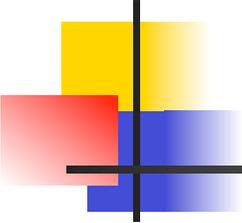




Informatica Generale





Scopi del corso

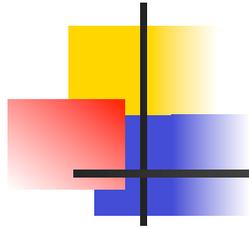


- Aspetti **fondazionali**
 - Cos'è un **elaboratore**
 - Cos'è un **linguaggio di programmazione**
 - Cos'è un **algoritmo**
- Aspetti **pratici**
 - **Compilazione**
 - **programmazione**



Parte I: Hardware

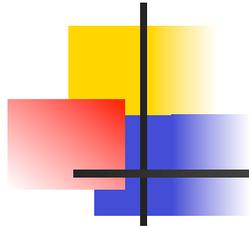
- Codifica dell'informazione
- Architettura dei sistemi informatici
 - Struttura dell'elaboratore
 - Linguaggio macchina
- Sistemi operativi
 - Gestione dei processi e della memoria
 - Come usare un sistema operativo



Parte II: Introduzione alla Programmazione

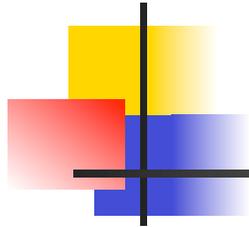
- Compilazione ed esecuzione
- Costrutti di programmazione
- Strutture dati semplici
- Metodologia di programmazione





Introduzione all'informatica





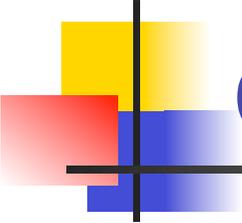
Cos'è l'informatica?

- Scienza della ***rappresentazione*** e dell'***elaborazione dell'informazione***
ovvero
- Studio degli ***algoritmi*** che ***descrivono*** e ***trasformano*** l'informazione



Nozione di Algoritmo

- Sequenza di passi per risolvere un determinato problema
- Calcolatore = Esecutore di algoritmi
- Gli algoritmi sono descritti tramite programmi scritti in linguaggi ad ***alto livello*** e poi tradotti in ***linguaggio macchina***



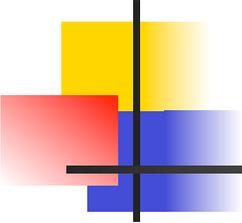
Criteri di valutazione

- *Correttezza*

- l'algoritmo risolve il problema in modo completo (spesso occorre provare la correttezza manualmente usando tecniche matematiche)

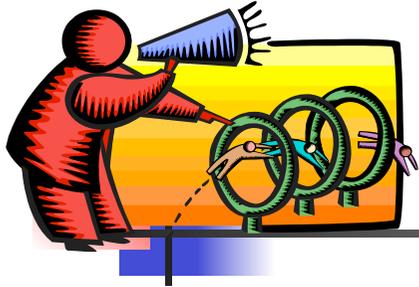
- *Efficienza*

- lo risolve nel modo più veloce possibile (esistono criteri matematici di valutazione)



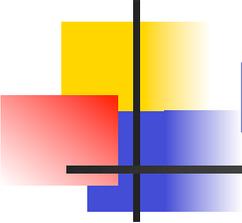
Esempio: elevamento a potenza

- *Problema:* Calcolare a elevato alla n
 - Utilizziamo le variabili N Ris
 - Inizialmente $Ris=1$ e $N=n$
- *Algoritmo:*
 - Fino a che $N>0$
 - Calcola $Ris * a$ e memorizzalo in Ris
 - Decrementa N
- *Correttezza:*
 - Al termine $Ris=a$ elevato alla n



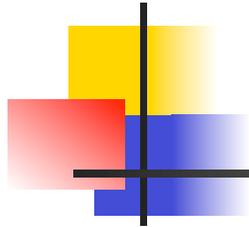
Linguaggi di Programmazione

- Scopo: *descrivere in maniera rigorosa un algoritmo*
- Classi di linguaggi:
 - Linguaggio macchina
 - Dipendono dall'hardware
 - Linguaggio ad alto livello
 - C, C++, Java, Virtual Basic



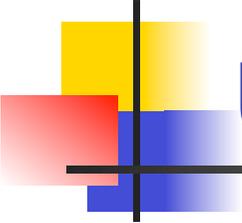
Esempio in Pseudo Pascal

```
Program potenza;  
  Integer Ris, N, A;  
  Read(N);Read(A);  
  Ris=1;  
  While (N>0) do  
    Ris=Ris*A;  
    N=N-1;  
  Print(Ris);
```



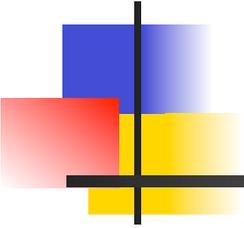
Esempio

- Il precedente programma va tradotto in **linguaggio macchina** (comprensibile all'elaboratore) cioè viene compilato in sequenze di **istruzioni**
- Quando le istruzioni vengono eseguite il programma prende dati in ingresso (valori iniziali di N e A) attraverso la tastiera (*input*) e poi stampa il risultato sul video (valore finale di Ris) (*output*)
- In generale un programma può essere visto infatti come una **funzione** da *input* ad *output*.



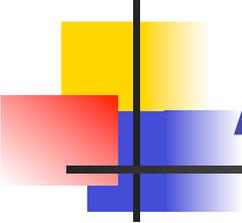
Utilizzo di un elaboratore

- Come utente:
 - Uso software applicativo esistente per creare documenti e interfacce grafiche, effettuare calcoli, navigare in rete
- Come sviluppatore:
 - Creo nuovi programmi sullo strato del software esistente
 - Nuovi programmi applicativi
 - Nuovi programmi di sistema (cioè che fanno funzionare il calcolatore)



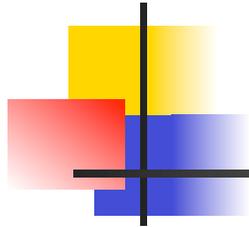
Parte I: Hardware





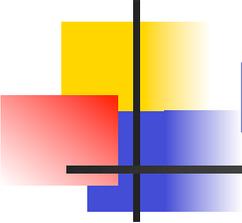
Architettura dei Sistemi Informatici

- *Sistemi informatici* PC, terminali e reti
- *Architettura* insieme delle componenti del sistema, descrizione delle loro funzionalità e della loro interazione
- Suddivisione principale *hardware e software*



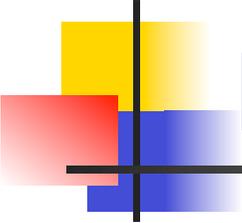
Hardware

- **Unità di Elaborazione** (Processore o CPU):
 - Svolge le elaborazioni
 - Coordina il trasferimento dei dati
 - Cioè esegue i programmi
- **Memoria Centrale**
 - Memorizza dati e programmi per l'elaborazione
 - Volatile
 - Accesso rapido
 - Capacità limitata



Hardware

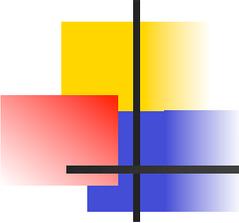
- **Memoria Secondaria** (DVD, harddisk, floppy)
 - Grande capacità
 - Persistente
 - Accesso piu lento della RAM
- **Unità Periferiche**
 - Interfaccia verso l'esterno
 - Terminali (tastiera, video)
 - Stampanti



Hardware

- Bus di Sistema

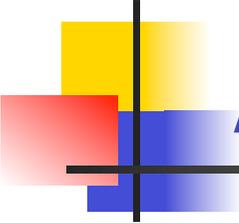
- Collega le altre componenti
 - RAM
 - Memorie Secondarie
 - Periferiche
- Insieme di collegamenti di vario tipo



Esempi: Personal Computer (PC)

- *Contentitore con*
 - CPU, RAM
 - Memoria Centrale
 - Fisso
 - Unità per Dischetti/CD
- Monitor
- Tastiera

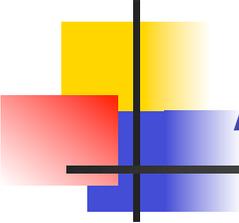




Alcuni accessori per PC

- Lettore Floppy, CD, DVD
- Modem
- Mouse
- Stampante
- Scanner
- Joystick

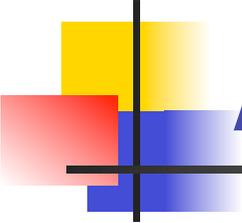




Altri Sistemi Informatici

- *Workstation*
 - Calcolatore con elevate prestazioni
- *Main-frame*
 - Servono reti di terminali con centinaia di utenti
- *Notebook (palmari) e palmari*
 - Elaboratori portatili





Altri Sistemi Informatici

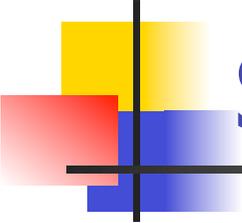
- Reti di Calcolatori

- *Reti Locali*

- collegano terminali vicini tra loro
(ad es. il nostro laboratorio)

- *Reti Geografiche*

- collegano dei calcolatori a medio-grandi
distanze
(ad es. Internet)



Software

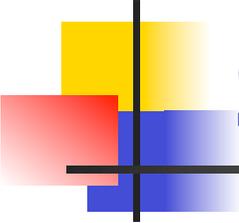
- *Software di base:*

- Dedicato alla gestione dell'elaboratore
- Esempio: **sistema operativo**

- *Software applicativo:*

- Dedicato alla realizzazione di specifiche applicative
- Esempio:
 - programmi per scrittura,
 - gestione aziendale,
 - navigazione su internet, ...





Sistema Operativo

- Rende la componente hardware facile da usare
- Fornisce funzionalità ad alto livello agli utenti
- Ad esempio:
 - organizza la memoria di massa
 - gestisce comandi immessi dall'utente:
 - Esegui un programma! Mostra i dati su video!
- Se il sistema è **multi-utente** deve gestire le risorse disponibili cercando di soddisfare tutti gli utenti
- Esempi: MS DOS, OS 2, Windows, Unix

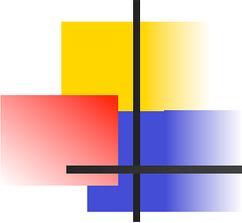




Software Applicativo

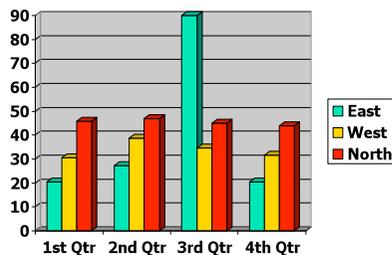
- Video Scrittura
 - per costruire e testi e definire formati di stampa
- Agende elettroniche
 - indirizzario, calendari
- Posta Elettronica
 - per comunicazione
- Fogli elettronici
 - per elaborazioni contabili
- Database
 - sistemi per la gestione di dati

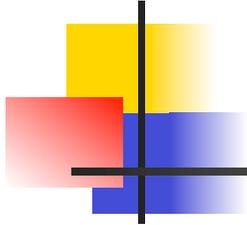




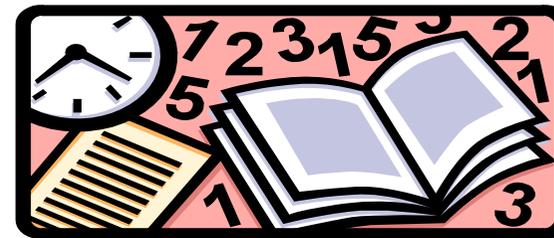
Applicazioni

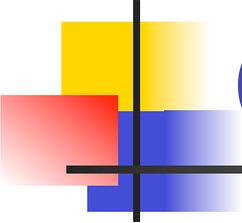
- **Calcolo Numerico:** statistiche, ecc
- **Gestione Aziendale:** banche, assicurazioni,
- **Telematica:** bancomat, ecc
- **Automazione industriale:**, robotica, ecc
- **Internet:** commercio virtuale, ecc





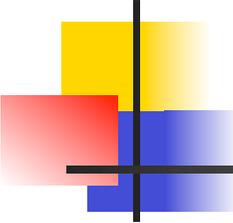
Rappresentazione della Informazione





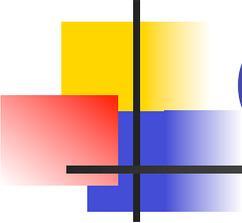
Codifica dell'informazione

- Il calcolatore memorizza ed elabora vari tipi di informazioni
 - Numeri, testi, immagini, suoni
- Occorre rappresentare tale informazione in formato facilmente manipolabile dall'elaboratore
- Si utilizza una rappresentazione digitale



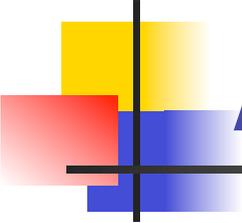
Codifica digitale

- L'unità minimale di rappresentazione è il ***bit*** (binary digit – cifra digitale): **0** o **1**
- Informazioni complesse si memorizzano come sequenze di bit
- Una sequenza di **8 bit** viene chiamata ***Byte***
 - 0 0 0 0 0 0 0 0
 - 0 0 0 0 0 0 0 1
 -



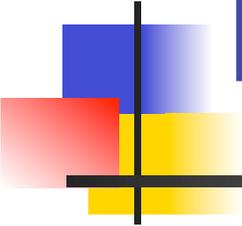
Codifica dell'informazione

- Per codificare in nomi delle province liguri mi bastano 2 bit
- Ad esempio:
 - 0 0 per rappresentare Genova
 - 0 1 per rappresentare La Spezia
 - 1 0 per rappresentare Imperia
 - 1 1 per rappresentare Savona
- In generale su **N** bit si possono codificare **2^N** informazioni (tutte le possibili combinazioni di 0 e 1 su N posizioni)
- Con un byte si possono codificare quindi **$2^8 = 256$** possibili informazioni



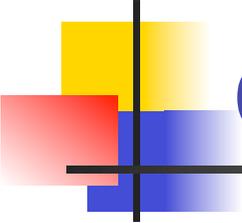
Altre unità di misura

- **KiloByte (KB), MegaByte (MB), GigaByte (GB)**
- Per ragioni storiche in informatica Kilo, Mega, e Giga indicano però le **potenze di 2** che più si avvicinano alle corrispondenti potenze di 10
- Più precisamente
 - $1 \text{ KB} = 1024 \times 1 \text{ byte} = 2^{10} \sim 10^3 \text{ byte}$
 - $1 \text{ MB} = 1024 \times 1 \text{ KB} = 2^{20} \sim 10^6 \text{ byte}$
 - $1 \text{ GB} = 1024 \times 1 \text{ MB} = 2^{30} \sim 10^9 \text{ byte}$
 - ...
- I multipli del byte vengono utilizzati come unità di misura per la capacità della memoria di un elaboratore



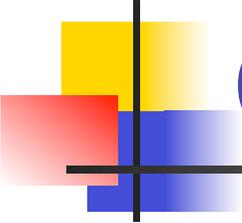
La Codifica dei Caratteri

A B ... a b & % \$...



Codici per i simboli dell'alfabeto

- Per rappresentare i simboli dell'alfabeto anglosassone (0 1 2 ... A B ... A b ...) bastano 7 bit
 - Nota: *B* e *b* sono simboli diversi
- Per l'alfabeto esteso con simboli quali &, %, \$, ... bastano 8 bit come nella codifica accettata universalmente chiamata ASCII
- Per manipolare un numero maggiore di simboli la Microsoft ha introdotto la codifica UNICODE a 32 bit (2^{32} caratteri)

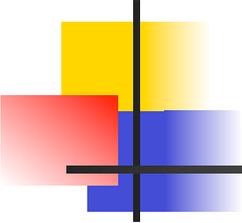


Codifica ASCII

- La codifica ASCII (American Standard Code for Information Interchange) utilizza codici su 8 bit
- Ad esempio
 - 0 1 0 0 0 0 1 rappresenta A
 - 0 1 0 0 0 0 1 0 rappresenta B
 - 0 1 0 0 0 0 1 1 rappresenta C
- Le parole si codificano utilizzando sequenze di byte
 - 01000010 01000001 01000010 01000001
B A B A

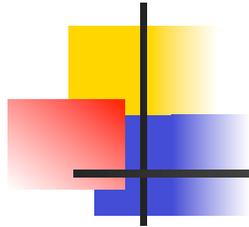
Codifica di immagini



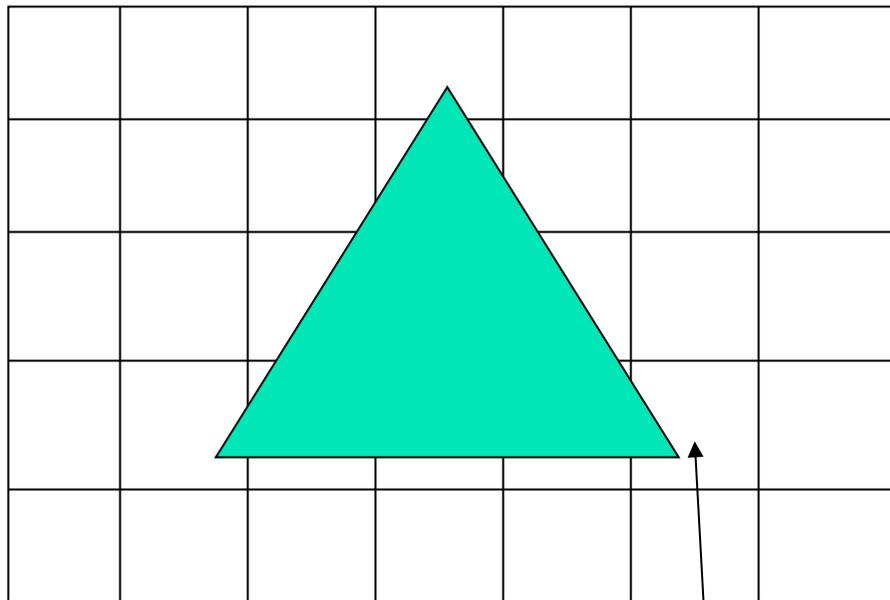


Pixel – Picture element

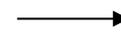
- Le immagini vengono scomposte in griglie
- Le caselle di una griglia vengono chiamate *pixel*
- La risoluzione indica il numero di pixel in cui è suddivisa un'immagine
 - Risoluzione tipica di uno schermo video 800 x 600, 1024 x 768



Codifica di un'immagine

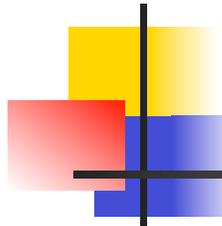


Pixel = 1



```
0 0 0 1 0 0 0
0 0 1 1 1 0 0
0 0 1 1 1 0 0
0 1 1 1 1 1 0
0 0 0 0 0 0 0
```

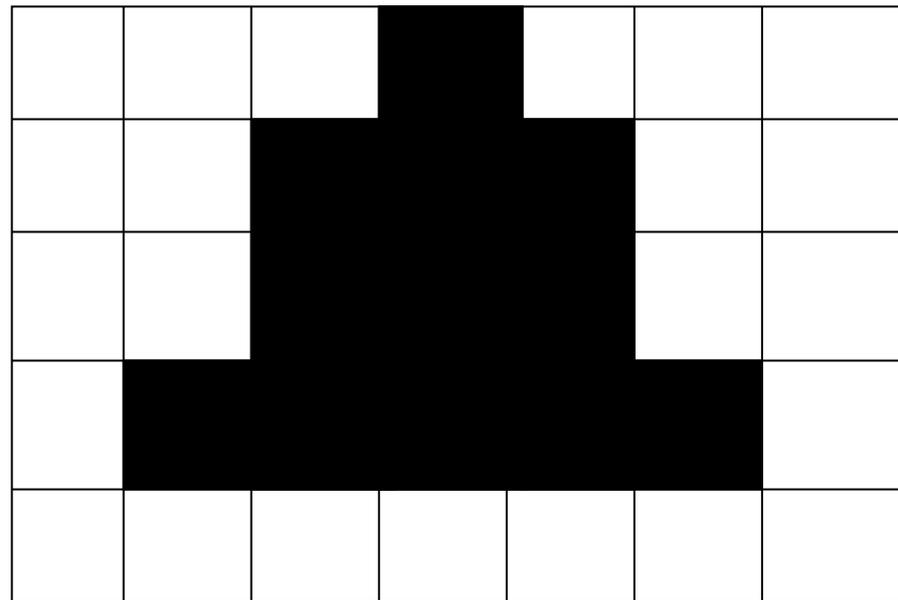
codifica



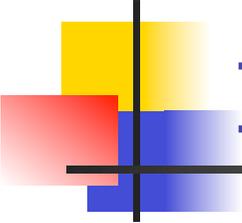
Decodifica

0 0 0 1 0 0 0
0 0 1 1 1 0 0
0 0 1 1 1 0 0
0 1 1 1 1 1 0
0 0 0 0 0 0 0

Codifica

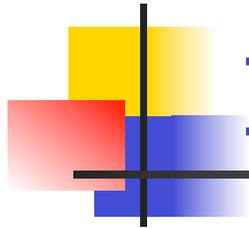


Immagine



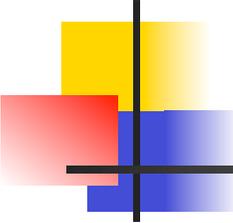
Immagini in toni di grigio

- Se si assegna un solo bit a ogni pixel si rappresentano immagini in bianco e nero
 - 0 = bianco 1 = nero
- Per poter rappresentare immagini più complesse
 - si codificano i toni di grigio
 - Si associa una codifica di un tono di grigio ad ogni pixel

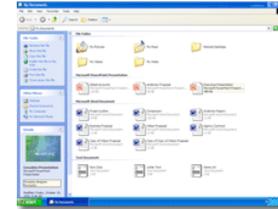


Immagini a colori

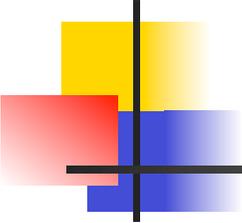
- Nella codifica RGB si utilizzano tre colori rosso
 - (Red), verde (Green) e blu (Blue):
- Ad ogni colore si associa un certo numero di sfumature codificate su N bit (2^N possibili sfumature)
- Ad esempio
 - se si utilizzano 2 bit per colore si ottengono 4 sfumature per colore
 - ogni pixel ha un codice di 6 bit
- Con 8 bit si ottengono 256 sfumature e 256^3 (16 milioni) possibili colori



Bitmap



- La rappresentazione di un'immagine mediante la codifica a pixel viene chiamata *bitmap*
- Il numero di byte richiesti per memorizzare una bitmap dipende dalla risoluzione e dal numero di colori
- Es. se la risoluzione è 640x480 con 256 colori occorrono 2.457.600 bit = 307 KB
- I formati bitmap più conosciuti sono BITMAP (.bmp), GIF (.gif), JPEG (.jpg)
- In tali formati si utilizzano metodi di *compressione* per ridurre lo spazio di memorizzazione



Rappresentazione dei suoni

- Si effettuano dei campionamenti su dati analogici
- Si rappresentano i valori campionati con valori digitali
- La frequenza del campionamento determina la fedeltà della riproduzione del suono