



## Prova in Itinere (soluzioni)

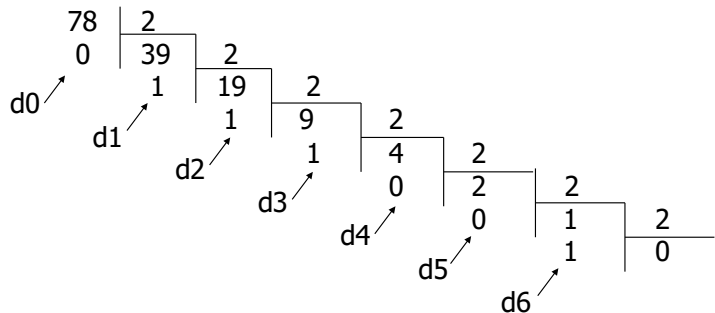
3) Convertire in binario, in **complemento a 2 su 8 bit**, il numero decimale **-78**

Valore assoluto:

$01001110_2$

Complemento a 2:

$10110010_2$



4) Convertire in decimale il seguente numero binario codificato in **modulo e segno su 8 bit**: **10111011**

Valore assoluto:  $2^5 + 2^4 + 2^3 + 2^1 + 2^0 = 32 + 16 + 8 + 2 + 1 = 59_{10}$

Segno: -

Risultato:  $-59_{10}$

## Prova in Itinere (soluzioni)

5) Convertire in decimale il seguente numero binario codificato in **complemento a 2 su 8 bit**: **10111011**

$$= -1 \times 2^7 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^1 + 1 \times 2^0 =$$

$$= -128 + 32 + 16 + 8 + 2 + 1 = -128 + 59 = -69_{10}$$

6) Eseguire la seguente somma tra numeri binari codificati in **complemento a 2 su 8 bit**: **10110101 + 01110011**

=  $00101000_2$

Verifica:  $-75_{10} + 115_{10} = 40_{10}$

$10110101_2 = -128 + 32 + 16 + 4 + 1 = -75_{10}$

$01110011_2 = 64 + 32 + 16 + 2 + 1 = 115_{10}$

$00101000_2 = 32 + 8 = 40_{10}$

$$\begin{array}{r}
 1111 \ 111 \\
 10110101 + \\
 01110011 = \\
 \hline
 100101000
 \end{array}$$

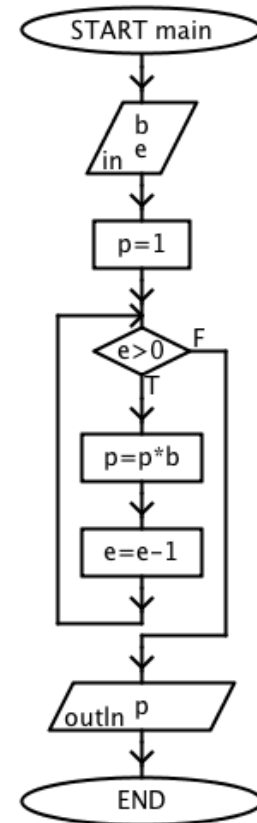
bit "perso" → 10010100 8 bit

## Prova in Itinere (soluzioni)

6) Descrivere brevemente cosa fa l'algoritmo descritto dal seguente diagramma di flusso

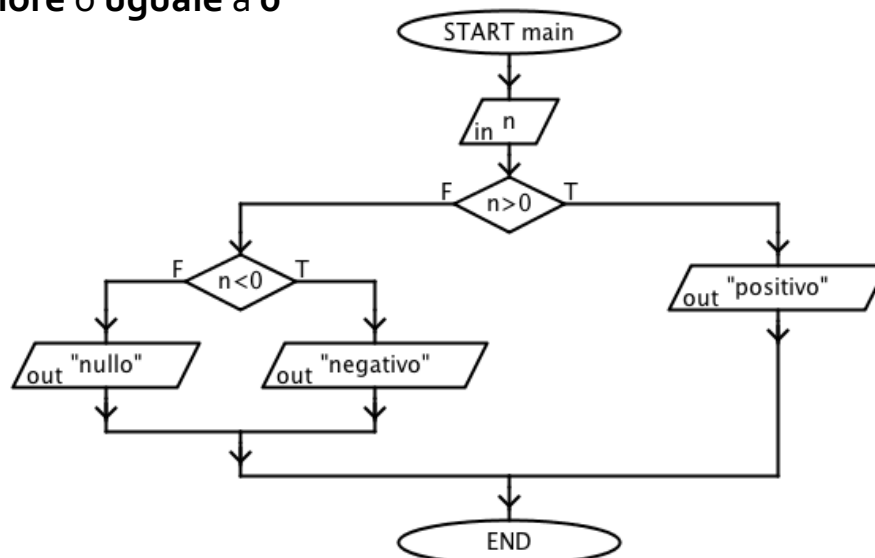
L'algoritmo chiede in input due numeri **b** ed **e**, solo se **e** è maggiore di 0, allora moltiplica **b** per se stesso **e** volte e restituisce il risultato.

In altre parole **calcola la potenza  $b^e$** .



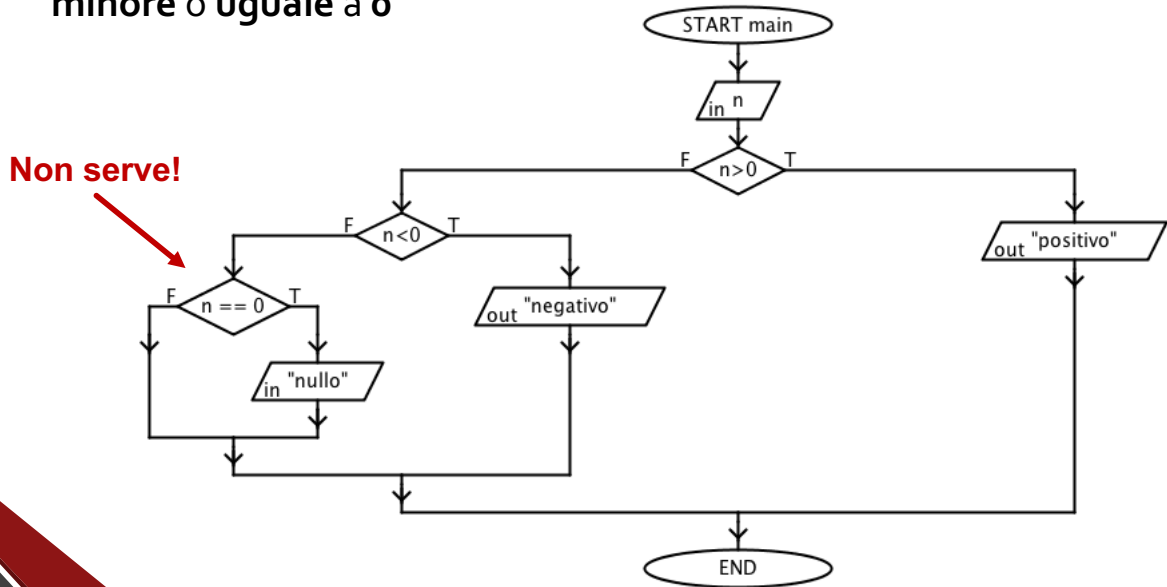
## Prova in Itinere (soluzioni)

8) Descrivere, mediante diagramma di flusso, un algoritmo che richiede in input un numero **n** e visualizza in output "positivo", "negativo" o "nullo" a seconda se **n** è **maggiore**, **minore** o **uguale** a 0



## Prova in Itinere (errore)

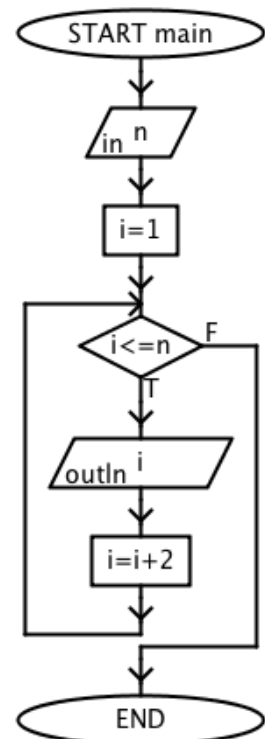
8) Descrivere, mediante diagramma di flusso, un algoritmo che richiede in input un numero  $n$  e visualizza in output "positivo", "negativo" o "nullo" a seconda se  $n$  è maggiore, minore o uguale a 0



## Prova in Itinere (soluzioni)

9) Descrivere, mediante diagramma di flusso, un algoritmo che richiede in input un numero  $n$  e visualizza in output la sequenza di numeri dispari compresi tra 1 e  $n$

Esempio: se in input l'utente inserisce 10 allora l'algoritmo restituisce la sequenza 1 3 5 7 9



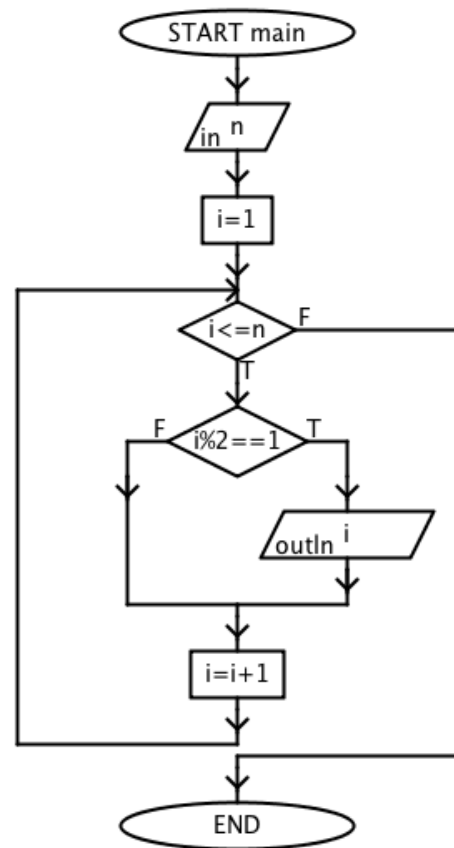


## Prova in Itinere (soluzioni)

9) Descrivere, mediante diagramma di flusso, un algoritmo che richiede in input un numero  $n$  e visualizza in output la sequenza di **numeri dispari compresi tra 1 e  $n$**

Esempio: se in input l'utente inserisce 10 allora l'algoritmo restituisce la sequenza 1 3 5 7 9

**Soluzione meno ottimizzata**

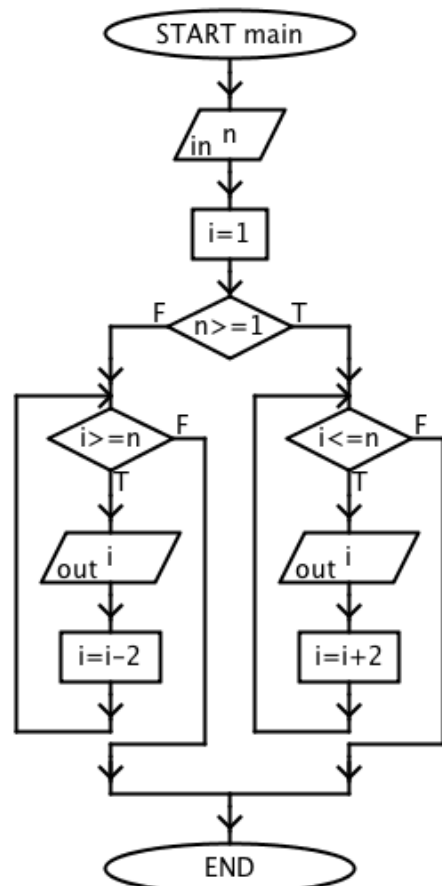


## Prova in Itinere (soluzioni)

9) Descrivere, mediante diagramma di flusso, un algoritmo che richiede in input un numero  $n$  e visualizza in output la sequenza di **numeri dispari compresi tra 1 e  $n$**

Esempio: se in input l'utente inserisce 10 allora l'algoritmo restituisce la sequenza 1 3 5 7 9

**Ammette anche  $n$  negativo**



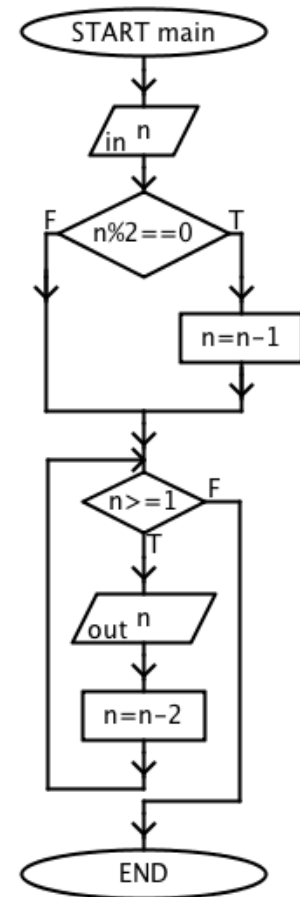
## Prova in Itinere (errore)

9) Descrivere, mediante diagramma di flusso, un algoritmo che richiede in input un numero  $n$  e visualizza in output la sequenza di **numeri dispari compresi tra 1 e  $n$**

Esempio: se in input l'utente inserisce 10 allora l'algoritmo restituisce la sequenza 1 3 5 7 9

**Scrive la sequenza a ritroso.**

**Esempio: se  $n = 10$  scrive 9 7 5 3 1**



## Programma del Corso

### Modulo 1 - Tecnologie dell'informazione e della comunicazione

- Introduzione alle ICT
- Rappresentazione Digitale dell'Informazione
- Rappresentazione Digitale dei Dati Multimediali
- Architettura Hardware di un Computer
- Software e Sistemi Operativi
- Reti di computer

# Software e Sistema Operativo

## Parte 2: Il Software Applicativo

### Bibliografia

- Par. 6.1: I programmi applicativi
- Par. 6.2: Programmi per la videoscrittura
- Par. 6.7 – 6.8: I fogli elettronici
- Par. 6.12: Le presentazioni multimediali
- Approfondimenti su queste slide



## Il Software

Il **software di sistema** racchiude in sé:

- I **linguaggi di programmazione** che servono a creare nuovi programmi
- il **sistema operativo**, necessario a far funzionare i vari elementi che compongono il computer

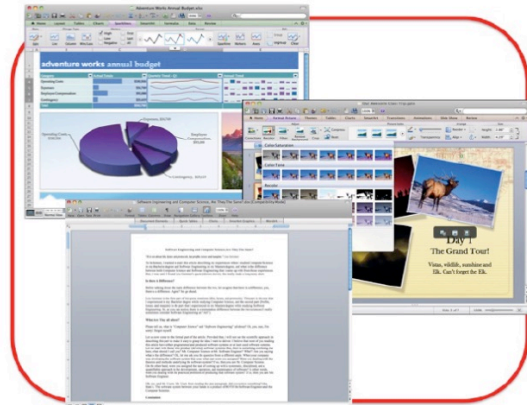
Il **software applicativo** comprende:

- programmi utilizzati per svolgere **compiti specifici**: gestire la posta, la contabilità di casa, redigere una lettera, creare una presentazione, telefonare, ecc.,
- **applicativi creati ad hoc** per risolvere un determinato problema

# Software Applicativo

## Comprende:

- **Software "orizzontale"** di utilizzo generale (posta elettronica, word processor, fogli elettronici, DBMS, browser, ecc.)
- **Software "verticale"** di utilizzo specialistico, rivolto a specifiche classi di utenti (es. gestione studio legale, software bancario, ecc.)
- **Software "personalizzato"** costruito ad hoc e dedicato a uno specifico utente (grande azienda, compagnia aerea, ecc.)



# Utility

Integrano le **funzioni** del **sistema operativo**

- **Compressore:** riduce le dimensioni di un documento
- **Antivirus:** protegge il sistema dai virus informatici
- **Uninstaller:** rimuove applicazioni e file
- **Diagnostico:** segnala e risolve problemi hardware o software
- **Tools prestazionali:** migliora le prestazioni del sistema



Integrano le **funzioni** dei **programmi applicativi**

- **Reader:** consente di visualizzare un documento senza dover acquistare il programma che lo ha generato
  - **Non permette di modificare il documento** ma solo di leggerlo
  - **Esempi:** Acrobat Reader, eBook reader, visualizzatori di Office

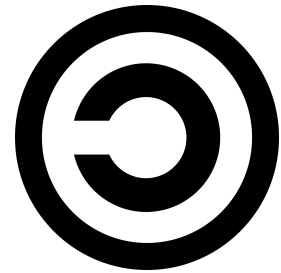
# Open Source

Identifica **software libero da copyright** distribuito insieme al **codice sorgente**

- L'utente può utilizzare il software gratuitamente
- L'utente può modificare il software
- L'utente può distribuire la versione modificata del software

## Esempi

- **Linux** (Sistema Operativo alternativo a Windows)
- **OpenOffice** (suite di programmi per l'ufficio)
- **Eclipse** (ambiente di sviluppo software)
- **MySQL** (Data Base Management System)



simbolo **copyleft**

# Open Source

## Vantaggi

- **Costi iniziali** spesso nulli
  - Non hanno necessità di **licenza d'uso**
  - Sono facilmente ottenibili (spesso tramite **download**)
- **Flessibilità**
  - Chiunque è libero di **modificare** il codice **aggiungendo funzionalità o modificando quelle esistenti**
- **Uso professionale**
  - In genere il prodotto **può essere utilizzato** in attività commerciali o professionali



# Open Source

## Svantaggi

- **Costi dei servizi a medio/lungo termine** spesso elevati
  - Formazione, gestione, migrazione dei dati, ecc.
- **Assistenza e manutenzione** non garantite
  - Alcune volte assenti, altre volte offerti solo **a pagamento**
  - Non c'è garanzia sugli **aggiornamenti**
- **Disomogeneità** del codice sorgente
  - Si tratta di un lavoro fatto a più mani il più delle volte senza (o con scarso) **coordinamento**
  - In genere la **robustezza** del software open source (assenza di bug) è inferiore a quella dal software a pagamento



# Freeware e Shareware

## Freeware

- L'utente può **utilizzare il software gratuitamente** ma non ha accesso ai sorgenti

## Shareware

- Indica un software che è possibile **provare prima di acquistare**
- La **versione di prova** (demo) tipicamente:
  - Si può scaricare dal sito del produttore
  - Ha **minori funzionalità** rispetto alla versione completa oppure ha tutte le funzionalità ma **limitazioni di tempo**
- Pagando un prezzo il produttore invia un **codice di registrazione** che sblocca le funzionalità mancanti o elimina i vincoli di tempo

# Office Automation

Tra i **software orizzontali** più diffusi ci sono quelli di **office automation** (o **produttività personale**)

- Permettono di **creare contenuti digitali** quali **documenti testuali**, **grafici** o **presentazioni**
- Sono raccolti in collezioni detti **suite**
- Spesso sono **già installati** assieme ai sistemi operativi nei comuni personal computer in vendita

## Esempi

- **Microsoft Office**
- **Apple iWork**
- **Apache OpenOffice** (open source)



# Office Automation

Le suite di **office automation** tipicamente includono:

- **Elaboratori di testo** che consentono di realizzare documenti (detti anche programmi di **videoscrittura** o **word processor**)

**Esempi:** Word, Pages, Writer



- **Fogli elettronici** che permettono di **effettuare calcoli**, **elaborare dati** e **tracciare grafici**

**Esempi:** Excel, Numbers, Calc



- **Programmi per presentazioni** che consentono di realizzare e proiettare presentazioni multimediali

**Esempi:** PowerPoint, Keynote, Impress



# Word Processor

Consentono di **comporre, modificare, formattare e stampare** documenti

## Precursori:

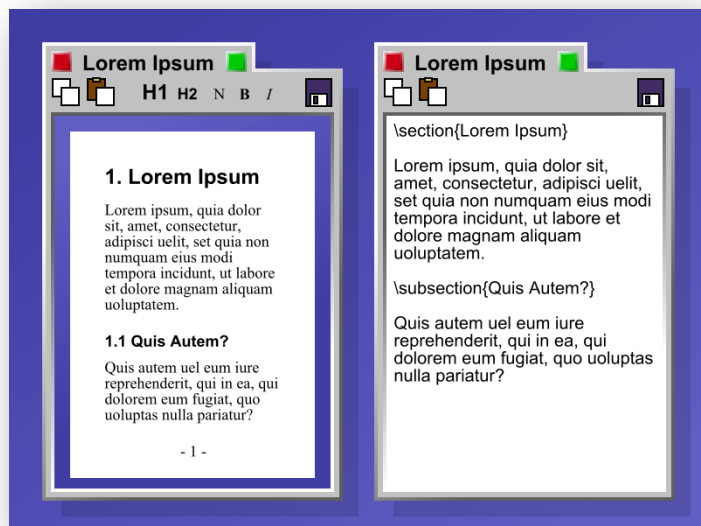
- Macchina per scrivere elettronica con **display di riga (anni '60)**
  - Consentiva la **visualizzazione delle parole** che si stavano scrivendo
  - L'utente poteva **correggere il testo** prima della stampa
- **1968**: Viene introdotto il primo **programma di videoscrittura per computer (Astrotype** per il minicomputer DEC PDP-8)
- **1983**: Viene rilasciato **Microsoft Word** per MS-DOS 1.0



# Word Processor

I moderni Word Processor usano interfacce **WYSIWYG** **What You See Is What You Get** (ciò che vedi è ciò che ottieni)

- Tali interfacce consentono di ottenere sulla carta una **disposizione di testo e immagini** uguale a quella visualizzata sullo schermo del computer





# Formattazione del Documento

1. I margini sono gli spazi bianchi che circondano il blocco del testo mettendolo in risalto. Hanno la funzione di rendere l'aspetto del documento più accattivante.

2. Le intestazioni e i piè di pagina vengono inseriti rispettivamente in cima e in fondo alle pagine per identificare le sezioni o per fornire altre informazioni utili.

3. Si possono creare elenchi numerati le cui voci vengono rinumerate automaticamente in caso di aggiunte o eliminazioni.

4. Se l'ordine delle voci è irrilevante si possono creare elenchi puntati.

Le intestazioni vengono stampate in cima alla pagina.

La formattazione del testo può avvenire durante la stesura del documento o quando questo è già stato completato.

1. È possibile creare elenchi numerati.
2. Se si aggiunge o si toglie una voce dall'elenco, i numeri vengono riassegnati automaticamente.
  - È possibile creare elenchi puntati.
  - Se si preme il tasto Invio alla fine dell'elenco, il paragrafo successivo viene puntato automaticamente.

I tipi di carattere a disposizione sono pressoché infiniti: *PEIGNOT*, *Helvetica*, *Courier*, **Chicago** ecc.

# Formattazione del Documento

5. Si può scegliere il tipo di carattere che si vuole utilizzare dalla lista dei font disponibili nel sistema. Per evidenziare alcune parti di testo, si può ricorrere ai formati grassetto, corsivo o sottolineato.

6. Le tabelle servono per visualizzare più chiaramente alcuni tipi di dati.

7. I rientri determinano di quanto deve rientrare il paragrafo dai margini destro e sinistro della pagina.

I tipi di carattere a disposizione sono pressoché infiniti: *PEIGNOT*, *Helvetica*, *Courier*, **Chicago** ecc.

Programma	Tipo	Grafici
Word	Elaboratore di testi	Sì
Excel	Foglio elettronico	Sì
Access	Base di dati	No

Ci sono diversi modi per segnalare i paragrafi. In questo caso è rientrata soltanto la prima linea, ma è possibile far rientrare anche tutte le linee, o tutte eccetto la prima.

8. I numeri di pagina possono essere posizionati sia in alto che in basso nella pagina ed essere scritti in numeri romani (i, ii, iii ecc.) o in numeri arabi (1, 2, 3 ecc.).

← Pagina 1

# Formattazione del Documento

- **Stile.** È l'insieme di **caratteristiche scelte per un testo** alle quali si dà un nome per applicarle tutte insieme
- **Modello.** Più stili possono essere conservati come un modello, che diviene una sorta di **contenitore di stili pronti all'uso**
- **Autocomposizione.** È una procedura automatizzata che **aiuta l'utente passo dopo passo a preparare documenti finali** di una determinata tipologia



Brochure



Volantino



Ricerca



Documento moderno

# Fogli Elettronici

Permettono di **effettuare calcoli, elaborare dati e tracciare grafici**

- L'utente **inserisce dei dati in una tabella** suddivisa in celle
- Ogni **cella** può contenere un **valore** o eseguire una **formula** utilizzando i valori di altre celle
- Al variare dei valori nel foglio i **risultati delle formule vengono ricalcolati automaticamente**

Il primo foglio elettronico fu **VisiCalc** per **Apple II** concepito dall'informatico **Dan Bricklin** nel 1978



	A	B	C	D
1	ITEM	NO.	UNIT	COST
2	RAKE	4	UNITS	556.00
3	CUT	4	UNITS	407.00
4	TONER	2	UNITS	4007.00
5	SNUFF	4	UNITS	9.00
6				
7			SUBTOTAL	13155.00
8			9.75% TAX	1282.66
9			<b>TOTAL</b>	<b>14438.16</b>

VisiCalc (1978)

## Fogli Elettronici

Le righe sono orizzontali e sono contrassegnate da numeri consecutivi a partire dalla riga più in alto, che porta il numero 1.

Il bordo contiene gli indicatori di colonna (lettere) nella parte superiore del foglio di lavoro e gli indicatori di riga (numeri) sul lato sinistro.

	A	B	C	D	E	F	G	H
1								
2								
3								
4								
5								
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								
16								
17								
18								
19								
20								
21								

Le colonne sono verticali e sono contrassegnate da lettere. Le prime 26 colonne sono contraddistinte dalle lettere che vanno dalla A alla Z, le successive 26 con le combinazioni da AA ad AZ, le 26 seguenti con le combinazioni da BA a BZ ecc.

Le celle sono l'unità fondamentale dei fogli elettronici e sono date dall'intersezione tra le colonne e le righe. Ogni cella è individuabile tramite un indirizzo costituito dalla lettera della colonna seguita dal numero della riga. La cella all'intersezione della colonna D con la riga 6, per esempio, ha come indirizzo D6.

## Formule

Le **formule** permettono di svolgere **calcoli immediati**

- Cominciano per = e contengono riferimenti ad altre celle

### Esempio

- La cella **B3** contiene la formula **=B1-B2**
- La formula viene sostituita nella cella con il **risultato del calcolo**
- Quando si modificano i valori delle celle B1 o B2 il valore della cella B3 **cambia automaticamente**

	A	B
1	Vendite	€ 100
2	Costi	€ 90
3	Profitti	€ 10

Valori originali

Formula  
**=B1-B2**

	A	B
1	Vendite	€ 110
2	Costi	€ 90
3	Profitti	€ 20

Variazione delle vendite

	A	B
1	Vendite	€ 100
2	Costi	€ 95
3	Profitti	€ 5

Variazione dei costi

# Formule

Nelle formule è possibile usare **librerie di funzioni predefinite**

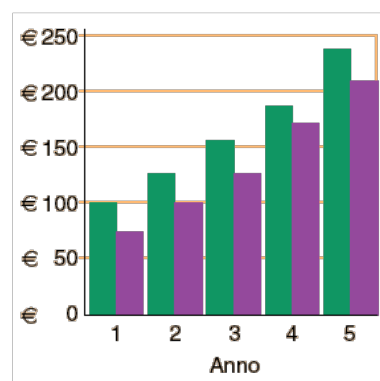
	A	B
1	<b>Regione</b>	<b>Vendite</b>
2	Est	10.000
3	Ovest	5.000
4	Centro	12.000
5	Sud	7.000
6	New England	5.000
7	Somma	39.000
8	Media	7.800
9	Minimo	5.000
10	Massimo	12.000
11	Deviazione standard	3.114,48

**=SOMMA (B2:B6)**  
**= MEDIA (B2:B6)**  
**= MIN (B2:B6)**  
**= MAX (B2:B6)**  
**= DEV.ST (B2:B6)**

# Grafici

I **grafici** sono il metodo migliore per **illustrare i dati** in maniera efficace e per interpretarli

- I fogli elettronici hanno a disposizione diversi **tipi di grafico**
- Una volta scelto il tipo, il grafico viene **generato automaticamente** a partire dall'insieme di celle selezionate



	A	B	C	D	E	F
1	<b>Voce</b>	<b>Anno 1</b>	<b>Anno 2</b>	<b>Anno 3</b>	<b>Anno 4</b>	<b>Anno 5</b>
2	Vendite	€ 100	€ 125	€ 156	€ 195	€ 244
3	Costi	€ 80	€ 100	€ 130	€ 165	€ 213
4	Profitti	€ 20	€ 25	€ 26	€ 30	€ 31

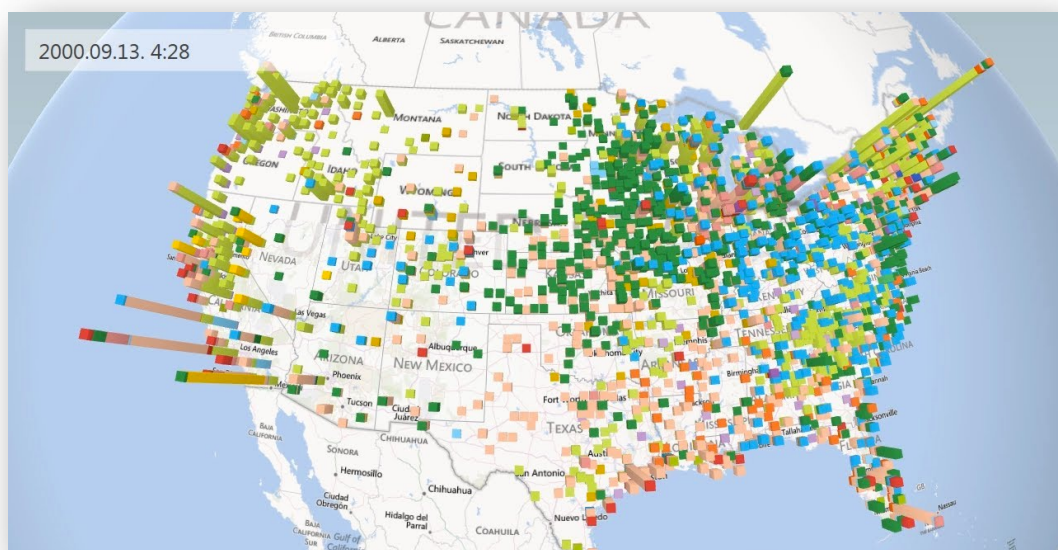
# Grafici

I fogli elettronici hanno a disposizione diversi **tipi di grafico** adatti a ogni **calcolo e statistica**



# Mappe

Per mettere in evidenza le relazioni tra **dati e luoghi** a cui i dati si riferiscono è possibile ricorrere alle **mappe**

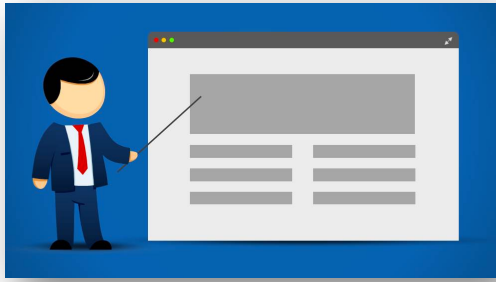




# Programmi per Presentazioni

Servono per proiettare ovvero **comunicare su schermo**, progetti, idee, lezioni e contenuti

- Possono integrare testo, immagini, grafici, filmati e audio
- Possono aggiungere effetti visivi e sonori



**Precursore:**

Proiettore per lucidi o diapositive



# Programmi per Presentazioni

Le presentazioni **automatiche** non accompagnano uno speaker ma sono **eseguite direttamente sullo schermo** dell'utente

- **non interattive** quando il flusso di informazioni non può essere alterato
- **interattive** quando l'utente è chiamato a decidere cosa e come vedere o leggere secondo diversi percorsi
  - Si usano pulsanti, icone e link ipertestuali per variare il flusso di esecuzione



Possono essere fruite su **dispositivi diversi**

# Programma del Corso

## Modulo 2 – Laboratorio di tecnologie dell'informazione e della comunicazione (aggiornamento)

- Algoritmi e Flow Chart
- Sistemi informativi e DBMS

## Sistemi Informativi e DBMS

### Bibliografia

- Par. 14.1: Introduzione alle Basi di Dati
- Par. 14.2: I Modelli di Basi di Dati
- Par. 14.3: Le Relazioni
- Par. 14.5: Query di selezione e criteri
- Approfondimenti su queste slide



# Basi di Dati

Una **Base di Dati** (o **Database** o **DB**) è una **collezione di dati** (tipicamente molto corposa)

- memorizzati in un **computer**
- organizzati in **forma strutturata**
- **condivisi tra più utenti** (anche migliaia) e da questi accessibili **contemporaneamente**
- sui quali è possibile **operare efficientemente ricerche e aggiornamenti**



Un **Data Base Management System** (o **DBMS**) è il **software** che permette di **gestire** un Database

- Memorizzare, cancellare, aggiornare o interrogare i dati

# Basi di Dati

Gli **esempi di utilizzo** di Basi di Dati sono molteplici

- Nel **commercio** per la registrazione delle informazioni su prodotti, fornitori, clienti e ordini
- Nelle **banche** per la gestione dei clienti e dei conti correnti
- Nelle **università** per la gestione delle informazioni su studenti, corsi e curriculum
- Nelle **compagnie aeree** per la gestione delle informazioni sui voli e sulle prenotazioni
- Nelle **biblioteche** per la gestione delle informazioni sui libri e sui prestiti
- ...

Volo	Compagnia aerea	Destinazione	Partenza	Stato	Neve	Flight Messaging	Area Check-in	Gate
LH 03638	Lufthansa	OLBIA COSTA SMERALDA	12:50	PARTITO	☀️		8	A36
AZ 00590 SU 00422	Alitalia	MOSCOW SHEREMETYEVO	12:50	PARTITO	-		14	B31
SK 04718	SAS	OSLO AIRPORT	13:00	PARTITO	☁️		7	A05
SQ 00377	Singapore Airlines	SINGAPORE CHANGI	13:05	PARTITO	☀️		7	B06
LH 01857	Lufthansa	MUNICH FRANZ JOSEF STRAUSS	13:15	PARTITO	☀️		8	A37
CX 00234	CATHAY PACIFIC	HONG KONG INTL	13:25	PARTITO	☀️		2	B13
KL 01624 DL 09495 AZ 07720	DELTA	AMSTERDAM	13:25	PARTITO	☀️		4	A09
PS 00312 AZ 07174	Royal Jordanian Delta International	KEY BORISPOL	13:30	IMBARCO	☀️	☐	1	B00
AT 00955	flydubai	CASABLANCA MOHAMED V	13:35	IMBARCO	☀️	☐	1	B11

L'interfaccia di un database che fornisce informazioni su **voli aerei**



## Differenza con i Fogli Elettronici

I **Fogli Elettronici** sono **adatti** a trattare:

- piccole quantità di dati
- prevalentemente di **tipo numerico**
- con **elevate dipendenze** reciproche

**Esempio:** il **bilancio di un'azienda** dove, cambiando il valore in una cella si modificano a cascata i valori presenti in tante altre celle

Inoltre i **fogli elettronici**:

- hanno notevoli **capacità analitiche**, grazie alla disponibilità di **funzioni predefinite** e alla facilità di costruirne nuove
- permettono di visualizzare i dati attraverso **grafici e diagrammi**, aiutando l'analista nella ricerca di trends

## Differenza con i Fogli Elettronici

I **Fogli Elettronici** sono invece **poco adatti** per:

- trattare **grandi quantità di dati**
- **di tipo diversificato** (testi, numeri, date, immagini, ...)
- correlati, ma **poco dipendenti** gli uni dagli altri
- **ricercare informazioni**
- gestire **numerosi utenti** contemporaneamente

**Esempio:** lo **schedario di una biblioteca**, dove:

- l'inserimento o l'eliminazione di un volume **non ha effetti** sulle informazioni che riguardano altri libri
- **più utenti contemporaneamente** possono effettuare **ricerche**

## Caratteristiche

**Definizione:** Un **DBMS** è un sistema software che gestisce collezioni di dati **grandi, persistenti e condivise** garantendo **affidabilità, privatezza, efficienza ed efficacia**



## Caratteristiche

Le **Basi di Dati** sono di **grandi dimensioni**

- Tipicamente un DB è di dimensioni molto **maggiori della memoria centrale** dei sistemi di calcolo utilizzati
- i dati devono quindi essere **gestiti in memoria di massa**
- il limite è solo quello **fisico dei dispositivi**

Le **Basi di Dati** sono **persistenti**

- i dati hanno un **tempo di vita indipendente** dalle singole esecuzioni dei programmi che li utilizzano
- I dati "sopravvivono" ai processi



# Caratteristiche

Le **Basi di Dati** sono **condivise**

- applicazioni e **utenti diversi** devono accedere agli **stessi dati** anche **contemporaneamente**

In questo modo:

- Si **riduce la ridondanza** (non è necessario duplicare gli stessi dati in più archivi con conseguente spreco di spazio)
- Si **evita l'inconsistenza** (non possono esistere versioni diverse dello stesso dato)
  - Ciò si può verificare quando un programma **aggiorna solo la propria copia di un dato**, lasciando inalterate le altre copie



# Caratteristiche

**Problematiche legate alla condivisione:**

- Alcune operazioni **non interferiscono** l'una con l'altra
  - **Esempio:** lo stesso dato può essere **letto da più utenti contemporaneamente** senza problemi
- Altre operazioni **non possono essere eseguite simultaneamente**
  - **Esempio:** un'operazione di **incremento** prevede di **leggere un dato, incrementarlo e scriverlo di nuovo**. Se nel frattempo il **dato è cambiato** l'operazione di incremento fornisce un **risultato errato**
  - In questi casi si adotta un **meccanismo di blocco**: mentre è in corso un'**operazione critica** su un dato sono **interdette altre operazioni** sullo stesso dato

## Caratteristiche

I DBMS garantiscono **Affidabilità** ovvero **resistenza a malfunzionamenti hardware e software**

- I DBMS sono in grado di **mantenere intatto** il contenuto della base di dati anche **in presenza di guasti hardware e software**
- Per far ciò si adottano **sofisticati meccanismi di backup** con i quali è possibile recuperare dati persi a seguito di guasti



## Caratteristiche

I DBMS garantiscono **Privatizza**

- Per ogni utente sono specificate le **tipologie di operazione** che può effettuare (**livello di autorizzazione**)
  - lettura / scrittura / eliminazione / esecuzione

**Esempio:** in un Database bancario:

- Un **cliente** può **leggere il suo estratto conto**, ma non modificarlo, se non tramite un'operazione di **prelievo**
- Un **cliente** non può leggere **l'estratto conto di altri**
- Lo **sportellista** può **leggere le informazioni di tutti** i clienti ma **modificarne solo alcuni** (non può modificare l'importo del fido)
- Il **direttore** può **leggere e modificare tutti i campi** dei conti

# Caratteristiche

I DBMS garantiscono **Efficienza**

- Svolgono le operazioni utilizzando il **minimo di risorse**
  - di **spazio** (memoria principale e secondaria)
  - di **tempo** (di esecuzione e di risposta)

I DBMS garantiscono **Efficacia**

- Offrono servizi e funzionalità **articolate, potenti e flessibili**
- I risultati ottenuti sono **corretti**

# Modello Logico

È l'insieme di concetti utilizzati per organizzare i dati e descriverne la **struttura**

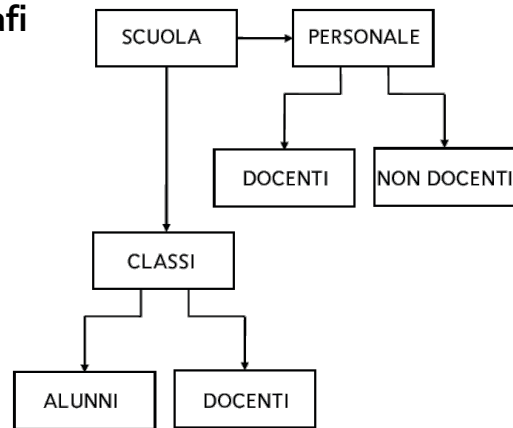
Il più diffuso è il **Modello Relazionale**, basato sul concetto matematico di **relazione**

- Definito nel **1970** dall'informatico statunitense **Edgar Frank Codd**
- Affermatosi a partire dagli **anni '80**
  - Un precursore è stato **System R** (non completamente relazionale) sviluppato nei primi anni '70 presso l'**IBM Research Lab**
  - Il primo vero DBMS Relazionale fu **RDMS**, sviluppato nei primi anni '70 presso il **MIT**
  - seguito da **Ingres** sviluppato nel 1974 all'**Università di Berkeley**

# Modello Logico

Altri **modelli logici** (meno diffusi) sono:

- **Gerarchico** (anni '60) basato sulle **strutture ad albero**
  - Ha caratterizzato i **primi DBMS**
- **Reticolare** (anni '70) basato su **grafi**
  - Estensione e standardizzazione del linguaggio **COBOL**
- **A Oggetti** (anni '80) evoluzione del modello relazionale
  - Estende alle Basi di Dati del paradigma della **programmazione a oggetti**



Esempio di **modello gerarchico**

# Modello Relazionale

Nel Modello Relazionale i dati sono **organizzati in tabelle**

- Un tabella memorizza informazioni su un **insieme di oggetti che hanno identica struttura** (insieme di studenti, libri, film, CD, ...)

Una tabella si compone di:

- **righe** (o **record** o **tuple** o **n-ple**) ognuna corrispondente a un **singolo dato** (un libro, uno studente, un film, un CD, ...)
- **colonne** (o **campi** o **attributi**), corrispondenti a **single informazioni relative ai dati** (es. matricola, titolo, ...)

Matricola	Cognome	Nome	Data di Nascita
276545	Rossi	Maria	25/11/1971
485745	Neri	Anna	23/04/1972
200768	Verdi	Fabio	12/02/1972
587614	Rossi	Luca	10/10/1971
937653	Bruni	Mario	01/12/1971

Esempio di **tabella** con informazioni su una **popolazione di studenti**

# Modello Relazionale

Le tabelle rappresentano **relazioni matematiche** tra insiemi di oggetti:  **$R (A_1, A_2, \dots, A_n)$**

- **R** è il nome della relazione (**tabella**)
- **$A_1, A_2, \dots, A_n$**  sono gli insiemi posti in relazione (**campi**)
- **n** è detto **grado** della relazione (numero di campi)

## Esempio

**Studenti (Matricola, Cognome, Nome, Data di Nascita)**

- **Matricola**: insieme di tutte le matricole esistenti
- **Cognome**: insieme di tutti i cognomi esistenti
- **Nome**: insieme di tutti i nomi esistenti
- **Data di Nascita**: insieme di tutte le date esistenti

# Modello Relazionale

Un **elenco di oggetti che soddisfa la relazione** si dice **tupla** (corrisponde ad una **riga** della tabella)

## Esempi:

- (276545, Rossi, Maria, 25/11/1971)
- (485745, Neri, Anna, 23/04/1972)
- ...
- Il **numero delle tuple** esistenti per una relazione si chiama **cardinalità** della relazione
- l'**insieme delle tuple esistenti** per una data relazione (in un dato istante) si chiama **istanza** della relazione (**tabella**)

# Modello Relazionale

Un database è tipicamente composto da **più tabelle** collegate tra di loro tramite **valori di campi comuni**

## CORSI

Codice	Nome corso	Docente
01	Analisi	Giani
03	Chimica	Melli
04	Chimica	Belli

## STUDENTI

Matricola	Cognome	Nome	Data di Nascita
276545	Rossi	Maria	25/11/1971
485745	Neri	Anna	23/04/1972
200768	Verdi	Fabio	12/02/1972
587614	Rossi	Luca	10/10/1971
937653	Bruni	Mario	01/12/1971

## ESAMI

Studente	Voto	Corso
276545	28	01
276545	27	04
937653	25	01
200768	24	04

# Modello Relazionale

**Esempio:** La carriera della studentessa **Maria Rossi** è ricostruita con i dati provenienti da **tre tabelle**

## CORSI

Codice	Nome corso	Docente
01	Analisi	Giani
03	Chimica	Melli
04	Chimica	Belli

## STUDENTI

Matricola	Cognome	Nome	Data di Nascita
276545	Rossi	Maria	25/11/1971
485745	Neri	Anna	23/04/1972
200768	Verdi	Fabio	12/02/1972
587614	Rossi	Luca	10/10/1971
937653	Bruni	Mario	01/12/1971

## ESAMI

Studente	Voto	Corso
276545	28	01
276545	27	04
937653	25	01
200768	24	04



# Modello Relazionale

**Esempio:** La carriera della studentessa **Maria Rossi** è ricostruita con i dati provenienti da **tre tabelle**

## CORSI

Codice	Nome corso	Docente
01	Analisi	Giani
03	Chimica	Melli
04	Chimica	Belli

## STUDENTI

Matricola	Cognome	Nome	Data di Nascita
276545	Rossi	Maria	25/11/1971
485745	Neri	Anna	23/04/1972
200768	Verdi	Fabio	12/02/1972
587614	Rossi	Luca	10/10/1971
937653	Bruni	Mario	01/12/1971

## ESAMI

Studente	Voto	Corso
276545	28	01
276545	27	04
937653	25	01
200768	24	04

# Modello Relazionale

**Esempio:** La carriera della studentessa **Maria Rossi** è ricostruita con i dati provenienti da **tre tabelle**

## CORSI

Codice	Nome corso	Docente
01	Analisi	Giani
03	Chimica	Melli
04	Chimica	Belli

## STUDENTI

Matricola	Cognome	Nome	Data di Nascita
276545	Rossi	Maria	25/11/1971
485745	Neri	Anna	23/04/1972
200768	Verdi	Fabio	12/02/1972
587614	Rossi	Luca	10/10/1971
937653	Bruni	Mario	01/12/1971

## ESAMI

Studente	Voto	Corso
276545	28	01
276545	27	04
937653	25	01
200768	24	04

# Modello Relazionale

**Esempio:** La carriera della studentessa **Maria Rossi** è ricostruita con i dati provenienti da **tre tabelle**

## CORSI

Codice	Nome corso	Docente
01	Analisi	Giani
03	Chimica	Melli
04	Chimica	Belli

## STUDENTI

Matricola	Cognome	Nome	Data di Nascita
276545	Rossi	Maria	25/11/1971
485745	Neri	Anna	23/04/1972
200768	Verdi	Fabio	12/02/1972
587614	Rossi	Luca	10/10/1971
937653	Bruni	Mario	01/12/1971

## ESAMI

Studente	Voto	Corso
276545	28	01
276545	27	04
937653	25	01
200768	24	04

# Modello Relazionale

**Esempio:** La carriera della studentessa **Maria Rossi** è ricostruita con i dati provenienti da **tre tabelle**

## CORSI

Codice	Nome corso	Docente
01	Analisi	Giani
03	Chimica	Melli
04	Chimica	Belli

## STUDENTI

Matricola	Cognome	Nome	Data di Nascita
276545	Rossi	Maria	25/11/1971
485745	Neri	Anna	23/04/1972
200768	Verdi	Fabio	12/02/1972
587614	Rossi	Luca	10/10/1971
937653	Bruni	Mario	01/12/1971

## ESAMI

Studente	Voto	Corso
276545	28	01
276545	27	04
937653	25	01
200768	24	04

# Chiave Primaria

Nel Modello Relazionale ogni riga di una tabella deve essere **diversa** dalle altre

- È necessario individuare uno o più **campi che identificano univocamente le righe**
- Tali campi vengono detti **chiavi primarie** della tabella

## Esempi:

La **matricola** identifica univocamente uno **studente**

### STUDENTI

Matricola	Cognome	Nome	Data di Nascita
276545	Rossi	Maria	25/11/1971
485745	Neri	Anna	23/04/1972
200768	Verdi	Fabio	12/02/1972
587614	Rossi	Luca	10/10/1971
937653	Bruni	Mario	01/12/1971



# Chiave Primaria

Nel Modello Relazionale ogni riga di una tabella deve essere **diversa** dalle altre

- È necessario individuare uno o più **campi che identificano univocamente le righe**
- Tali campi vengono detti **chiavi primarie** della tabella

## Esempi:

Il **codice** identifica univocamente un **corso**

### CORSI

Codice	Nome corso	Docente
01	Analisi	Giani
03	Chimica	Melli
04	Chimica	Belli



# Chiave Primaria

Nel Modello Relazionale **ogni riga di una tabella deve essere diversa dalle altre**

- È necessario individuare uno o più **campi che identificano univocamente le righe**
- Tali campi vengono detti **chiave primarie** della tabella

## Esempi:

- La **matricola dello studente + il codice del corso** identificano univocamente un **esame**

## ESAMI

Studente	Voto	Corso
276545	28	01
276545	27	04
937653	25	01
200768	24	04



# Relazioni tra Tabelle

In un Database Relazionale **una tabella è già una relazione tra i campi** che la compongono

- Tuttavia spesso è necessario creare **relazioni anche tra dati memorizzati in tabelle diverse**
- Tali relazioni sono caratterizzate dal **numero di entità coinvolte** per ogni archivio
  - **Relazioni uno-a-uno:** ad ogni record della prima tabella può corrispondere **al più un record** della seconda tabella e viceversa
  - **Relazione uno-a-molti:** ad un record della prima tabella possono corrispondere **più record della seconda** mentre ad ogni record della seconda è associato **un solo record della prima**
  - **Relazioni molti-a-molti:** ad ogni record della prima tabella possono corrispondere **più record della seconda e viceversa**

# Relazioni tra Tabelle

**Relazioni uno-a-uno:** ad ogni record della prima tabella può corrispondere **al più un record** della seconda tabella e viceversa

- Le due tabelle devono avere la stessa chiave primaria

## Esempio

La relazione che collega **uno studente con il suo voto di laurea**

### STUDENTI

Matricola	Cognome	Nome	Data di Nascita
276545	Rossi	Maria	25/11/1971
485745	Neri	Anna	23/04/1972
200768	Verdi	Fabio	12/02/1972
587614	Rossi	Luca	10/10/1971
937653	Bruni	Mario	01/12/1971



### ESAMI DI LAUREA

Studente	Voto	Lode
276545	110	S
485745	90	N
587614	100	N



# Relazioni tra Tabelle

**Relazioni uno-a-uno:** ad ogni record della prima tabella può corrispondere **al più un record** della seconda tabella e viceversa

- Le due tabelle devono avere la stessa chiave primaria

## Esempio

La relazione che collega **uno studente con il suo voto di laurea**

### STUDENTI

Matricola	Cognome	Nome	Data di Nascita
276545	Rossi	Maria	25/11/1971
485745	Neri	Anna	23/04/1972
200768	Verdi	Fabio	12/02/1972
587614	Rossi	Luca	10/10/1971
937653	Bruni	Mario	01/12/1971



### ESAMI DI LAUREA

Studente	Voto	Lode
276545	110	S
485745	90	N
587614	100	N



# Relazioni tra Tabelle

**Relazioni uno-a-uno:** ad ogni record della prima tabella può corrispondere **al più un record** della seconda tabella e viceversa

- Le due tabelle devono avere la stessa chiave primaria

## Esempio

La relazione che collega **uno studente con il suo voto di laurea**

### STUDENTI

Matricola	Cognome	Nome	Data di Nascita
276545	Rossi	Maria	25/11/1971
485745	Neri	Anna	23/04/1972
200768	Verdi	Fabio	12/02/1972
587614	Rossi	Luca	10/10/1971
937653	Bruni	Mario	01/12/1971

### ESAMI DI LAUREA

Studente	Voto	Lode
276545	110	S
485745	90	N
587614	100	N



# Relazioni tra Tabelle

**Relazioni uno-a-uno:** ad ogni record della prima tabella può corrispondere **al più un record** della seconda tabella e viceversa

- Le due tabelle devono avere la stessa chiave primaria

## Esempio

La relazione che collega **uno studente con il suo voto di laurea**

### STUDENTI

Matricola	Cognome	Nome	Data di Nascita
276545	Rossi	Maria	25/11/1971
485745	Neri	Anna	23/04/1972
200768	Verdi	Fabio	12/02/1972
587614	Rossi	Luca	10/10/1971
937653	Bruni	Mario	01/12/1971

### ESAMI DI LAUREA

Studente	Voto	Lode
276545	110	S
485745	90	N
587614	100	N



# Relazioni tra Tabelle

**Relazioni uno-a-molti:** ad un record della prima tabella possono corrispondere **più record della seconda** mentre ad ogni record della seconda è associato **al più un record della prima**

- Alla seconda tabella viene aggiunto il campo chiave della prima tabella

## Esempio

La relazione che collega un corso di laurea con gli studenti iscritti

### STUDENTI

Matricola	Cognome	Nome	Codice CDL
276545	Rossi	Maria	01
485745	Neri	Anna	01
200768	Verdi	Fabio	02
587614	Rossi	Luca	02
937653	Bruni	Mario	03



### CORSI DI LAUREA

Codice CDL	Nome CDL
01	Economia
02	Informatica
03	Matematica



# Relazioni tra Tabelle

**Relazioni uno-a-molti:** ad un record della prima tabella possono corrispondere **più record della seconda** mentre ad ogni record della seconda è associato **al più un record della prima**

- Alla seconda tabella viene aggiunto il campo chiave della prima tabella

## Esempio

La relazione che collega un corso di laurea con gli studenti iscritti

### STUDENTI

Matricola	Cognome	Nome	Codice CDL
276545	Rossi	Maria	01
485745	Neri	Anna	01
200768	Verdi	Fabio	02
587614	Rossi	Luca	02
937653	Bruni	Mario	03



### CORSI DI LAUREA

Codice CDL	Nome CDL
01	Economia
02	Informatica
03	Matematica



# Relazioni tra Tabelle

**Relazioni uno-a-molti:** ad un record della prima tabella possono corrispondere **più record della seconda** mentre ad ogni record della seconda è associato **al più un record della prima**

- Alla seconda tabella viene aggiunto il campo chiave della prima tabella

## Esempio

La relazione che collega un corso di laurea con gli studenti iscritti

### STUDENTI

Matricola	Cognome	Nome	Codice CDL
276545	Rossi	Maria	01
485745	Neri	Anna	01
200768	Verdi	Fabio	02
587614	Rossi	Luca	02
937653	Bruni	Mario	03

### CORSI DI LAUREA

Codice CDL	Nome CDL
01	Economia
02	Informatica
03	Matematica



# Relazioni tra Tabelle

**Relazioni uno-a-molti:** ad un record della prima tabella possono corrispondere **più record della seconda** mentre ad ogni record della seconda è associato **al più un record della prima**

- Alla seconda tabella viene aggiunto il campo chiave della prima tabella

## Esempio

La relazione che collega un corso di laurea con gli studenti iscritti

### STUDENTI

Matricola	Cognome	Nome	Codice CDL
276545	Rossi	Maria	01
485745	Neri	Anna	01
200768	Verdi	Fabio	02
587614	Rossi	Luca	02
937653	Bruni	Mario	03

### CORSI DI LAUREA

Codice CDL	Nome CDL
01	Economia
02	Informatica
03	Matematica





# Relazioni tra Tabelle

**Relazioni multi-a-multi:** ad ogni record della prima tabella possono corrispondere **più record della seconda e viceversa**

- Viene realizzata attraverso un **tabella di join** che contiene i **campi chiave delle tabelle in relazione**

## Esempio

La relazione che collega **uno studente con i corsi completati**

### STUDENTI

Matricola	Cognome	Nome	Data di Nascita
276545	Rossi	Maria	25/11/1971
485745	Neri	Anna	23/04/1972
200768	Verdi	Fabio	12/02/1972
587614	Rossi	Luca	10/10/1971
937653	Bruni	Mario	01/12/1971

### CORSI

Codice	Nome corso	Docente
01	Analisi	Giani
03	Chimica	Melli
04	Chimica	Belli

### ESAMI

Studente	Voto	Corso
276545	28	01
276545	27	04
937653	25	01
200768	24	04

# Relazioni tra Tabelle

**Relazioni multi-a-multi:** ad ogni record della prima tabella possono corrispondere **più record della seconda e viceversa**

- Viene realizzata attraverso un **tabella di join** che contiene i **campi chiave delle tabelle in relazione**

## Esempio

La relazione che collega **uno studente con i corsi completati**

### STUDENTI

Matricola	Cognome	Nome	Data di Nascita
276545	Rossi	Maria	25/11/1971
485745	Neri	Anna	23/04/1972
200768	Verdi	Fabio	12/02/1972
587614	Rossi	Luca	10/10/1971
937653	Bruni	Mario	01/12/1971

### CORSI

Codice	Nome corso	Docente
01	Analisi	Giani
03	Chimica	Melli
04	Chimica	Belli

### ESAMI

Studente	Voto	Corso
276545	28	01
276545	27	04
937653	25	01
200768	24	04

# Relazioni tra Tabelle

**Relazioni multi-a-multi:** ad ogni record della prima tabella possono corrispondere **più record della seconda e viceversa**

- Viene realizzata attraverso un **tabella di join** che contiene i **campi chiave** delle tabelle in relazione

## Esempio

La relazione che collega **uno studente con i corsi completati**

### STUDENTI

Matricola	Cognome	Nome	Data di Nascita
276545	Rossi	Maria	25/11/1971
485745	Neri	Anna	23/04/1972
200768	Verdi	Fabio	12/02/1972
587614	Rossi	Luca	10/10/1971
937653	Bruni	Mario	01/12/1971

### CORSI

Codice	Nome corso	Docente
01	Analisi	Giani
03	Chimica	Melli
04	Chimica	Belli

### ESAMI

Studente	Voto	Corso
276545	28	01
276545	27	04
937653	25	01
200768	24	04

# Relazioni tra Tabelle

**Relazioni multi-a-multi:** ad ogni record della prima tabella possono corrispondere **più record della seconda e viceversa**

- Viene realizzata attraverso un **tabella di join** che contiene i **campi chiave** delle tabelle in relazione

## Esempio

La relazione che collega **uno studente con i corsi completati**

### STUDENTI

Matricola	Cognome	Nome	Data di Nascita
276545	Rossi	Maria	25/11/1971
485745	Neri	Anna	23/04/1972
200768	Verdi	Fabio	12/02/1972
587614	Rossi	Luca	10/10/1971
937653	Bruni	Mario	01/12/1971

### CORSI

Codice	Nome corso	Docente
01	Analisi	Giani
03	Chimica	Melli
04	Chimica	Belli

### ESAMI

Studente	Voto	Corso
276545	28	01
276545	27	04
937653	25	01
200768	24	04

# Relazioni tra Tabelle

Relazioni **multi-a-multi**: ad ogni record della prima tabella possono corrispondere **più record della seconda e viceversa**

- Viene realizzata attraverso un **tabella di join** che contiene i **campi chiave delle tabelle in relazione**

## Esempio

La relazione che collega **uno studente con i corsi completati**

### STUDENTI

Matricola	Cognome	Nome	Data di Nascita
276545	Rossi	Maria	25/11/1971
485745	Neri	Anna	23/04/1972
200768	Verdi	Fabio	12/02/1972
587614	Rossi	Luca	10/10/1971
937653	Bruni	Mario	01/12/1971

### CORSI

Codice	Nome corso	Docente
01	Analisi	Giani
03	Chimica	Melli
04	Chimica	Belli

### ESAMI

Studente	Voto	Corso
276545	28	01
276545	27	04
937653	25	01
200768	24	04

# Interrogare un Database

Una **tabella** può contenere **centinaia di campi** (colonne) e **milioni di record** (righe)

- In genere un utente è interessato solo ad una **piccola parte** di tutta questa informazione
- Esempi**: i CD di un certo artista, i voti di un certo studente, ecc.

Una **Query** (interrogazione) è una frase scritta in un linguaggio speciale che **permette di estrarre** da una o più tabelle **solo i dati che interessano**

- Una Query **crea una tabella più piccola** che contiene solo i dati che interessano



# Interrogare un Database

**Esempio.**  
data la **tabella**:

**Corsi**

Corso	Docente	Aula
Basi di dati	Rossi	DS3
Sistemi	Neri	N3
Reti	Bruni	N3
Controlli	Bruni	G

L'interrogazione "nome dei corsi che si svolgono nell'aula N3 e dei relativi docenti" produce il seguente risultato:

Corso	Docente
Sistemi	Neri
Reti	Bruni

# Interrogare un Database

Le principali modalità di interrogazione di un DB sono le seguenti:

- **QBE Query by Example** (interfaccia grafica)
  - Modalità **semplice e diretta**
  - Adatta per **interrogazioni semplici**
  - Modalità non standard, spesso legata al DBMS
- **SQL Structured Query Language** (linguaggio)
  - Modalità **più complessa** della precedente
  - Valida per **tutti i tipi di interrogazione**
  - Linguaggio universale, adottato da tutti i DBMS



# Interrogare un Database

Esempio di interrogazione **QBE**:

Campo:	Corso	Docente	Aula	
Tabella:	Corsi	Corsi	Corsi	
Ordinamento:				
Mostra:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Criteri:			"N3"	
Oppure:				

**Corsi**

Corso	Docente	Aula
Basi di dati	Rossi	DS3
Sistemi	Neri	N3
Reti	Bruni	N3
Controlli	Bruni	G

**Risultato**

Corso	Docente
Sistemi	Neri
Reti	Bruni

# Interrogare un Database

Esempio di interrogazione **SQL**:

```
SELECT Corso, Docente  
FROM Corsi  
WHERE Aula = "N3"
```

**Corsi**

Corso	Docente	Aula
Basi di dati	Rossi	DS3
Sistemi	Neri	N3
Reti	Bruni	N3
Controlli	Bruni	G

**Risultato**

Corso	Docente
Sistemi	Neri
Reti	Bruni

# Interrogare un Database

**Sintassi** di base di una query SQL:

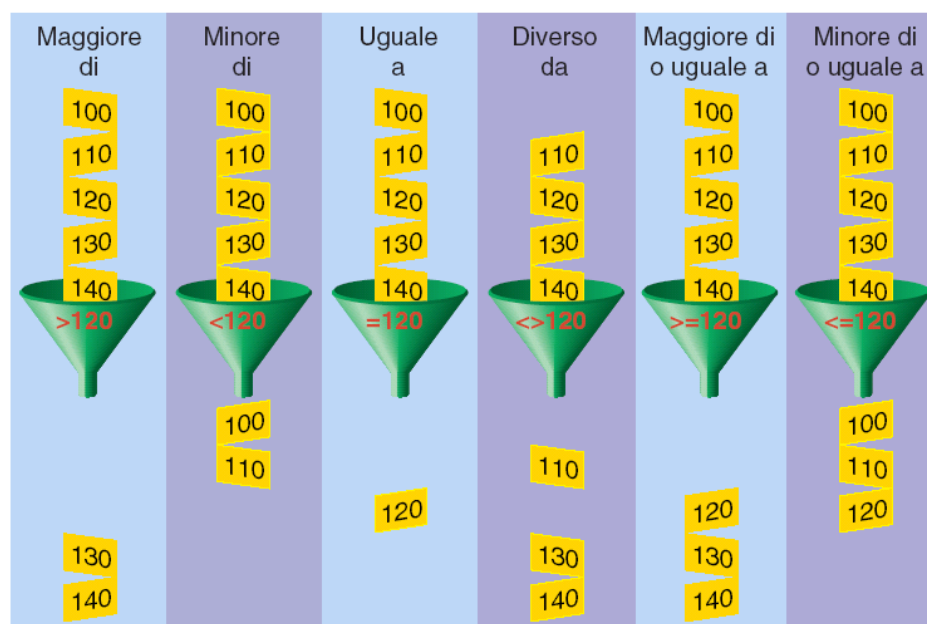
```
SELECT [nome campi separati da una virgola, * indica tutti i campi]  
FROM [nome tabella]  
WHERE [condizioni di ricerca]
```

Perché una query riesca ad **estrarre solo i dati che interessano** occorre specificare delle **condizioni di ricerca** che sono **soddisfatte dai record che interessano**

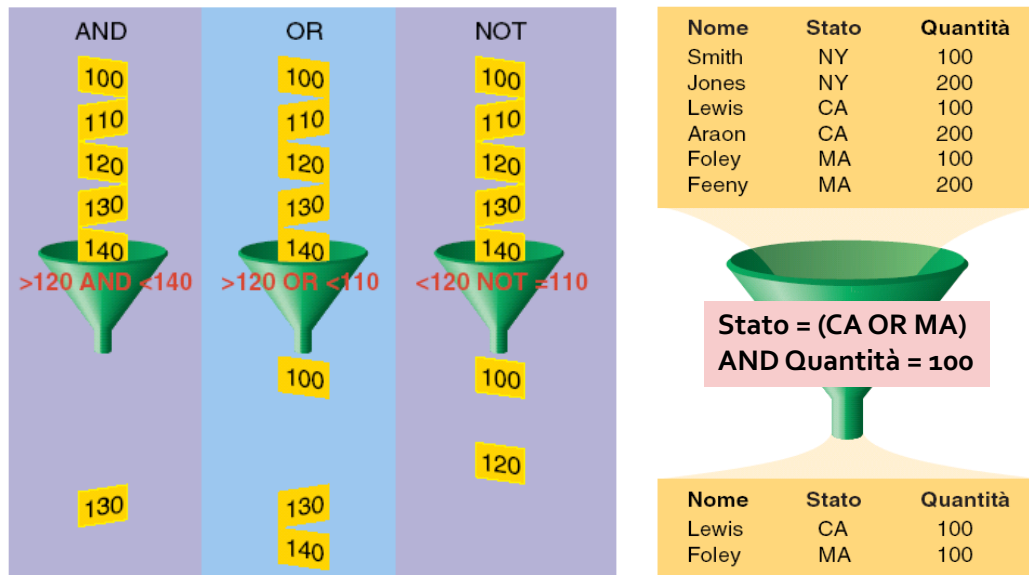
Le **condizioni di ricerca** possono essere espresse tramite:

- operatori relazionali: =, <, >, <=, >=, <> (diverso)
- operatori logici: AND, OR, NOT

# Interrogare un Database



# Interrogare un Database



# Esempi di Interrogazione

## STUDENTI

Matricola	Cognome	Nome	Data di Nascita
276545	Rossi	Maria	25/11/1971
485745	Neri	Anna	23/04/1972
200768	Verdi	Fabio	12/02/1972
587614	Rossi	Luca	10/10/1971
937653	Bruni	Mario	01/12/1971

Ottenere tutte le informazioni sullo **studente** con **matricola 485745**

```
SELECT *
FROM STUDENTI
WHERE Matricola = 485745
```

### Risultato

Matricola	Cognome	Nome	Data di Nascita
485745	Neri	Anna	23/04/1972

# Esempi di Interrogazione

## STUDENTI

Matricola	Cognome	Nome	Data di Nascita
276545	Rossi	Maria	25/11/1971
485745	Neri	Anna	23/04/1972
200768	Verdi	Fabio	12/02/1972
587614	Rossi	Luca	10/10/1971
937653	Bruni	Mario	01/12/1971

Ottenere nome, cognome e matricola di tutti gli **studenti nati nel 1972**

```
SELECT Nome, Cognome, Matricola
FROM STUDENTI
WHERE [Data di Nascita] >= #01/01/1972# AND [Data di Nascita] <= #31/12/1972#
```

Risultato:

Nome	Cognome	Matricola
Anna	Neri	485745
Fabio	Verdi	200768

## Bibliografia

Considerando anche le lezioni precedenti...

- **Cap. 1:** fino a 1.4
- **Cap. 2:** tutto tranne 2.6, 2.7 e 2.17
- **Cap. 3:** fino a 3.13 tranne 3.5
- **Cap. 4:** fino a 4.6
- **Cap. 5:** fino a 5.6 tranne 5.4
- **Cap. 6:** 6.1, 6.2, 6.7, 6.8, 6.12
- **Cap. 11:** fino a 11.6 tranne 11.3
- **Cap. 14:** fino a 14.5 tranne 14.4
- Approfondimenti su queste slide

