

# Informatica

I linguaggi



# Comunicazione

La comunicazione è un processo che consente di scambiare informazioni e coinvolge un mittente, un destinatario, un messaggio ed un canale comunicativo.

Affinché la comunicazione sia efficace è necessario che:

- ❖ il destinatario sia attivo (pronto a ricevere),
- ❖ il canale sia attivo (consenta la trasmissione e, possibilmente, non modifichi il contenuto del messaggio),
- ❖ si usi un codice comunicativo e un protocollo condiviso tra mittente e destinatario.

# Comunicazione

Un protocollo di comunicazione è un insieme di regole che governano lo scambio di messaggi.

Un protocollo di comunicazione definisce, per esempio:

- i segnali di inizio e fine messaggio,
- la lunghezza dei messaggi ed il loro formato.

# Comunicazione

Interazione:

- ❖ Uomo-uomo
- ❖ Uomo-macchina
- ❖ Macchina-macchina

1. Trasmissione dei simboli
2. Trasmissione del significato

# I Linguaggi

- Linguaggio verbale  
Suoni, regole grammaticali
- Linguaggio scritto  
Segni, regole grammaticali
- Gestuale (linguaggi dei segni per audiolesi)  
Gesti, regole
- Ideografico (ideogrammi egizi, spartiti musicali)  
Disegni, regole

# I Linguaggi

$$\boxed{\text{segno}} = \boxed{\text{significante}} + \boxed{\text{significato}}$$

Segno (Qualcosa che sta per qualcos'altro)

Mezzo attraverso il quale si trasmettono messaggi

Significante

Parte fisicamente percepibile (attraverso i sensi) del segno.

Significato

Idea, il concetto che la mente umana collega ad un certo segno

# I Linguaggi



## Esempi:

- Segnali di fumo  
Il fumo (significante), inizia la guerra (significato)
- Linguaggio morse  
... --- ... (significante, scritto o udito),  
*Save Our Souls* , richiesta d'aiuto (significato)
- Il simbolo ☺ o :)

# I Linguaggi

Il rapporto tra significante e significato è frutto di una convenzione e può dipendere dal contesto.

## **Esempi:**

- Il diverso uso dell'apice in matematica e chimica
- Cingomma (Sardegna) o Cicca (Sicilia) per convenzione sociale
- Muovere la testa per annuire



# I Linguaggi

- **Linguaggio naturale**

Si sviluppa spontaneamente in comunità di individui. Può essere soggetto ad ambiguità (più significanti con lo stesso significato)

- **Linguaggio artificiale** (matematico, chimico, di programmazione, etc)

Nasce per accordarsi su come esprimersi su certi ambiti ed è trasversale alle comunità

- ❖ NON ambiguità
- ❖ rigidità sintattica

# Linguaggi o Codici

Sono costituiti da

❖ una serie **finita** di simboli, detta alfabeto. Una sequenza di simboli è detta parola ed una sequenza di parole è detta frase. L'insieme delle parole di un linguaggio è il suo **vocabolario**.

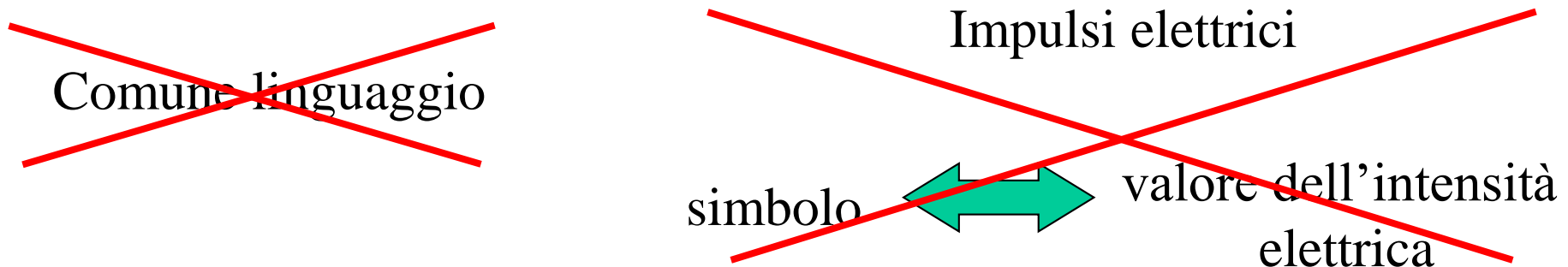
❖ **Sintassi**: insieme di regole che stabiliscono come i simboli possano essere composti a formare le parole e come le parole debbano succedersi per formare le frasi.

❖ **Semantica**: interpretazione del significato da attribuire alla frase e alle parole in esso contenute.

# Codifica

Il processo di traduzione o codifica consiste nella trasformazione delle espressioni linguistiche (sequenze di simboli) di un linguaggio in espressioni linguistiche di un altro linguaggio ed è necessaria ogni volta che i due interlocutori usano linguaggi diversi.

# Come si può “dialogare” con il calcolatore?



## Codifica di tipo binario

0	Non arriva corrente	Non magnetizzato	Riflette la luce
1	Arriva corrente.	Magnetizzato.	Assorbe la luce

# I linguaggi di programmazione

Sono linguaggi artificiali che costituiscono lo strumento per scrivere un algoritmo in modo che sia comprensibile, e quindi eseguibile, da un calcolatore.

Un linguaggio di programmazione dovrebbe essere:

- ❖ Semplice da usare (fornire strutture che permettano una veloce composizione dell'algoritmo)
- ❖ Facile da mantenere, consentendo modularità e parametrizzazione
- ❖ Portabile.

# I linguaggi di basso livello

## Linguaggi macchina

Costituiti da sequenze di numeri binari in cui una parte indica l'istruzione e la restante parte l'indirizzo del dato su cui eseguire l'istruzione. Dipendono dal processore.

Le funzioni di un linguaggio macchina sono:

- ❖ Trasferire istruzioni dalla RAM ai registri e/o alle periferiche
- ❖ Operazioni aritmetiche e logico-relazionali
- ❖ Operazioni di controllo dell'esecuzione delle istruzioni

istruzione	indirizzo
+1300042774	
+1400593419	
+1200274027	

# I linguaggi di basso livello

## Linguaggi assembly

Legati all'hardware.

Codici per le istruzioni (espressioni simili al linguaggio naturale).

Indirizzamento simbolico (uso delle variabili).

Load basepay

Add overpay

Store grosspay

**Assemblatore:** traduttore del linguaggio assembly in linguaggio macchina

# I linguaggi di alto livello

- ❖ Consentono di concentrarsi sull'algoritmo più che sulle caratteristiche del linguaggio
- ❖ Vocabolario più simile al linguaggio naturale.
- ❖ Programmi più portabili.

`grosspay=basepay+overpay`



# Alcuni linguaggi di programmazione

FORTRAN = FORmula TRANslator

COBOL = Common Business Oriented Language

PASCAL → Nato per scopi didattici.  
Programmazione strutturata

ADA → multitasking = parallelizzazione delle attività

# I linguaggi interpretati

Vengono tradotti da un apposito programma detto **interprete** durante la stesura del codice.

L'interprete riceve in ingresso dati e istruzioni.

Meno efficienti in fase di esecuzione ma più efficienti in fase di debugging.

# I linguaggi compilati

Al termine della scrittura, vengono tradotti da un apposito programma detto **compilatore** che produce un file detto programma oggetto ed un file eseguibile. Esso può essere eseguito varie volte, con dati diversi, ed esportato su supporti di memoria o su altre macchine.

Il compilatore riceve le istruzioni e le traduce. I dati vengono inseriti in fase di esecuzione.

Sono più efficienti in fase di esecuzione ma non in fase di debugging.

# Compilazione di un programma

Dopo la costruzione dell'algoritmo, si effettua la codifica in linguaggio di programmazione utilizzando un editor di testi. Il prodotto di questa fase è un file di testo chiamato 'sorgente'.

# Compilazione di un programma

Il file sorgente viene passato al compilatore che lo traduce in linguaggio macchina attivando le seguenti componenti:

- Preprocessore: sostituisce parti di codice (costanti, librerie, macros, etc.)
- Linker: inserisce i riferimenti a porzioni esterne di codice (altri programmi, funzioni di libreria, funzioni del S.O.)
- Debugger: eseguendo un'analisi lessicale e sintattica, per identificare eventuali errori di sintassi.

Se questa fase va a buon fine viene creato il file 'oggetto'.

# Compilazione di un programma

Il file oggetto viene finalmente tradotto in linguaggio macchina ed è pronto per essere caricato in RAM dal loader e poi eseguito.

Il risultato si chiama file eseguibile ed è specifico per il sistema operativo che lo ospiterà.

