UNIVERSITÀ DEGLI STUDI DELLA CAMPANIA
LUIGI VANVITELLI
SCUOLA POLITECNICA E DELLE SCIENZE DI BASE
DIPARTIMENTO DI INGEGNERIA
INDUSTRIALE E DELL'INFORMAZIONE

Introduzione al corso

Prof. Salvatore Venticinque
Prof. Massimiliano Rak
Ing. Salvatore D'Angelo

Informazioni utili

Prof. Salvatore Venticinque

Prof. Massimiliano Rak

Ing. Salvatore D'Angelo

- Contatti:
  Dipartimento di Ingegneria Industriale e dell'Informazione *(DIII)*
  mailto: salvatore.venticinque@unina2.it
  massimiliano.rak@unina2.it
  salvatore.dangelo@unina2.it

http://elearning.unina2.it

Per le comunicazioni: forum del corso (collegato a email ufficiale e rss-feeds)

- Ricevimento studenti:
  Martedì *9:30/11:00* Prof. Salvatore Venticinque

Composizione del corso

Elementi di Programmazione (9CFU) - ING-INF/05

- Argomenti teorici di base

  - oggetto delle lezioni in aula

  - verificati tramite quesiti ed esercizi della prova scritta

- Programmazione in linguaggio C

  - oggetto delle lezioni e delle esercitazioni in aula

  - verificata tramite problemi della prova scritta

Modalità d'esame

- **Prova scritta.**

  - Composta di una parte teorica e di una applicativa

  - Obbligatoria per il superamento dell'esame

- **Prova orale.**

Libri di testo

Testo di riferimento:

Bellini, Guidi. Linguaggio C. *Mc-Graw-Hill,*

Altri testi consigliati:

- Kernighan, Ritchie – "Il linguaggio C", Pearson
- Pasquale Foggia, Mario Vento - "Algoritmi e strutture dati", McGraw Hill

Programma

- Lezioni frontali (9 crediti, 72 ore):
  - Algoritmi e Programmazione Strutturata
  - Programmazione codifica in linguaggio C

- Esercitazioni in aula con portatile personale
  - Progettazione di Algoritmi
  - Codifica di Algoritmi in Linguaggio C
  - Progettazione e Codifica Algoritmi di Ordinamento e Ricerca
  - Uso di funzioni e procedure e gestione dei file
  - Utilizzo degli ambienti di sviluppo

Il programma completo è già disponibile sul sito del corso.

Cosa Impariamo in questo Corso

Costruzione di
un programma

In C

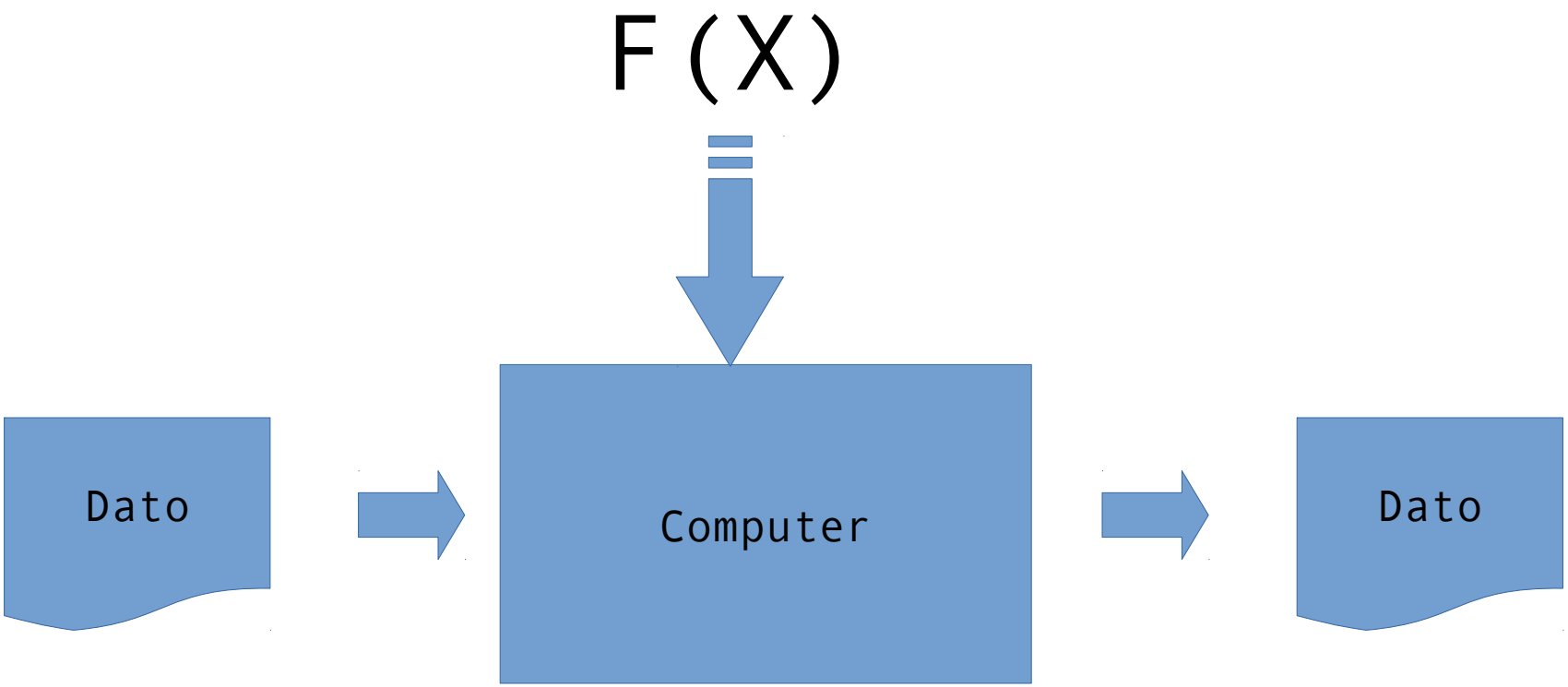
Informatica

Elaborazione automatica dell'informazione

Cosa serve per l'elaborazione automatica?

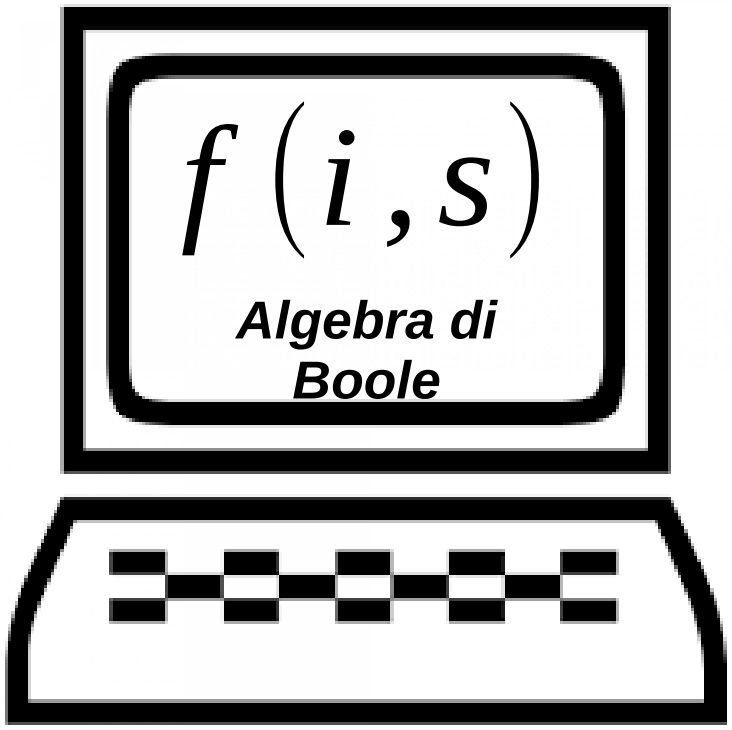
un sistema digitale a programma registrabile per l'elaborazione automatica dell'informazione

L'eLaboratore



L'elaboratore

I: input



O: output

Informatica

- Definizione

  Scienza della rappresentazione e dell'elaborazione dell'*informazione*

- Definizione Association of Computing Machinery(ACM)

  Studio sistematico degli algoritmi(sequenze precise di operazioni comprensibili e perciò eseguibili da uno strumento automatico) che descrivono e trasformano l'informazione: la loro teoria, analisi, progetto, efficienza, realizzazione e applicazione

Sistema Informatico

- Oggetto complesso,che può assumere nature molto differenti, costituito da molte parti che interagiscono tra loro per eseguire algoritmi
- Classificazione a livello generico delle componenti:
  - **Hardware**: componenti fisici del sistema
  - **Software**: programmi eseguiti dal sistema

Il confine tra HW e SW è piuttosto sfumato se si pensa che le stesse funzioni possono essere svolte a seconda dei casi da circuiti e dispositivi HW o da particolari *microprogrammi(**firmware**)* definiti dal costruttore del calcolatore

Hardware

Insieme di elementi funzionali:

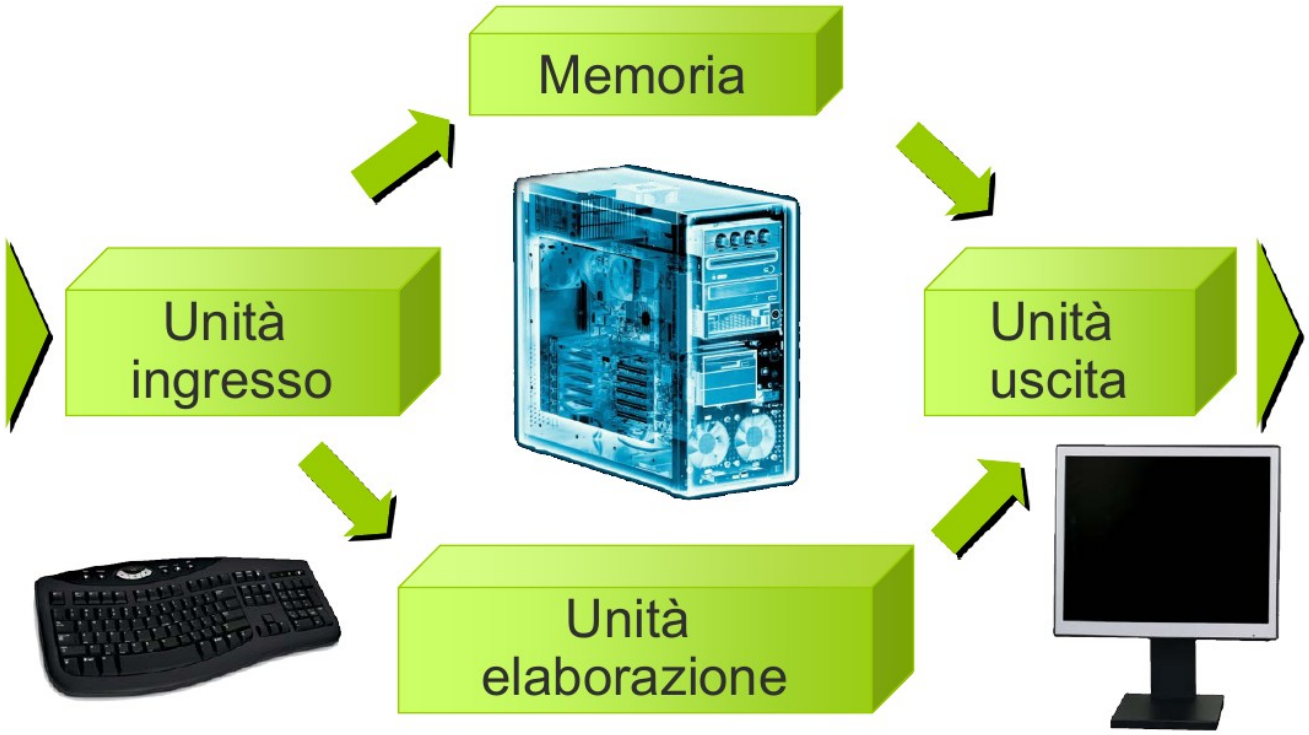
- **Unità di elaborazione o processore**:
  - esegue i programmi
- **Memoria centrale**:
  - memorizza dati e programmi per il funzionamento dell'elaboratore
  - capacità limitata
  - volatile
  - rapido acceso all'informazione
- **Memoria secondaria** (o di massa):
  - capacità significativa
  - persistente
  - accesso all'informazione lento
- **Bus di sistema**:
  - collega gli elementi funzionali consentendo lo scambio di dati
- **Unità periferiche:**
  - fanno comunicare il calcolatore con l'ambiente esterno

Esempio di HW di sistema informatico

Personal computer: elaboratore dedicato ad un solo utente

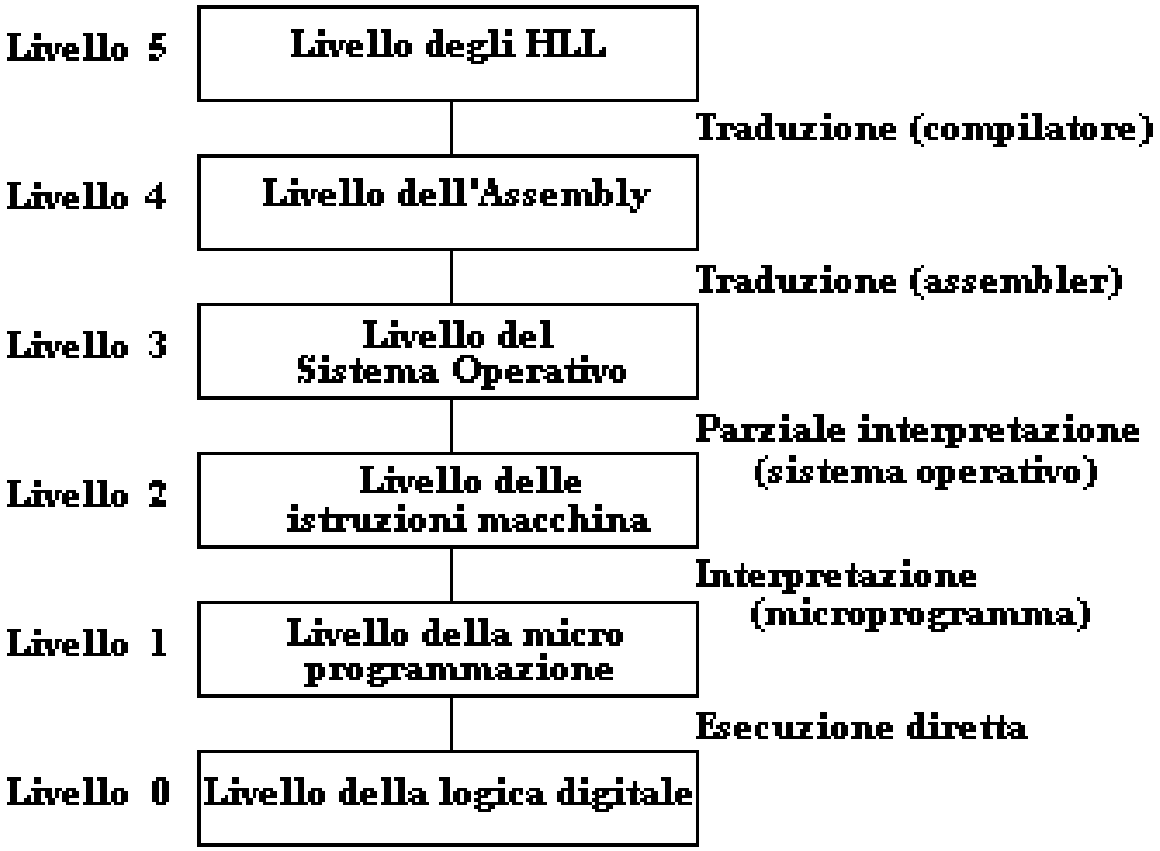
- Corpo contenente:
  - unità di elaborazione
  - memoria centrale
  - memoria di massa: informazione organizzata in file
    - Disco fisso(hard disk): inamovibile e di elevata capacità
    - Floppy disk, chiavi USB, dischi ottici(compact disk o CD-ROM o DVD)
- Tastiera mouse e video collegati col corpo centrale

Il Modello di Von Neumann




Memoria centrale (RAM)
Memoria
Memoria di massa (dischi)
Unità ingresso
Unità uscita
Unità elaborazione

Livelli di un elaboratore (Tanebaum)



Software

- SW di base
  - dedicato alla gestione dell'elaboratore
  - opera direttamente al di sopra di HW e firmware
  - Esempi:
    - SISTEMA OPERATIVO
    - Sistema di gestione di basi di dati
    - Protocolli di comunicazione:
      garantiscono la corretta comunicazione sui canali di trasmissione che trasportano segnali(tipicamente elettrici), garantendo la trasmissione dei dati tra elaboratore e terminali o tra elaboratori collegati in rete

- SW applicativo:
  - dedicato alla realizzazione di specifiche esigenze applicative
  - utilizza linguaggi di alto livello
  - opera al di sopra del SW di base
  - non risente delle caratteristiche architetturali del sistema informatico:trasportabile

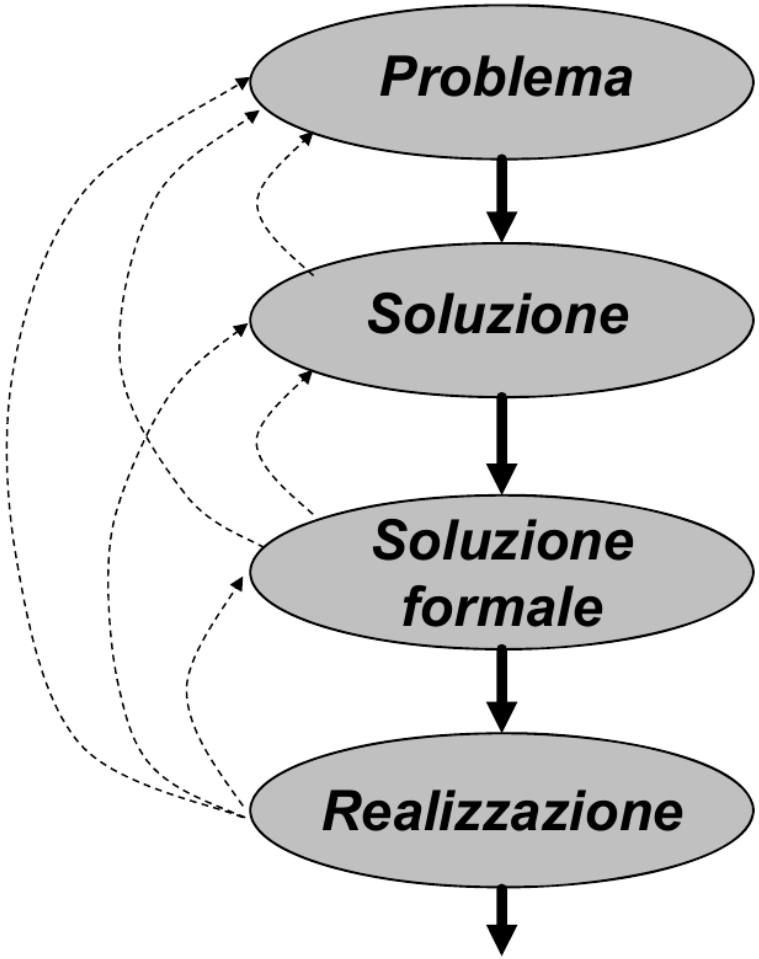
Sistema Operativo

- Funzioni:
  - Interpreta ed esegue comandi elementari
  - Organizza le risorse della macchina
  - Gestisce l'accesso alla rete
- Sia che sia venduto dal costruttore del sistema informatico che realizzato da ditte di SW, il SO non può essere modificato dall'utente nelle sue istruzioni ed è necessario come tramite d'uso tra la macchina fisica e l'utente
- Complessità crescente con quella del sistema informatico:
  - In sistemi multi-utente il SO distribuisce le parti del calcolatore tra i vari utenti in maniera apparentemente "dedicata"

Cosa vuol dire programmare

- Scrivere un programma significa adattare la logica funzionale della macchina alle esigenze operative di un certo problema, secondo uno svolgimento sequenziale.
- Le esigenze operative del problema scaturiscono dalla logica risolutiva idonea al conseguimento dei risultati voluti.
- La logica funzionale di macchina è l'insieme di operazioni elementari che questa è in grado di svolgere.
- Per svolgimento sequenziale si intende la successione nel tempo delle operazioni elementari.

Progettazione



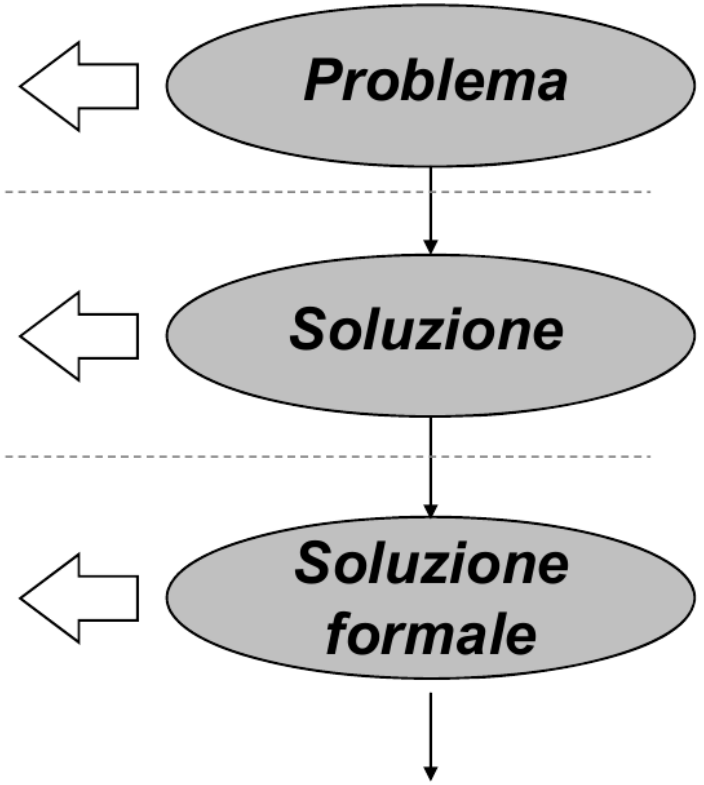
Difficoltà

- I punti critici nello sviluppo di un progetto risiedono essenzialmente in:
  - Sviluppo di una soluzione “informale”
- Formalizzazione di una soluzione
  - Permette una più semplice “traduzione” nelle regole di realizzazione
  - La soluzione di un problema passa generalmente attraverso lo sviluppo di un algoritmo

Esempio di Progettazione

- ***Problema:*** Calcolo del massimo tra due valori A e B
- ***Soluzione:*** Il massimo è il più grande tra A e B...
- ***Soluzione formale:***

1. inizialmente: max = 0
2. se A >B allora max = A; stop
3. altrimenti max = B; stop



Realizzazione

Codifica della soluzione utilizzando:

un modello eseguibile da un calcolatore

Ovvero

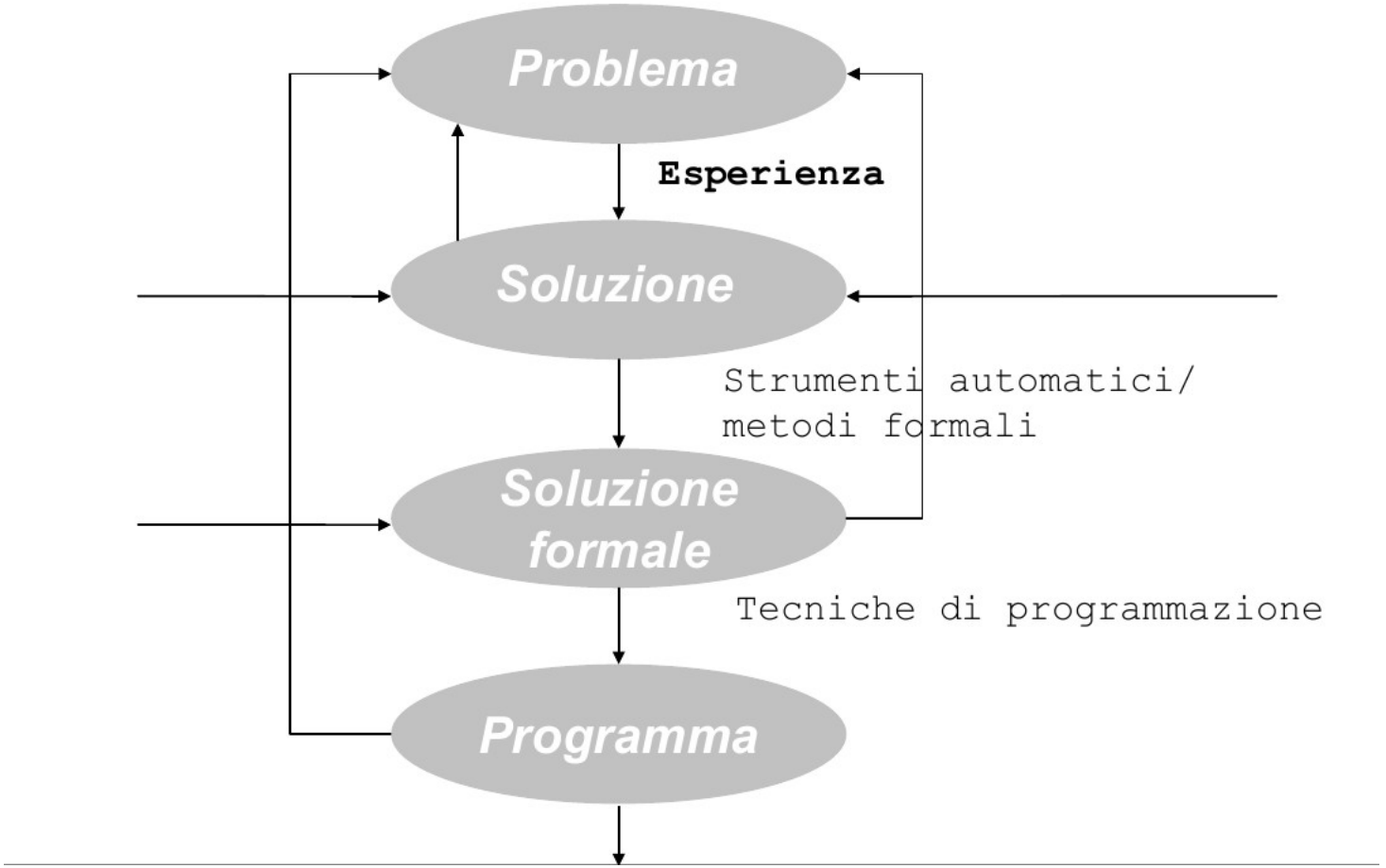
Codifica della soluzione attraverso:

un ***linguaggio di programmazione***

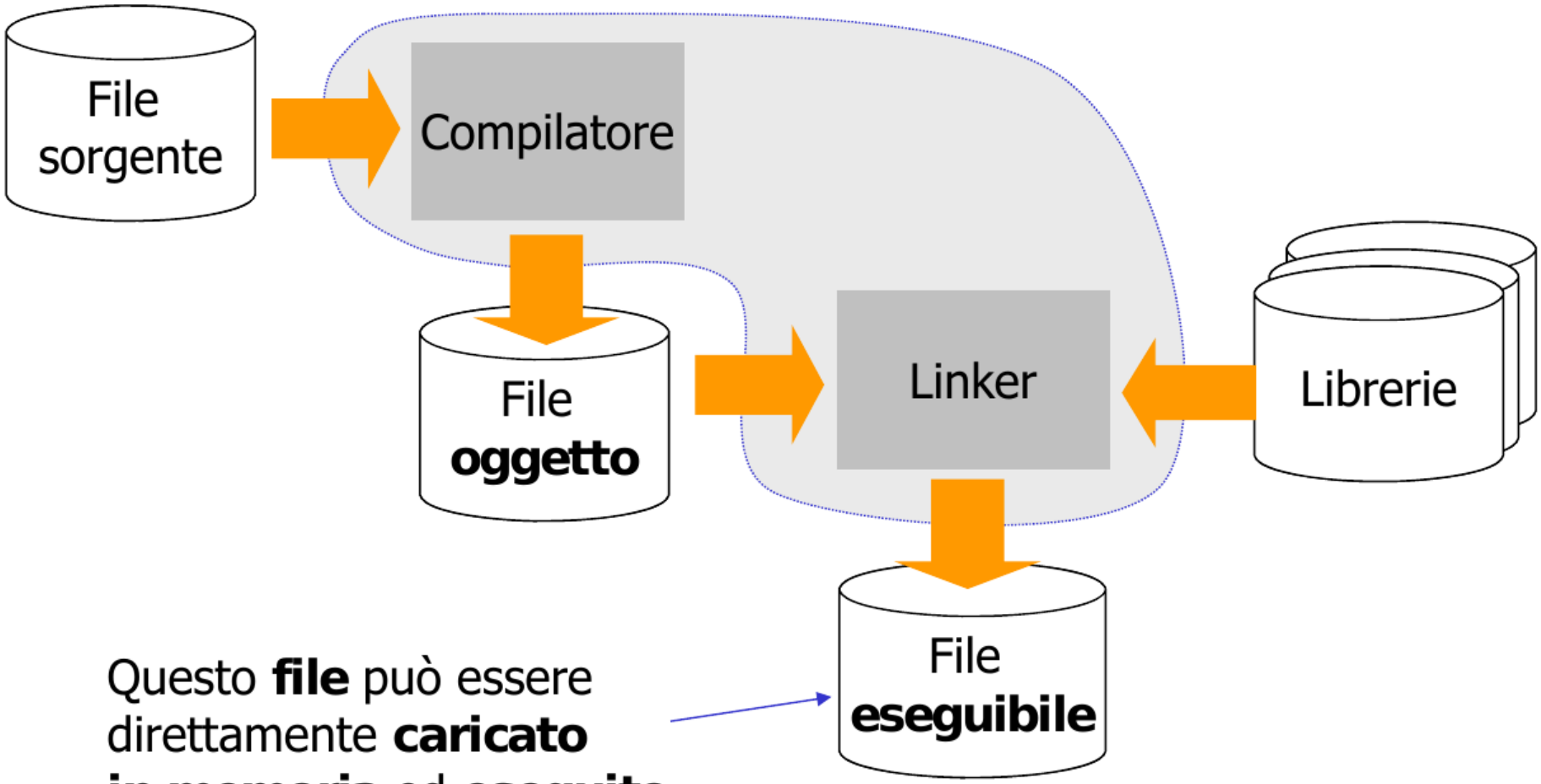
Il risultato dello sviluppo

- *un* ***software:*** *un programma* ***Eseguibile***

Progettazione Rivista



Ciclo di Sviluppo del Software



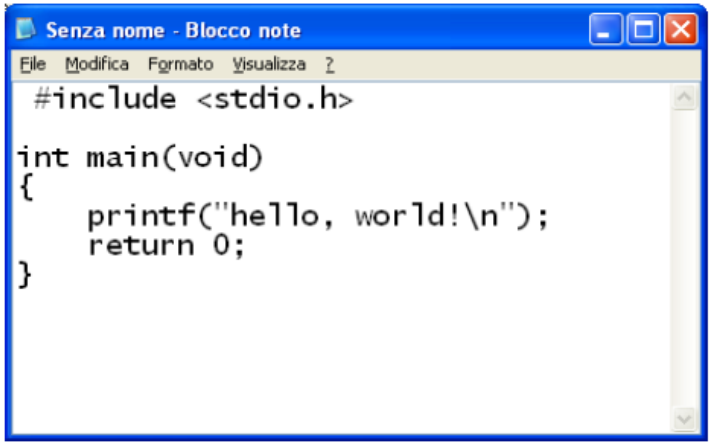
Linguaggi di Programmazione

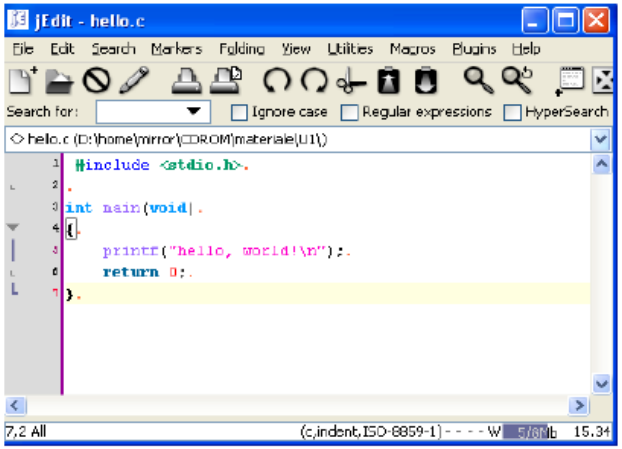
- Linguaggi per la codifica di algoritmi: scrittura sotto forma di programmi che possano essere compresi da un elaboratore
- Dal linguaggio della macchina ai linguaggi di alto livello: sforzo di traduzione da linguaggio naturale a linguaggio macchina sempre più affidato alla macchina stessa

Scrivere un programma

Strumento : Editor di testo

- Notepad
- Notepad++
- Geany





Output : file di testo

- mioprogramma.c

Il pogramma C

```
#include <stdio.h>

int main()
{
        printf("hallo world!!");
}
```

Compilare il programma

Strumento: Compilatore (un software)

- Intel compiler
- GNU compiler
- Mingw compiler

Output: un file oggetto

- Il risultato è la traduzione del sorgente in linguaggio macchina (myoprogramma.o)

Differenza tra linguaggio assembler e linguaggio macchina ???

Ottenere un linguaggio assembler?

Linking

Input: file(s) oggetto

Strumento: Compilatore

Output: File eseguibile

Mioprogramma.exe

Test

Input: file(s) Dati di Test

Strumento: Debugger

Output: ***Risultato Corretto?***