



**Corso Luigi Einaudi, 55 - Torino**

**Appunti universitari**

**Tesi di laurea**

**Cartoleria e cancelleria**

**Stampa file e fotocopie**

**Print on demand**

**Rilegature**

**NUMERO: 533**

**DATA: 22/04/2013**

# **A P P U N T I**

**STUDENTE: Marchisa**

**MATERIA: Informatica Risassunto**

**Prof. Mezzalama**

Il presente lavoro nasce dall'impegno dell'autore ed è distribuito in accordo con il Centro Appunti.

Tutti i diritti sono riservati. È vietata qualsiasi riproduzione, copia totale o parziale, dei contenuti inseriti nel presente volume, ivi inclusa la memorizzazione, rielaborazione, diffusione o distribuzione dei contenuti stessi mediante qualunque supporto magnetico o cartaceo, piattaforma tecnologica o rete telematica, senza previa autorizzazione scritta dell'autore.

**ATTENZIONE: QUESTI APPUNTI SONO FATTI DA STUDENTIE NON SONO STATI VISIONATI DAL DOCENTE.  
IL NOME DEL PROFESSORE, SERVE SOLO PER IDENTIFICARE IL CORSO.**

L'INFORMATICA è la scienza che studia l'elaborazione automatica di dati mediante apparati realizzati con tecnologia elettronica.

Un MICROPROCESSORE contiene in sé tutti i circuiti necessari per realizzare un piccolo elaboratore.

Un computer ha un alfabeto ridotto a due soli simboli; "0" e "1" per conseguenza avere due soli simboli crea non pochi problemi.

La codifica dell'informazione in un computer si ottiene componendo successioni di un unico simbolo che può assumere i valori 0 e 1. Questa entità è indicata come BIT (cifra binaria) ed è l'elemento base su cui operano i nostri computer.

I bit sono entità infinitesime che necessitano di essere raggruppati per poter essere facilmente manipolati.

Il modo più comodo di raggrupparli in entità più corpose è quello di considerare 8 bit insieme, dette BYTE.

MB = megabyte = milioni di byte  $\approx 2^{20}$

K → Kilo → mille  $\approx 2^{10}$

G → Giga → miliardi  $\approx 2^{30}$

T → tera → migliaia di miliardi  $\approx 2^{40}$

P → peta → milione di miliardi  $\approx 2^{50}$

non sono esattamente uguali ai nostri \*k nel contario in base 10

calcolatori per contare, a vircoli tecnologici, adoperano la base 2.

I bit e i byte per essere ~~trasmessi~~ elaborati e ~~trasmessi~~ hanno bisogno di apparecchiature che li manipolino.

La componente fisica di un sistema di elaborazione è detta HARDWARE. Si tratta principalmente di circuiti elettronici racchiusi in contenitori; il corpo del computer. Ma nel corpo risiede un'anima, che dà vita al corpo, è il SOFTWARE, esso è l'insieme di programmi che risiedono sui nostri elaboratori.

Ogni programma è una sequenza di istruzioni che realizza un algoritmo, cioè una sequenza di passi che eseguono un dato compito.

I programmi, come i dati, per essere eseguiti dal computer devono essere critti come sequenze di 0 e di 1.

Le prestazioni di calcolo di un computer si misurano valutando quante istruzioni è in grado di eseguire in un secondo.

L'unità di misura predefinita è il MIPS (milioni di istruzioni al secondo). Le istruzioni considerate sono quelle interne al computer, quelle codificate come sequenze di bit, dette istruzioni macchina.

FLOPS sono le numero di operazioni aritmetiche su numeri reali svolte in un secondo.

La notazione posizionale può essere facilmente estesa a numeri con parte frazionaria introducendo le potenze negative della base.

Nel caso di numeri binari, un generico numero  $N^*$  con parte frazionaria è espresso in forma polinomiale come:

$$N^* = \sum_{k=-j}^k d_k \cdot 2^k = d_k 2^k + d_{k-1} 2^{k-1} + \dots + d_1 2^1 + d_0 2^0 + d_{-1} 2^{-1} + \dots + d_{-j} 2^{-j}$$

Il passaggio dalla parte frazionaria di un numero decimale, alla corrispondente binaria si ottiene attraverso un algoritmo:

- si inizia dalla parte frazionaria  $F$  del numero decimale e la moltiplica per 2. Il numero così ottenuto avrà una parte intera e una frazionaria  $F'$ . La cifra intera è la cifra più significativa del numero binario frazionario corrispondente.
- si considera il valore  $F'$  e lo si moltiplica per 2, la cifra intera ottenuta è la nuova cifra del numero binario frazionario.
- si continua finché la parte frazionaria  $F'$  si annulla oppure si raggiunge un prefissato numero di cifre binarie.

Le operazioni aritmetiche nel sistema binario si eseguono secondo le stesse regole del sistema decimale.

- 1+1 genera un riporto (carry) che deve essere sommato alle cifre della colonna adiacente a sinistra.

ADDIZIONE

0+0 = 0
0+1 = 1
1+0 = 1
1+1 = 0

SOTTRAZIONE

0-0 = 0
1-1 = 0
1-0 = 1
0-1 = 1 → borrow

↓  
~~riporto~~ prestito to

- 0-1 genera un ~~riporto~~ <sup>prestito</sup> dalla colonna adiacente a sinistra quindi sarà = 1 e nella colonna a sx ~~to~~ si andrà ad aggiungere un -1

L'operazione di moltiplicazione, come quella di divisione, seguono le stesse regole che applichiamo per i numeri decimali. Esse consistono di somme o sottrazioni successive.

STENA ESASIMALE → base 16  
ciascuna cifra esadecimale corrisponde a 4 bit.

Nella maggior parte delle applicazioni servono numeri positivi e negativi. Nella prassi normale la rappresentazione dei numeri relativi è ottenuta antepoendo il valore (assoluto) del numero i segni + e -. Tale prassi può essere anche adottata negli elaborati, grazie al fatto che le neghe è un'entità binaria potendo assumere due soli stati + e -. Essi possono pertanto essere codificati con un bit: si assume che il + sia uno 0 e che il meno sia a 1 → sia il segno, sia le cifre sono codificati dagli stessi numeri.

Tutti i sistemi di elaborazione utilizzano, al loro interno, un numero fisso di bit per rappresentare qualsivoglia numero, indipendentemente dal numero di cifre numeriche che esso richiede

Essa prende spunto dall'asserzione che ogni numero può essere approssimato come prodotto delle sue cifre significative per una potenza della base di numerazione la cui esponente dipende dalla posizione della virgola nel numero di partenza.

$$\text{es. } 0,0012 = 12 \times 10^{-4}$$

In generale qualunque numero ammette la seguente rappresentazione:

$$N = \pm M \times r^E$$

↓  
mantissa del numero

↓  
rappresenta le sole cifre significative del numero

In sintesi un generico numero  $N$  può essere memorizzato in un computer mediante una terzina: segno, mantissa (valore assoluto), esponente (numero intero relativo).

a base, essendo identica per tutti i numeri, può essere omissa.

Nella memoria degli elaboratori, un numero in virgola mobile è organizzato al seguente modo:

Se si utilizzano 32 bit, pari a 4 byte, i bit sono suddivisi secondo la seguente convenzione:

- 1 bit più significativo (a sinistra) rappresenta il segno
- i successivi 8 bit l'esponente
- i restanti 23 bit la mantissa

rispetto alla rappresentazione dei numeri come interi, quella in virgola mobile a se vantaggioso che, a parità di bit utilizzati, si possono rappresentare intervalli di numeri molto più ampi. La rappresentazione intera, però, mantiene costante la precisione assoluta; la rappresentazione in virgola mobile mantiene costante la precisione relativa.

Tutti i sistemi di elaborazione utilizzano, al loro interno, un numero fisso di bit per rappresentare qualsivoglia numero, indipendentemente dal numero di cifre significative che esso possiede. Fissato il numero di bit, stabilito dall'hardware, l'intervallo di rappresentazione dei numeri è fissato e non estendibile.

Con  $n$  bit, si hanno i seguenti intervalli di rappresentazione:

- numeri assoluti  $(0, 2^n - 1)$
- numeri interi in modulo e segno  $(-2^{n-1}, +2^{n-1} - 1)$
- numeri interi in complemento a due  $(-2^{n-1}, +2^{n-1} - 1)$

Il fatto di avere un intervallo di rappresentazione finito può determinare alcuni problemi nella manipolazione dei numeri.

Un problema si presenta quando il risultato di un'operazione aritmetica è esterno all'intervallo di rappresentazione. Tale situazione è indicata con il termine di **OVERFLOW**. Quando un errore di questo tipo si presenta, i circuiti lo rilevano e l'hardware blocca il computer o ~~lo~~ avverte il sistema operativo.

Nella rappresentazione dei numeri ~~esiste~~ un altro errore è molto rilevante; l'**ERRORE di APPROSSIMAZIONE**, che consiste nel fatto che un numero espresso con un certo numero di cifre è rappresentato su un numero di cifre inferiore.

Boole traslò che il ragionamento umano è basato su proposizioni che possono assumere due valori: VERO o FALSO.

Qualsiasi sistema i cui elementi assumono due stati può essere modellato mediante l'algebra booleana. I circuiti logici che ne derivano sono gli elementi costitutivi dei calcolatori.

Nell'ALGEBRA di BOOLE si mettono in corrispondenza le proposizioni, o gli eventi binari, con le variabili logiche (o booleane), che possono assumere solo due valori: 1 - VERO

0 - FALSO

Una funzione è descritta in modo esaustivo stabilendo, per ogni combinazione delle variabili di ingresso, se vale 1 oppure 0. Si crea dunque una tabella, detta tavola di verità della funzione.

La funzione che assume il valore 0 per tutte le combinazioni delle variabili d'ingresso è detta CONTRADDIZIONE; quella che assume il valore 1 per tutte le combinazioni è detta TAUTOLOGIA.

Alle variabili logiche esistono gli operatori logici, il cui effetto può essere descritto la tavola di verità.

→ prodotto logico ⇒ simile prodotto matematico

\* operatore AND: Ha il significato di congiunzione logica. ( $A \text{ AND } B$ ;  $A \wedge B$ ;  $A \cdot B$ )

\* operatore OR: Ha il significato di disgiunzione logica ( $A \text{ OR } B$ ;  $A \vee B$ ;  $A + B$ )

Viene anche detto somma logica, data la somiglianza con l'analogo operatore matematico.

\* operatore NOT: Ha il significato di negazione logica ( $\text{NOT } A$ ;  $\bar{A}$ ;  $\neg A$ )

Per realizzare la funzione di NOT occorre un dispositivo attivo.

Combinando variabili e operatori logici si ottengono le espressioni logiche. A queste sono state trasferite alcune regole che erano proprie delle espressioni aritmetiche.

La Priorità degli operatori: prevale il NOT, poi l'AND e infine l'OR.

Le espressioni possono essere valutate per ogni combinazione delle variabili riferenti e possono assumere il valore 0 o 1; dunque anche le espressioni si appresentano mediante la tavola di verità.

Due espressioni  $T_a$  e  $T_b$  sono equivalenti se hanno la stessa tavola di verità.

Due espressioni  $T_a$  e  $T_b$  sono complementari se sono una l'opposto dell'altra.

Due espressioni  $T_a$  e  $T_b$  sono dette duali se l'una è ottenuta dall'altra scambiando gli AND con gli OR e vice versa.

L'operatore XOR indica la diversità: fornisce 1 in uscita se e solo se gli ingressi sono differenti.

Esistono teoremi che permettono di derivare espressioni equivalenti con minor numero di termini, e sono quindi utilizzabili per semplificare le espressioni booleane. È stato dimostrato che ne vale un teorema, vale anche il teorema duale.

$$\text{a) } x \cdot 0 = 0$$

$$\text{Duale: } x \parallel 1 = 1$$

$$\text{b) } x \cdot 1 = x$$

$$\text{Duale: } x \parallel 0 = x$$

I registri sono quindi celle di memoria che, costituiscono le cuore del calcolatore (CPU)

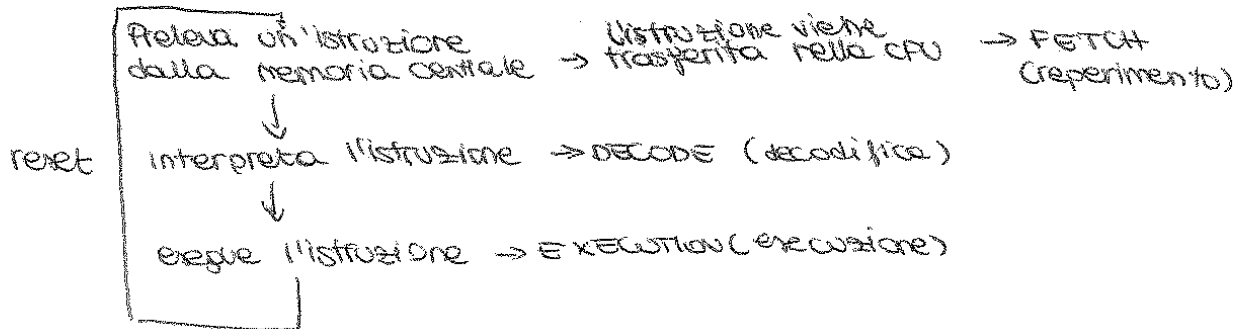
Un'istruzione è sempre formata da un codice operativo, cioè da una sequenza di bit che la identifica in modo univoco.

2 CPU commerciali tendono ad avere linguaggi simili, se prodotte dallo stesso costruttore.

Una CPU è pertanto un dispositivo che esegue un'istruzione dopo l'altra di un dato programma in formato binario.

Un programma per poter essere eseguito deve risiedere nella memoria centrale, o generale, la RAM del sistema. È compito del sistema operativo trasferire un programma dal file eseguibile, che lo contiene, alla memoria centrale.

CPU → esecuzione di un'istruzione macchina:



Le varie fasi che intervengono nell'esecuzione di un'istruzione, si succedono sequenzialmente nel tempo in modo sincrono scandite da un orologio interno, strettamente dipendente dal clock primario, quello che scandisce tutte le operazioni del microprocessore. La sua frequenza è determinata da un circuito speciale, detto oscillatore, e espressa in Hertz, o cicli al secondo. Solo in pochissime situazioni la durata di un'istruzione coincide con il periodo di tempo del clock.

L'estrazione di un'istruzione dalla memoria, la sua decodifica ed ed esecuzione richiedono generalmente alcuni periodi del segnale di clock.

Il tempo occorrente per eseguire un'istruzione è detto "ciclo di istruzione", ed è quindi il numero di periodi di clock necessari ad eseguire un'istruzione.



• UNITÀ di CALCOLO o ALU

Esegue le operazioni matematiche (aritmetiche), le operazioni logiche (AND, OR, EX-OR) e altre elaborazioni sui dati.

Al completamento di ogni operazione la ALU memorizza in opportuni elementi hardware, detti FLAG, le condizioni risultanti. In base al valore di questi flag l'elaboratore è in grado di prendere opportune decisioni, quali ad esempio interrompere il programma nel caso di overflow.

Nella maggior parte degli elaboratori la ALU è associata a un'ulteriore unità che opera direttamente numeri in formato floating point e calcola le principali funzioni trigonometriche ed esponenziali. Tale modulo, detto FPU, permette di accelerare i calcoli aritmetici o di funzioni elementari eseguendo direttamente in hardware, mediante circuiti dedicati, i calcoli stessi.

• REGISTRI e MEMORIA di LAVORO

I registri sono unità di memoria ad altissima velocità di accesso, interne alla CPU.

Essi contengono i dati di immediato impiego, evitando di doverli reperire



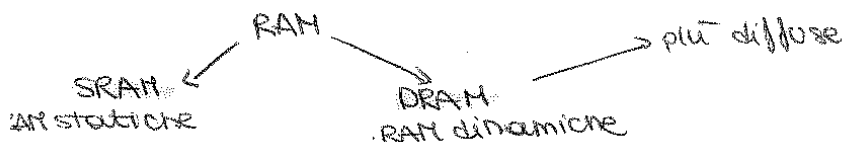
Le memorie RAM sono memorie in cui l'informazione può essere letta o scritta dalla CPU in una qualsiasi locazione con un tempo di accesso indipendente dalla sua posizione. Esse sono di tipo volatile: l'informazione contenuta nelle celle è persa se manca l'alimentazione. Per ovviare a tale inconveniente sono state introdotte le memorie ROM, cioè memorie a sola lettura.

L'operazione di scrittura viene effettuata una volta per tutte dal costruttore e l'informazione non si perde in assenza di alimentazione. Tutti gli elaboratori devono disporre almeno di una piccola ROM. In caso contrario non potrebbero avviarsi.

La ROM contiene un programma chiamato BIOS, che svolge le azioni di avviamento dell'elaboratore, dette bootstrap, e che si incarica di caricare nella memoria RAM, che all'inizio è vuota, il sistema operativo prelevandolo dai file contenuti nel disco rigido.

Un altro tipo di memoria simile è rappresentata dalle memorie PROM (programmable PROM) che si comportano in modo analogo alle ROM, ma possono essere scritte, anche se una volta sola, dall'utente, mediante dispositivi ad hoc.

Le EPROM possono essere cancellate e riscritte più volte.



Tecnologicamente differiscono nel modo in cui è realizzata la memorizzazione del singolo bit in termini circuitali.

Le prime sono caratterizzate da una maggiore velocità di lettura/scrittura ma da una minor densità di bit per chip; le seconde, più lente, hanno però il pregio di garantire un maggior impaccamento, richiedono un numero assai inferiore di chip e di spazio.

I singoli dispositivi (chip) di RAM sono di norma saldati su un'unica piastrina per realizzare un modulo o banco di memoria. Tali moduli sono chiamati DIMM.

La memoria centrale di un calcolatore può essere costituita da dispositivi di un solo tipo o di più tipi di memoria, RAM o ROM, a seconda delle applicazioni. Le macchine destinate ad un uso generale possiedono una piccola parte di memoria ROM, necessaria all'avviamento, e una gran quantità di memoria RAM, in cui vengono caricati di volta in volta i diversi programmi attivati dall'utente.

Per fine di migliorare le prestazioni di memoria, rendendo più breve il tempo medio di accesso, i progettisti di elaboratori hanno introdotto uno speciale sistema di memoria detto ~~RAM~~ **CACHE MEMORY**. Tale sistema è una memoria di limitate dimensioni realizzata con dispositivi molto veloci, del tipo SRAM.

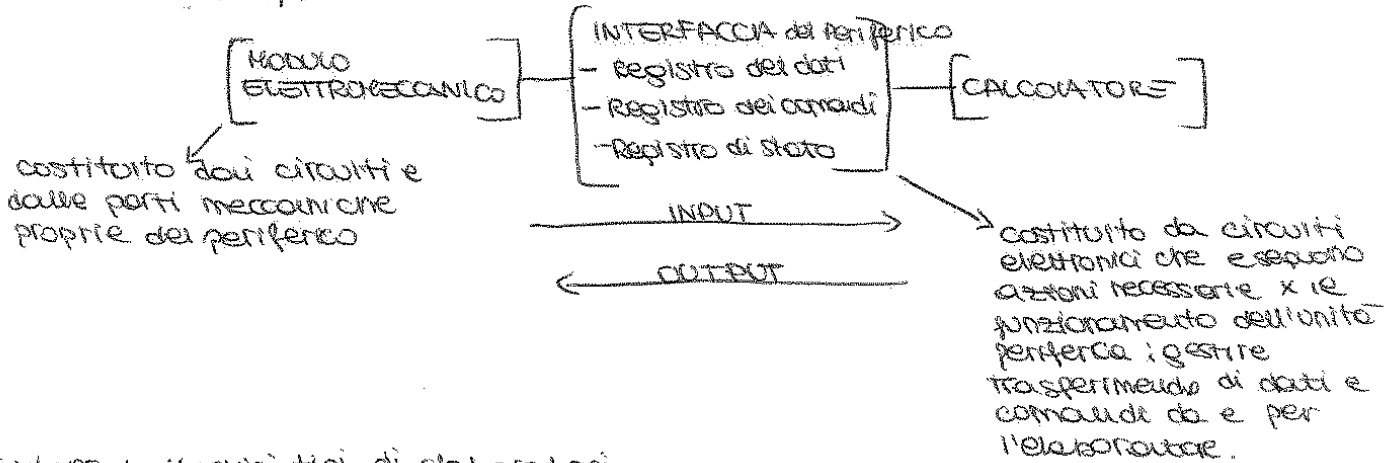
In essa vengono memorizzate, replicandole, le informazioni, dati e istruzioni, ne hanno la maggior probabilità di essere utilizzate in un breve istante temporale.

Nei recenti elaboratori è organizzata a più livelli, permettendo così un incremento di prestazioni.

Le unità di Input/Output sono gli organi che collegano il calcolatore al mondo esterno tramite dispositivi genericamente chiamati unità periferiche.

Un'unità di ingresso è un dispositivo che provvede all'introduzione di dati o istruzioni all'interno del calcolatore; un'unità di uscita, al contrario, consente l'uscita dei dati che potranno essere i risultati delle elaborazioni del nostro programma, ma anche altri segnali diretti ad altri dispositivi.

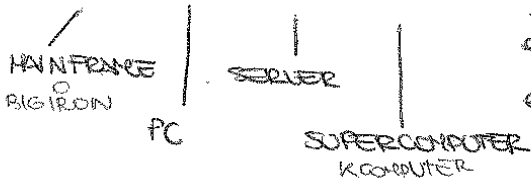
Anche le unità di memoria di massa sono gestite dall'hardware dell'elaboratore come unità di I/O



Esistono molteplici tipi di elaboratori.

- **SPECIAL PURPOSE COMPUTER** → sistemi di elaborazione dedicati  
 ↓  
 i cosiddetti sistemi "nascosti" (DVD, MP3, cellulari, ecc)  
 ↓  
 eseguono un programma fisso

- **GENERAL PURPOSE COMPUTER** → sistemi di elaborazione di tipo generale  
 ↓  
 quelli che noi chiamiamo propriamente computer  
 ↓  
 esecuzione di un qualsiasi programma



• **MAINFRAME** → potenti calcolatori, adoperati da grandi aziende pubbliche o private principalmente per applicazioni gestionali.  
 Questi sistemi più che da un'elabora potente di calcolo di tipo scientifica, sono caratterizzati dalla capacità di gestire archivi di ampie dimensioni.  
 Il vantaggio di questa classe di elaboratori è quello di permettere di poter eseguire su un'unica piattaforma hardware una molteplicità di eterogenee applicazioni software molto differenti.

• **SERVER** → erogano servizi su una rete cui sono connessi vari client di norma PC.  
 detto modello client-server  
 Si ha un server per ogni specifica applicazione.  
 • **SERVER FARM** → insieme di elaboratori server collocati in un apposito locale (centro di calcolo) presso una media o grande azienda

• **SUPERCOMPUTER** → poco diffusi, di altissimo costo e prestazioni.  
 elaboratori con una struttura interna molto particolare mirata ad ottenere capacità di calcolo molto spinte.  
 Sono realizzati con tecniche particolari che permettono di connettere migliaia di microprocessori.

Con le termine "PERIFERICHE" si fa riferimento a tutti quegli elementi che non sono propriamente classificabili come memoria centrale, tipicamente RAM, o unità di elaborazione, microprocessore.

Tali dispositivi permettono di trasferire dati dall'utente umano all'elaboratore e viceversa.

I sistemi di MEMORIZZAZIONE DI MASSA memorizzano in modo permanente, anche in assenza di alimentazione elettrica, significative quantità di dati. Inoltre memorizzano localmente grandi quantità di dati, non contenibili nella memoria centrale (RAM o ROM).

Questi sistemi sono caratterizzati dai seguenti parametri:

- capacità: la quantità di informazione memorizzabile nel dispositivo
- tempo di accesso: il tempo medio richiesto tra l'istante in cui i dati sono richiesti e quando sono disponibili.
- velocità di trasferimento: la velocità con cui i bit sono trasferiti dal supporto fisico alla memoria del computer.
- fattore di forma: indica il diametro del supporto fisico su cui sono memorizzati i dati.

es: Floppy Disk

Un significativo svantaggio delle memorie di massa rispetto alla memoria centrale è che i dispositivi di memorizzazione di massa richiedono movimenti meccanici ed hanno pertanto tempi di risposta molto più lunghi rispetto alla memoria centrale.

I dischi MAGNETICI sono il principale supporto per la memorizzazione di grandi quantità di dati. Essi consistono essenzialmente di uno o più piatti rotanti, costituiti da un'anima rivestita da materiale ferromagnetico.

A ogni superficie del disco è associata una testina di lettura/scrittura. Nell'unità a disco deve essere presente una circuiteria sofisticata, designata con le termine di CONTROLLER del DISCO che si occupa:

- del controllo della rotazione del disco
- del posizionamento delle testine
- della serializzazione dei dati in scrittura e lettura
- del trasferimento dei dati da e verso la CPU.

Le sequenze di bit sulla superficie di un disco magnetico sono organizzate in tracce concentriche, a loro volta organizzate in settori.

I dischi ottici costituiscono uno degli strumenti più diffusi nell'archiviazione di dati. (CD)

Oltre alle prestazioni grafiche di un computer o di un videogioco sono determinate dalle caratteristiche delle schede grafiche adottate. Una scheda grafica è un insieme di dispositivi che risiedono sulla scheda madre o su un'autonoma scheda, collegata internamente al computer mediante un connettore o slot, in grado di generare le immagini in termini di pixel da inviare al monitor ad essa connesso.

I programmi sono l'anima del calcolatore; senza di essi l'hardware sarebbe una inutile accozzaglia di dispositivi elettronici. Ma per scrivere un programma ci vuole un linguaggio.

Il calcolatore è una macchina che interpreta un linguaggio costituito da codici binari e lo ~~interpreta~~ esegue.

Combinando ed eseguendo molto velocemente un gran numero di operazioni elementari, il riescono a risolvere i problemi più disparati.

Si narra lo sviluppo di un programma o un insieme di programmi coordinati è strumentale alla soluzione di un problema da risolvere mediante un calcolatore.

Il primo passo è lo studio del problema nel suo dominio specifico:

Il concetto di programma presuppone tre passi fondamentali:

- definizione del problema
- esistenza di una macchina che operi secondo un determinato schema
- esecuzione sequenziale dei singoli passi del programma.

Il passo fondamentale è stabilire il meccanismo di soluzione del problema, che porta alla realizzazione di un algoritmo, cioè la sequenza di operazioni eseguendo le quali è possibile risolvere il problema dato.

Un algoritmo è costituito da un insieme finito di passi distinti e non ambigui che, eseguiti a partire da assegnate condizioni iniziali, producono output corrispondente e terminano in un tempo finito.

Le operazioni che un algoritmo può prescrivere possono essere di natura assai differente, ma devono possedere, per poter essere considerate tali, caratteristiche precise:

- ogni operazione deve essere fattibile e deve terminare entro un intervallo di tempo finito dall'inizio della sua esecuzione.
- ogni operazione deve produrre, se eseguita, un effetto osservabile che possa essere descritto.
- ogni operazione deve produrre lo stesso effetto ogni volta che venga eseguita a partire dalle stesse condizioni iniziali.

↓  
condizione che porta ad avere algoritmi deterministici: a partire dagli stessi input, si ottengono gli stessi risultati.

Il programma necessita di un linguaggio chiaro e non ambiguo. Le azioni primitive e i costrutti permessi devono essere definiti in modo chiaro. Per rendere efficaci le operazioni, si fa riferimento a degli oggetti pensandoli come contenitori detti VARIABILI, le cui nome costituisce l'IDENTIFICAZIONE.

Il linguaggio grafico volto a descrivere la sequenza di azioni: tabelle di flusso o flow chart.

Costrutto di SELEZIONE: IF... THEN... ELSE

In alcuni casi può mancare la parte Else del costrutto. → l'istruzione viene eseguita solo se la condizione è vera. In tutti i casi si prosegue con le istruzioni successive.

Costrutto DO... DO

Finché la condizione è vera, esegui l'istruzione. Solitamente l'istruzione non è una singola istruzione, ma un blocco di istruzioni, ora delle quali prima o poi modifica la condizione facendola diventare falsa, in modo da uscire dal ciclo.

Costrutto REPEAT... UNTIL = DO... WHILE

Finché prima o al momento di entrare nel ciclo la condizione fosse falsa, allora si uscirebbe subito dal ciclo. L'istruzione non verrebbe mai eseguita se l'istruzione IF