
INTRODUZIONE

INFORMATICA

Scienza che studia la **rappresentazione** e l'**elaborazione** delle informazioni mediante **macchine**, dette calcolatori elettronici.

In inglese: ***Computer Science***

Macchine a **funzionalità intrinseca**, o «cablata»:

sono macchine (elettroniche, ma anche meccaniche, elettriche, ecc.) il cui comportamento è determinato dalla loro **struttura** costruttiva.

Approccio **efficiente** ma rigido.

Macchine a **funzionalità programmata**:

sono macchine il cui comportamento è determinato **esecuzione di programmi**, cioè descrizioni di soluzioni di problemi.

Approccio meno efficiente ma **molto flessibile**.

La massima flessibilità di impiego delle macchine programmabili si ha quando in esse è possibile inserire, a scelta, diversi programmi.

Questa flessibilità è la causa principale della notevole diffusione dei calcolatori e dello sviluppo dell'informatica.

In questo corso ci interessiamo quindi di macchine a funzionalità programmata.

Problemi da risolvere

I problemi che siamo interessati a risolvere sono di natura *molto varia*:

- 1) Trovare il maggiore fra due numeri
- 2) Dato un elenco di nomi e numeri di telefono, trovare il numero di una data persona
- 3) Problema del lupo, della capra e del cavolo
- 4) Dati a e b , risolvere l'equazione $ax+b=0$
- 5) Stabilire se una parola precede alfabeticamente un'altra
- 6) Ordinare una lista di elementi

- 7) Identificare e prenotare aerei, treni, hotel, ...
- 8) Creare, modificare e alterare suoni, canzoni, ...
- 9) Analizzare e riconoscere immagini, ...
- 10) Salvare e recuperare delle informazioni, ...
- 11) Trasmettere delle informazioni, ...

E' fondamentale capire la differenza tra...

- **Specifica di un problema**
- **Specifica del processo di risoluzione**
- **Codifica del processo di risoluzione**

Risoluzione di un problema

Con questo termine si indica il processo che:

- dato un *problema*
- individuato un opportuno *metodo risolutivo*

trasforma i dati iniziali nei corrispondenti risultati finali

E' indispensabile capire a quale "macchina" ci si riferisce"

COMPUTER



Computer

- E' uno strumento in grado di eseguire insiemi di *azioni* (“mosse”) elementari
- Le azioni vengono eseguite su oggetti (***dati***) per produrre altri oggetti (***risultati***)
- L'esecuzione di azioni viene richiesta all'elaboratore attraverso “**frasi scritte in un qualche linguaggio**” (***istruzioni***)

Componenti di un Sistema di Elaborazione

- **Hardware**
- **Software**
 - Software di sistema: Sistema operativo
 - Software applicativo

Domande fondamentali

- **Come viene rappresentata TUTTA l'informazione all'interno del computer?**
- **Quali istruzioni esegue un computer?**
- **Quali problemi può risolvere un computer?**
- **Esistono problemi che un computer non può risolvere?**

Algoritmo

ALGORITMO

- Sequenza finita di mosse che risolve in un tempo finito una classe di problemi

CODIFICA o IMPLEMENTAZIONE

- Fase di scrittura di un algoritmo attraverso un insieme ordinato di **frasi** (“**istruzioni**”), scritte in un **linguaggio di programmazione**, che specificano le **azioni** da compiere in modo formale interpretabile dal computer

Programma

- **Testo** scritto secondo la *sintassi* (alfabeto+regole grammaticali) e la *semantica* di un linguaggio di programmazione
- **Un programma può non essere un algoritmo!**

PROGRAMMAZIONE

È l'attività con cui si predispose l'elaboratore ad eseguire un *particolare insieme di azioni* su *particolari dati*, allo scopo di *risolvere un problema*.

ALGORITMO e PROGRAMMA (a confronto)

- Ogni elaboratore è una macchina in grado di eseguire *azioni* elementari su *dati*
- L'esecuzione delle azioni elementari è richiesta all'elaboratore tramite comandi chiamati *istruzioni*
- Le istruzioni sono espresse attraverso *frasi* di un opportuno *linguaggio di programmazione*
- **Un *programma* non è altro che la formulazione testuale di un *algoritmo* in un linguaggio di programmazione**

Elementi degli Algoritmi

In generale negli algoritmi si possono evidenziare i seguenti aspetti:

- **oggetti -> i dati**
- **operazioni -> le operazioni che modificano i dati**
- **flusso di controllo -> in che ordine si opera**

Oggetti. Costituiscono le entità su cui opera l'algoritmo.

In genere si tratta di dati iniziali del problema, informazioni ausiliarie e risultati parziali e finali. Si noti che in informatica le informazioni sono generalmente dette **dati**, anche se costituiscono risultati parziali o finali di elaborazioni. Possono essere **variabili** o **costanti** di vari tipi.

Operazioni. Sono gli interventi da effettuare sugli oggetti, cioè sui dati.

In genere si tratta di calcoli, confronti, ricopiature, acquisizioni emissioni, ecc.

Flusso di controllo. Costituisce l'indicazione delle possibili evoluzioni dell'esecuzione delle operazioni, cioè le possibili successioni dei passi

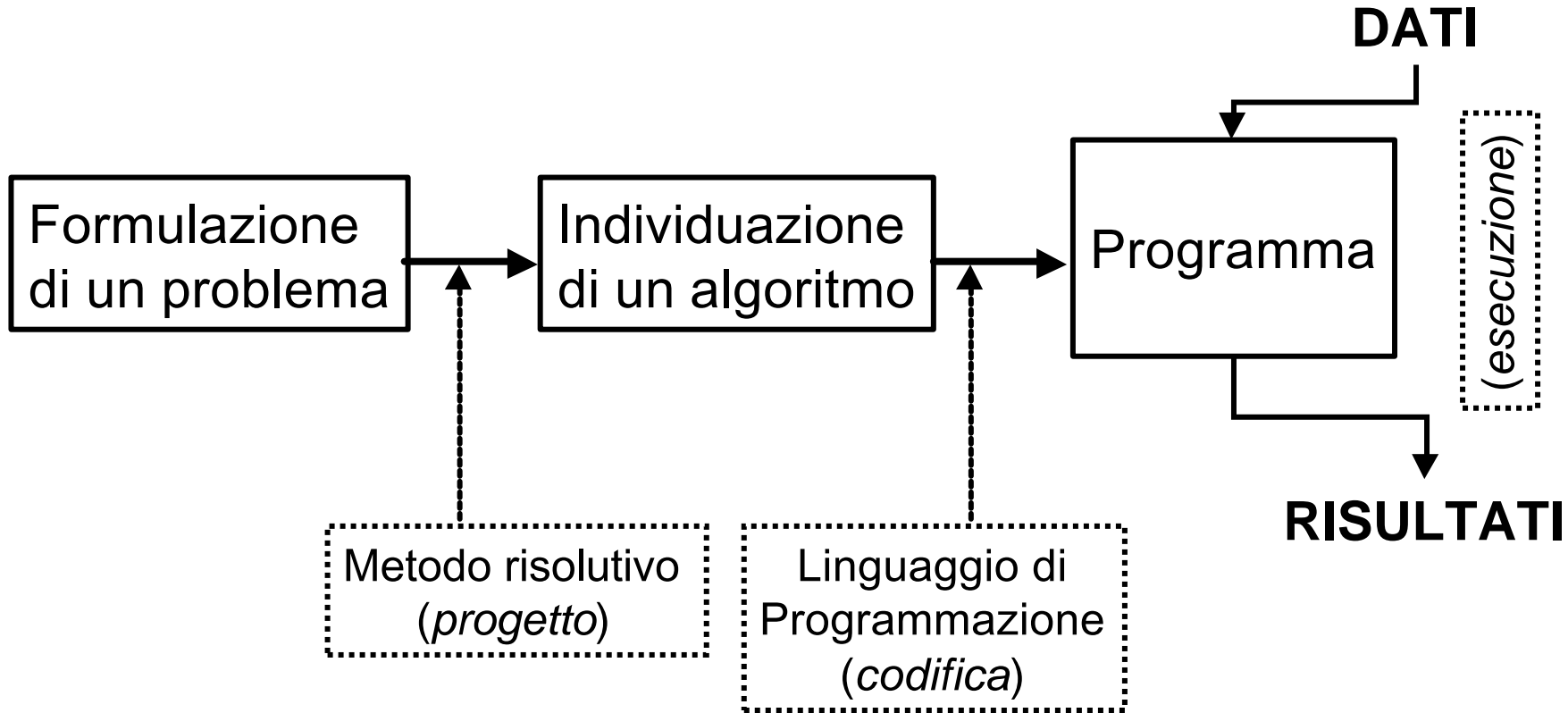
E' importante notare che la correttezza dei risultati dipende non solo dalla corretta esecuzione delle singole operazioni, ma anche dalla corretta sequenza con cui sono eseguite.

NOTA: è importante chiarire la differenza tra

Flusso di controllo e flusso di esecuzione.

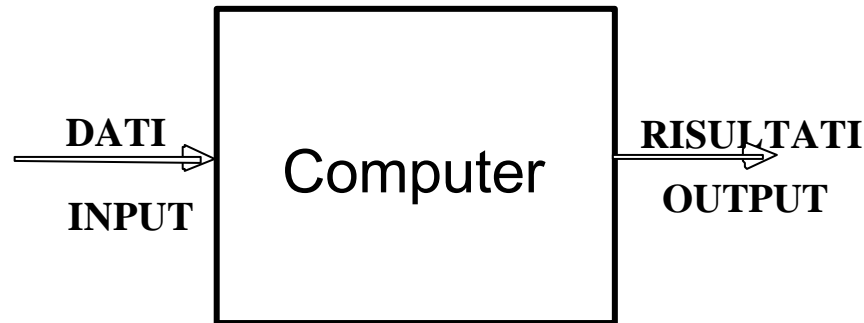
- Il **flusso di controllo** è la descrizione a priori di tutte le possibili sequenze nell'esecuzione dei passi dell'algoritmo, in particolare di operazioni in alternativa e di operazioni da ripetere più volte ciclicamente.
- Il **flusso di esecuzione** è la sequenza di operazioni effettivamente seguita durante una particolare esecuzione dell'algoritmo e che dipende dai particolari valori che i dati assumono in quella esecuzione.

Riassumendo...



Esecuzione di un programma

- L'esecuzione delle azioni *nell'ordine specificato dall' algoritmo* consente di ottenere, a partire dai dati di ingresso, i risultati che risolvono il problema



Esempio 1

Descrizione del problema

Stampare la somma di due numeri dati

Individuazione di un algoritmo

- Leggere il primo numero (p.es., da tastiera)
- Leggere il secondo numero (p.es., da tastiera)
- Effettuare la somma
- Stampare il risultato (p.es., su video)

Esempio 1 (*cont.*)

Codifica in un programma

```
main()  
{  
    int A, B, ris;  
    stampa("Immettere due numeri: ");  
    leggi("decimale", &A);  
    leggi("decimale", &B);  
    ris=A+B;  
    stampa("decimale", ris);  
}
```

Caratteristiche di un algoritmo

- **Eseguibilità:** ogni azione deve essere *eseguibile* da parte dell'esecutore dell'algoritmo in un tempo finito
- **Non-ambiguità:** ogni azione deve essere *univocamente interpretabile* dall'esecutore
- **Terminazione:** il numero totale di azioni da eseguire, per ogni insieme di dati di ingresso, deve essere finito

Algoritmo (*cont.*)

Quindi, l'algoritmo deve:

- essere *applicabile a qualsiasi insieme di dati di ingresso appartenenti al dominio di definizione dell'algoritmo*
- essere costituito da operazioni appartenenti ad un determinato insieme di operazioni fondamentali
- essere costituito da regole non ambigue, cioè interpretabili in modo univoco qualunque sia l'esecutore (persona o "macchina") che le legge

ALTRE PROPRIETA'

- **Determinismo**
- **Efficienza**
- **Terminazione**

Algoritmi equivalenti

Due algoritmi si dicono **equivalenti** quando:

- hanno lo stesso dominio di ingresso
- hanno lo stesso dominio di uscita
- in corrispondenza degli stessi valori nel dominio di ingresso *producono gli stessi valori* nel dominio di uscita

Due algoritmi equivalenti:

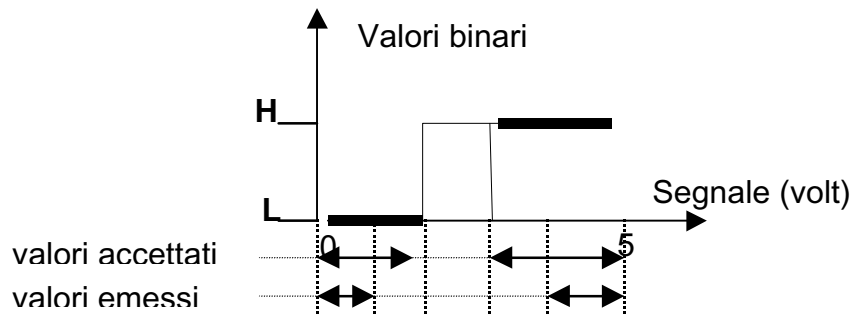
- forniscono lo **stesso risultato**
- ma possono avere **diversa efficienza**
- e possono essere **profondamente diversi !**

Linguaggio macchina

- E' il linguaggio di un computer
- Il linguaggio macchina è direttamente eseguibile dall'elaboratore, senza alcuna intermediazione
- Computer con architetture interne (CPU) differenti hanno linguaggi macchina differenti
- Pertanto, un programma scritto nel linguaggio macchina di un computer non è eseguibile su di un computer con un'architettura differente
(Si dice che non è portabile!)

Rappresentazione delle informazioni con segnali digitali binari

La grandezza fisica che si utilizza (segnale elettrico di tensione) assume solo **due valori discreti** (binaria)



L'elemento tecnologico base per la realizzazione di circuiti digitali è il **transistore** il cui funzionamento può essere modellato (in modo molto semplice) come il funzionamento di un interruttore (*aperto* o *chiuso*), quindi con due stati fisici, cui corrispondono 2 opportune tensioni (in genere 0V e 5V)..

BIT (*binary digit*) = cifra binaria. (unità di informazione elementare)

Un bit può assumere due valori che possono essere associati ai simboli:

L(ow)	H(igh)	<i>aspetto fisico del segnale</i>
0	1	<i>aspetto aritmetico</i>
false	true	<i>aspetto logico</i>

Terminologia e «unita' di misura»

1 cifra	= bit	1 Kilo	= $2^{10} = 1024$	$\cong 10^3$
8 bit	= byte	1 Mega	= $2^{20} = 1048576$	$\cong 10^6$
16 bit	= word (parola)	1 Giga	= 2^{30}	$\cong 10^9$
32 bit	= double word			
64 bit	= quad word			

Linguaggi di alto livello

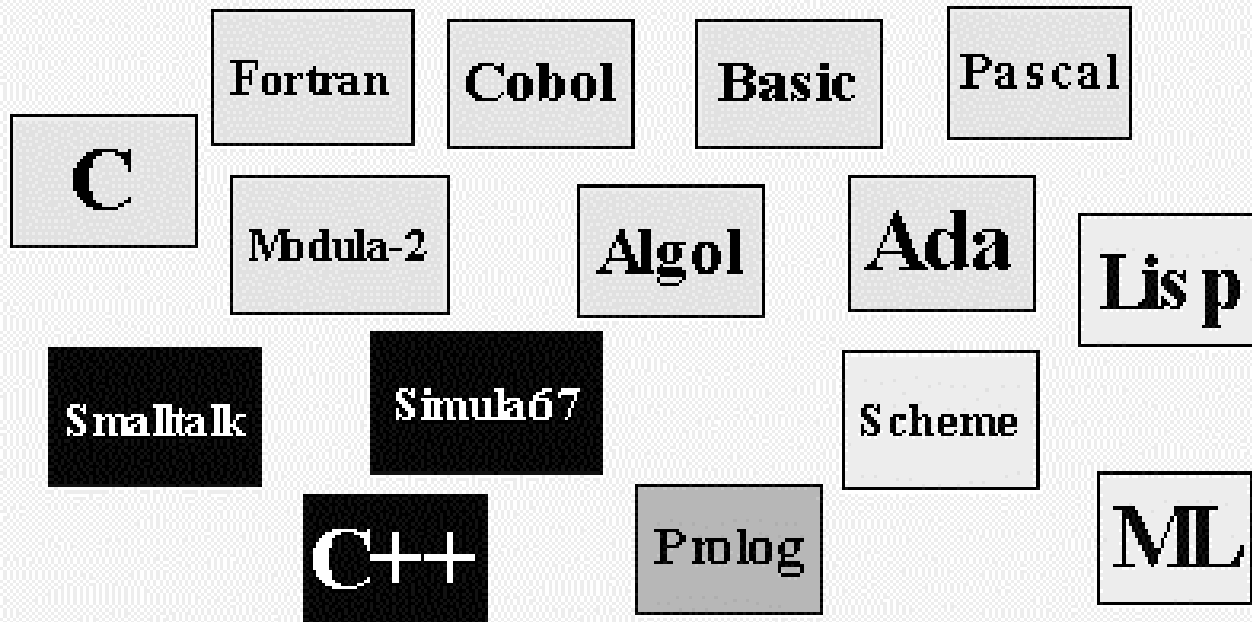
- Si basano su una “macchina” le cui “mosse” non sono quelle della macchina hardware
→ realizzano una “**macchina virtuale**”
- Supportano concetti ed astrazioni
- Promuovono metodologie per agevolare lo sviluppo del software da parte del programmatore
- Hanno capacità espressive molto superiori rispetto a quelle del linguaggio macchina
- **Esistono centinaia di linguaggi di programmazione!**
(anche se pochi sono in uso)

Linguaggi di alto livello (*cont.*)

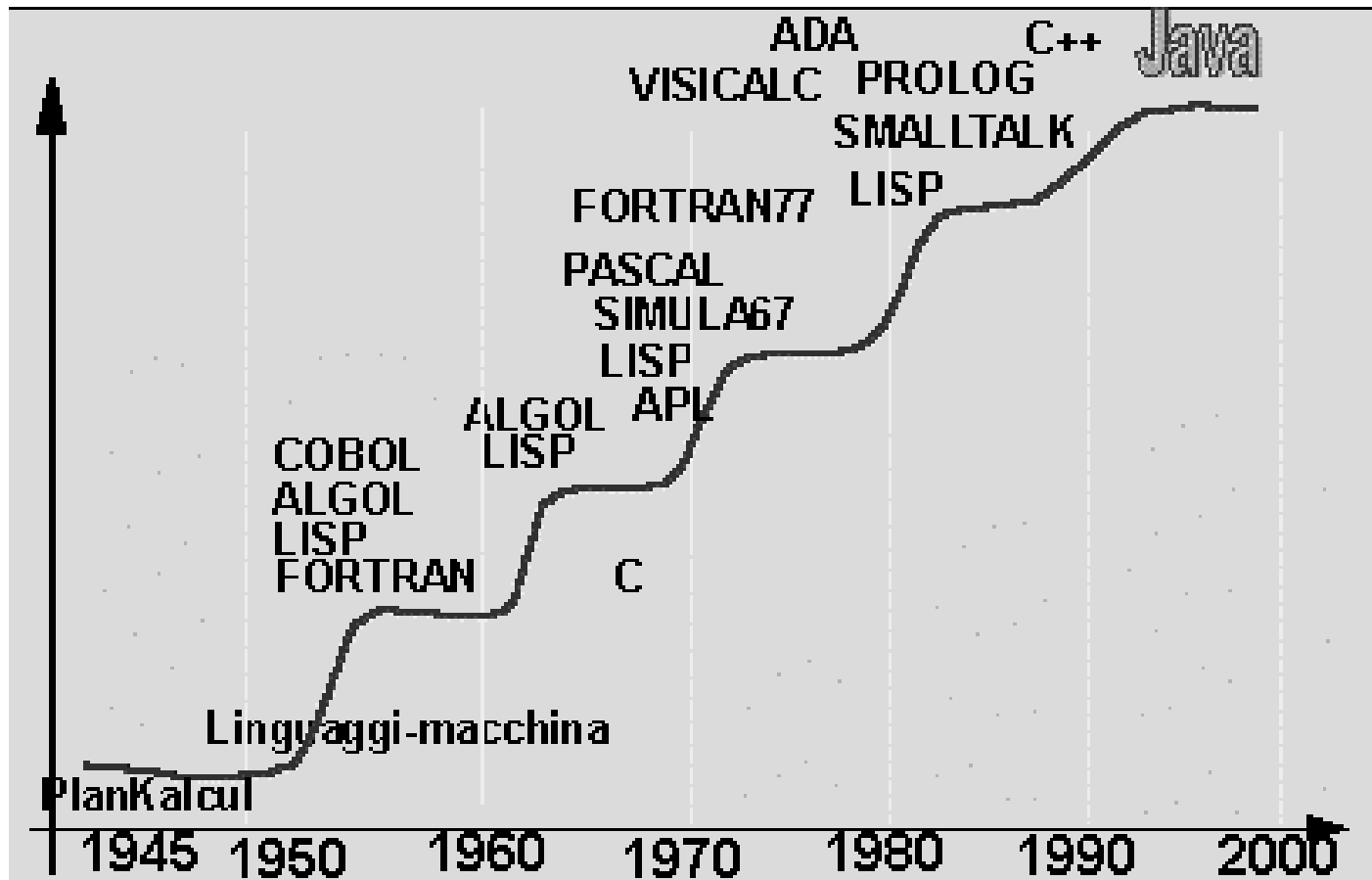
Linguaggi di alto livello



Barriera di astrazione

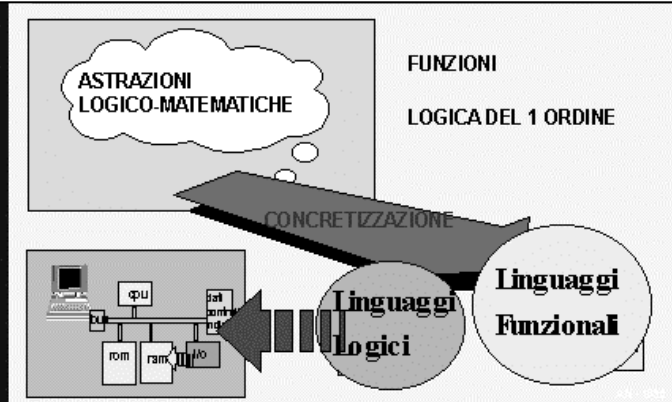
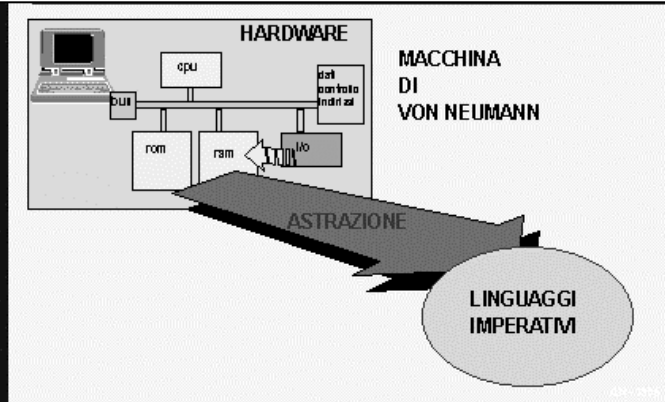


Evoluzione dei linguaggi



Perché esistono tanti linguaggi?

- **Contesto applicativo:**
 - Scientifico: **Fortran**
 - Gestionale: **Cobol**
 - Sistemi Operativi: **C**
 - Applicazioni di rete: **Java**
- **Modello di partenza:**



Ambiente di programmazione

- È l'insieme degli strumenti (*tool*) che consentono la codifica, la verifica e l'esecuzione di nuovi programmi (*fasi di sviluppo*)

Sviluppo di un Programma

- Affinché un programma scritto in un qualsiasi linguaggio di programmazione ad alto livello sia comprensibile (e quindi eseguibile) da un calcolatore, **occorre tradurlo dal linguaggio di programmazione originario al linguaggio comprensibile al calcolatore (linguaggio macchina)**
- Questa operazione viene normalmente svolta da speciali programmi, detti **traduttori**

Processo di traduzione

Programma

```
main( )  
{ int A;  
  ...  
  A=A+1;  
  if ...
```

Traduzione

```
000001  
  ...  
11001..  
1011100 ...
```

I traduttori convertono il testo dei programmi scritti in un particolare linguaggio di programmazione (***programmi sorgenti***) nella corrispondente rappresentazione in linguaggio macchina (***programmi eseguibili***)