli precedenti

Da tutto questo Turing formula la seguente Tesi: *"ogni funzione parziale calcolabile con un algoritmo è una funzione parziale calcolabile da una macchina di Turing."* (appunto per questo all'inizio si parla dell'algoritmo come una funzione parziale della macchina di Turing). Questa tesi ci dice che **ogni funzione calcolabile da un algoritmo può essere calcolata da una macchina di Turing**, poiché ogni algoritmo può essere descritto in termini di un insieme finito di regole.

Ulteriori informazioni:

Bottleneck

La CPU è progettata per operare a velocità estremamente elevate, elaborando milioni di istruzioni al secondo. Tuttavia, la RAM, sebbene veloce, ha tempi di accesso più lunghi quando la CPU richiede dati o istruzioni deve attendere che questi vengano recuperati dalla RAM. Questo processo di attesa genera un rallentamento, poiché la CPU rimane inattiva in attesa dei dati necessari per continuare l'elaborazione. Per mitigare questo problema, le CPU fanno uso della cache, una memoria più veloce e più vicina al processore, dove vengono conservati i dati e le istruzioni più frequentemente utilizzati. Tuttavia, anche con l'uso della cache, il bottleneck rimane una preoccupazione, specialmente in scenari di carico elevato o quando vengono eseguiti più processi contemporaneamente. In questi casi, se la RAM non è in grado di tenere il passo con le richieste della CPU, si crea un rallentamento significativo, influenzando negativamente l'efficienza complessiva del sistema.

Il test di Turing

Turing cercò di rispondere alla domanda "can machines think?", per fare ciò formulò quello che viene chiamato test di Turing ovvero un giudice umano comunica con due interlocutori nascosti: uno è un essere umano, e l'altro è una macchina. Se il giudice non riesce a distinguere chi è la macchina e chi è l'umano basandosi solo sulle risposte fornite, allora si dice che la macchina ha **superato il test** e può essere considerata "intelligente". Fino a qualche mese fa nessuno era riuscito a superarlo tranne ChatGPT-4 di recente