

Corso di Programmazione 1 a.a.2006/2007

Prof.ssa Chiara Petrioli
Corso di Laurea in Informatica
Università degli Studi “La Sapienza”

Benvenuti !

- Una breve introduzione al corso di laurea ed agli obiettivi di questo corso...

Come è strutturata l'Università “La Sapienza” ...



Facoltà e Dipartimenti

Offre i
Docenti
Per il
Corso di
Laurea

Facoltà di Scienze
Matematiche Fisiche
E Naturali

**Dipartimento di
Informatica**

Dip. di Biologia
Cellulare...

Dipartimento di
Matematica

Dipartimento
di Fisica

Dipartimento
di Chimica

.....

RICERCA

**Corso di laurea
In Informatica**

Corso di laurea
In Biologia

Corso di laurea
In Matematica

Corso di laurea
In Fisica

Corso di laurea
In Chimica

Corsi di laurea
In Scienze Naturali
e geologiche

DIDATTICA

Dipartimento e Corso di Laurea in Informatica

- Direttore del Dipartimento di Informatica:
Prof.ssa Rossella Petreschi

(il Dipartimento è responsabile delle strutture –inclusi laboratori, biblioteca-, gestisce le risorse del dipartimento, pianifica le assunzioni di nuovi docenti e nuovo personale)

- Presidente del Corso di Laurea in Informatica:
Prof.ssa Emanuela Fachini

(il corso di laurea decide ad esempio il calendario didattico, pianifica i corsi, stabilisce le regole per i corsi di laurea,...)

Commissioni

■ Dipartimento

- Commissione Scientifica
- Commissione Biblioteca
- Commissione Finanziamenti (Petrioli)

...

■ Corso di Laurea

- Commissione piani di Studio (Tronci)
- Commissione didattica (Prof. Bongiovanni)

....

Perché “La Sapienza”

- Elencata come migliore università italiana in tutti i ranking internazionali (che tengono conto della riuscita degli studenti, dell'eccellenza scientifica dei docenti)
 - 150.000 studenti ← gestione non facile ma titolo spendibile anche in ambito internazionale

Perché la Facoltà di Scienze

- Metodo scientifico di analisi, capacità di problem solving vi aiuteranno nelle vostre esperienze lavorative
- Tra tutte le Facoltà di S.M.F.N. italiane la Facoltà della “Sapienza” risulta prima nei ranking internazionali. Secondo il Ranking delle migliori Università del mondo stilato annualmente dal ‘The Times’ è 27° nel mondo!!
 - Cercate di sfruttare la possibilità di interagire con docenti di eccellenza

Perché Informatica

- **Non solo programmazione....**
- **Alcuni ambiti in cui gli informatici hanno avuto un ruolo importante negli ultimi 5-10 anni:**
 - **Industria del SW (programmatore, capi progetto, manager,...)**
 - **Industria cinematografica ed entertainment (grafica, effetti speciali, video giochi,...)**
 - **Reti di calcolatori (progettista di reti di calcolatori, amministratore di reti, programmatore di rete, manager, progettista di nuove tecnologie nei lab di ricerca industriali, progettista di sistemi di monitoraggio ambientale, ...)**
 - **Sistemi (amministratore di sistema, progettista di nuovi S.O., esperto di sicurezza,...)**
 - **Industria automobilistica**
 - **Ambiti biologici e medici (informatica medica, biologia computazionale)**
 -
- **e tra l'altro...**
 - **Secondo la valutazione CIVR il gruppo misto di Information e Communications Technologies (docenti di Informatica, Ingegneria Informatica, Telecomunicazioni) migliore a livello nazionale**
 - **Più di un terzo dei docenti con esperienza (dottorato/postdoc) in istituzioni estere di eccellenza**

Organizzazione del Corso di Laurea

- Prevede un misto di corsi di Informatica teorica, di Matematica/Fisica, di Programmazione.... PERCHE'?
 - Un esempio: progettista di reti che lavora in TIM
 - Deve conoscere le reti radiomobili (architettura di Internet, programmazione di rete, lab. di programmazione di rete, **reti avanzate, progettazione di sistemi wireless, progettazione di rete**)
 - Si occupa di gestire reti cellulari: pianificazione di reti cellulari (valutazione delle prestazioni, basata su calcolo delle probabilità ← a sua volta richiede competenze di analisi matematica), deve capire come avvenga la trasmissione sul mezzo radio (fisica), deve poter progettare soluzioni per la comunicazione di informazioni nella rete (protocolli di rete ← serve una conoscenza delle reti e degli algoritmi distribuiti), deve poter implementare le soluzioni che ha sviluppato (programmazione) e verificare che funzionino correttamente.
- ... SERVE QUINDI UNA FIGURA CHE ABBIA TUTTE LE COMPETENZE PREVISTE DAL CORSO DI LAUREA (Un misto di conoscenze matematiche/fisiche e informatiche servono anche per gli altri profili lavorativi ☺)..ulteriori esempi nel seguito...

Organizzazione del Corso di Laurea

■ Alcuni messaggi

- **Partite con l'obiettivo di conseguire la laurea Specialistica**
 - Preparazione dopo 3 o 5 anni molto diversa (e le industrie che vi possono offrire i posti più interessanti lo riconoscono)
- **Lavorate su voi stessi e sulle vostre capacità**
 - Sfruttate tutte le occasioni per incrementare le vostre conoscenze su argomenti d'avanguardia (interazioni con i docenti, seminari, workshop organizzati), mettetevi in competizione con voi stessi (problem solving, gare di informatica), entrate in un contesto internazionale;
 - Andate al di là di quanto sentite in classe (il corso è una guida –fondamentale e da seguire- allo studio, studiare sui libri di testo e sui libri di approfondimento è fondamentale per ottenere buoni voti e per una preparazione approfondita)
 - Acquisite un buon metodo di studio, imparate l'inglese
 - L'ingresso nel mondo del lavoro non differenzia moltissimo tra 3 e 3+2, ma a 3,5,10 anni dalla laurea una buona preparazione universitaria fa la differenza in termini di livello di interesse del lavoro, ruolo (manageriale o meno) e stipendio
 - Se potete dedicatevi full time allo studio (le materie sono tante, non banali, avere la possibilità di acquisire un metodo di raggiungere una comprensione approfondita ed una elevata capacità di problem solving –cosa che richiede tempo e impegno- è ciò che farà la differenza per il vostro successo futuro)

Prog 1: Introduzione

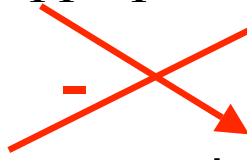
- Consideriamo un problema semplice, ovvero la risoluzione di un sistema di equazioni lineari in x_1 e x_2 con a_{11} , a_{12} , a_{21} , a_{22} , b_1 e b_2 costanti a cui è stato assegnato un valore.
- Se il determinante $a_{11}a_{22}-a_{21}a_{12} \neq 0$ le soluzioni sono date da

$$x_1 = \frac{b_1 a_{22} - b_2 a_{12}}{a_{11} a_{22} - a_{21} a_{12}}, x_2 = -\frac{b_1 a_{21} - b_2 a_{11}}{a_{11} a_{22} - a_{21} a_{12}}$$

- Altrimenti il sistema è malcondizionato o inconsistente

Prog 1: Introduzione

- Considera il sistema di equazioni lineari in x_1 e x_2 con $a_{11}, a_{12}, a_{21}, a_{22}$ assegnati

$$\begin{cases} a_{11}x_1 + a_{12}x_2 = b_1 \\ a_{21}x_1 + a_{22}x_2 = b_2 \end{cases}$$


ovvero la
lineari in x_1 e
ti a cui è stato

- Se il determinante $a_{11}a_{22} - a_{21}a_{12} \neq 0$ le soluzioni sono date da

$$x_1 = \frac{b_1a_{22} - b_2a_{12}}{a_{11}a_{22} - a_{21}a_{12}}, x_2 = -\frac{b_1a_{21} - b_2a_{11}}{a_{11}a_{22} - a_{21}a_{12}}$$

- Altrimenti il sistema è malcondizionato o inconsistente

Risoluzione di un sistema di equazioni lineari (algoritmo)

■ Un algoritmo è una sequenza di passi elementari che devono essere eseguiti, in un determinato ordine, per risolvere un problema.

■ Nel nostro caso:

– Passo 1: Calcolare la quantità $\Delta = a_{11}a_{22} - a_{21}a_{12}$

– Passo 2: Se $\Delta=0$ e $\frac{a_{11}}{a_{21}} \neq \frac{b_1}{b_2}$ il sistema è inconsistente.
(non ammette soluzioni)

Se invece $\frac{a_{11}}{a_{21}} = \frac{b_1}{b_2}$ e $a_{11} \neq 0$ allora $x_1 = \frac{b_1}{a_{11}} - x_2 \left(\frac{a_{12}}{a_{11}} \right)$

Se $a_{11}=0$ invece va posto $x_2 = -\frac{(b_1a_{22} - b_2a_{12})}{\Delta}$

Risoluzione di un sistema di equazioni lineari (algoritmo)

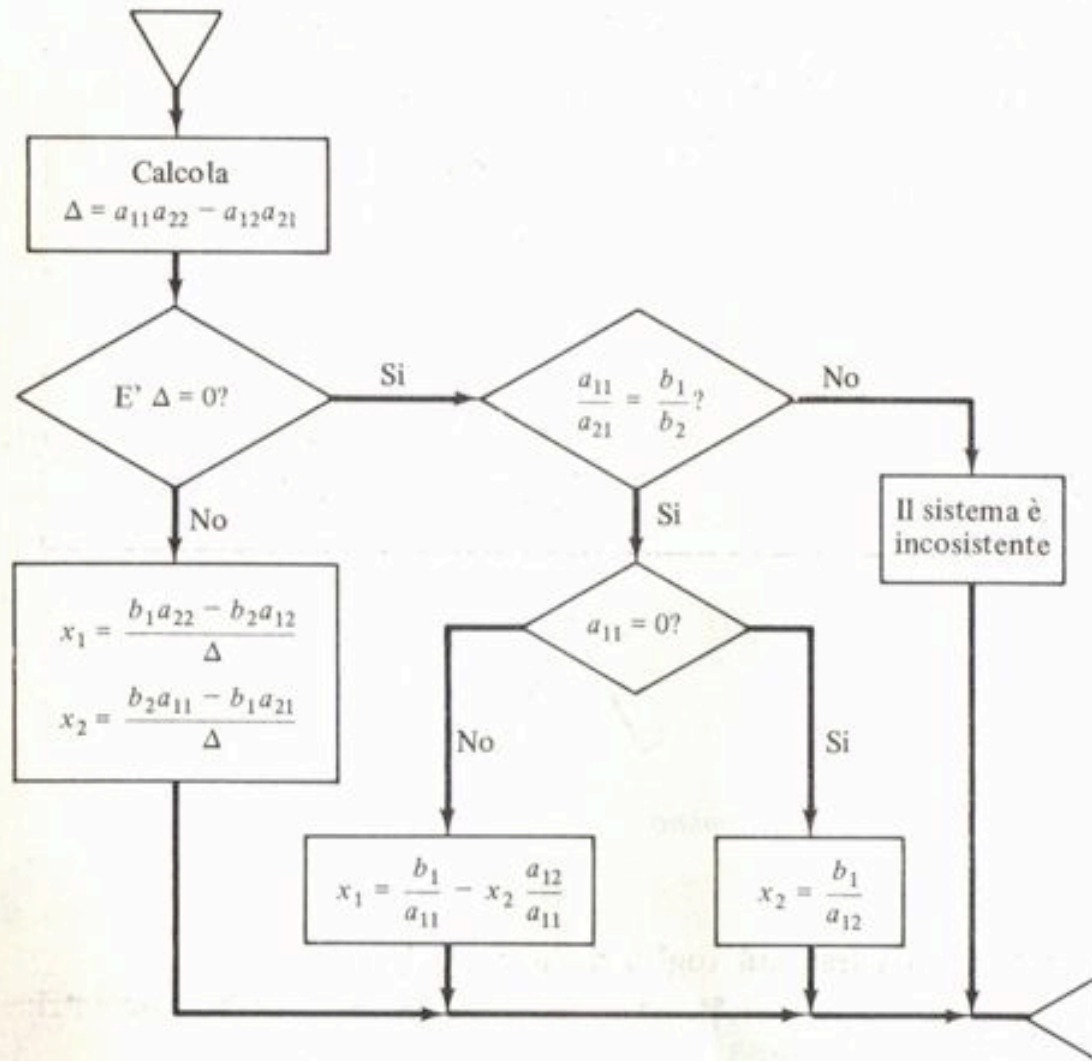
- Un algoritmo è una sequenza di passi elementari che devono essere eseguiti, in un determinato ordine, per risolvere un problema.
- Nel nostro caso:

- ...

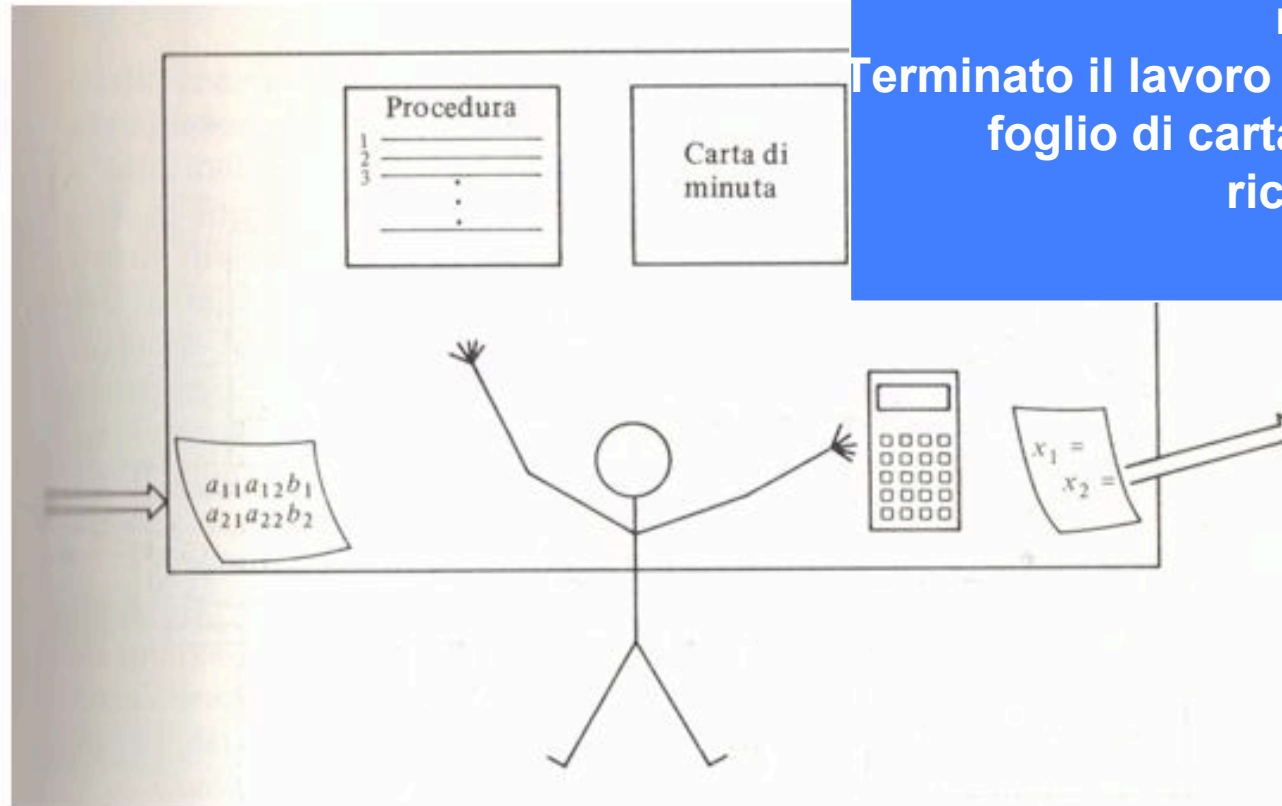
- Passo 3: Se $\Delta \neq 0$ il sistema ammette un'unica soluzione.

$$x_1 = \frac{b_1 a_{22} - b_2 a_{12}}{a_{11} a_{22} - a_{21} a_{12}}, x_2 = -\frac{b_1 a_{21} - b_2 a_{11}}{a_{11} a_{22} - a_{21} a_{12}}$$

Risoluzione di un sistema di equazioni lineari (diagramma di flusso)



Risoluzione di un sistema lineare (esecuzione)



Attività:

- leggere un passo sul foglio della procedura
- eseguire il passo, eventualmente facendo uso della calcolatrice e della carta di minuta

Terminato il lavoro scrivere il risultato su un foglio di carta da consegnare al richiedente

Possibile automatizzare l'intera procedura. Per capire come vediamo quali sono le componenti di un calcolatore...

Elaboratore

- Un elaboratore è un dispositivo capace di effettuare calcoli e di prendere decisioni logiche ad una velocità milioni/miliardi di volte più veloce di quella di un essere umano
- Gli elaboratori processano dati sotto il controllo di insiemi di istruzioni (i programmi) che specificano i passi elementari da effettuare sui dati per ottenere il risultato voluto.
- I programmi sono scritti in linguaggi di programmazione.

Elaboratore

- Un elaboratore è un dispositivo capace di effettuare calcoli e di prendere decisioni logiche ad una velocità milioni/miliardi di volte più veloce di quella di un essere umano

- G 15.7 milioni di secondi in un anno! La vita di una persona circa 80 anni
1.2 miliardi di secondi.

controllo di insiemi di istruzioni (i programmi)
che specificano i passi elementari da
effettuare sui dati per ottenere il risultato

Considerando che per 1/3 del tempo dormiamo un calcolatore è in grado
di calcolare in un secondo più di una persona in una vita

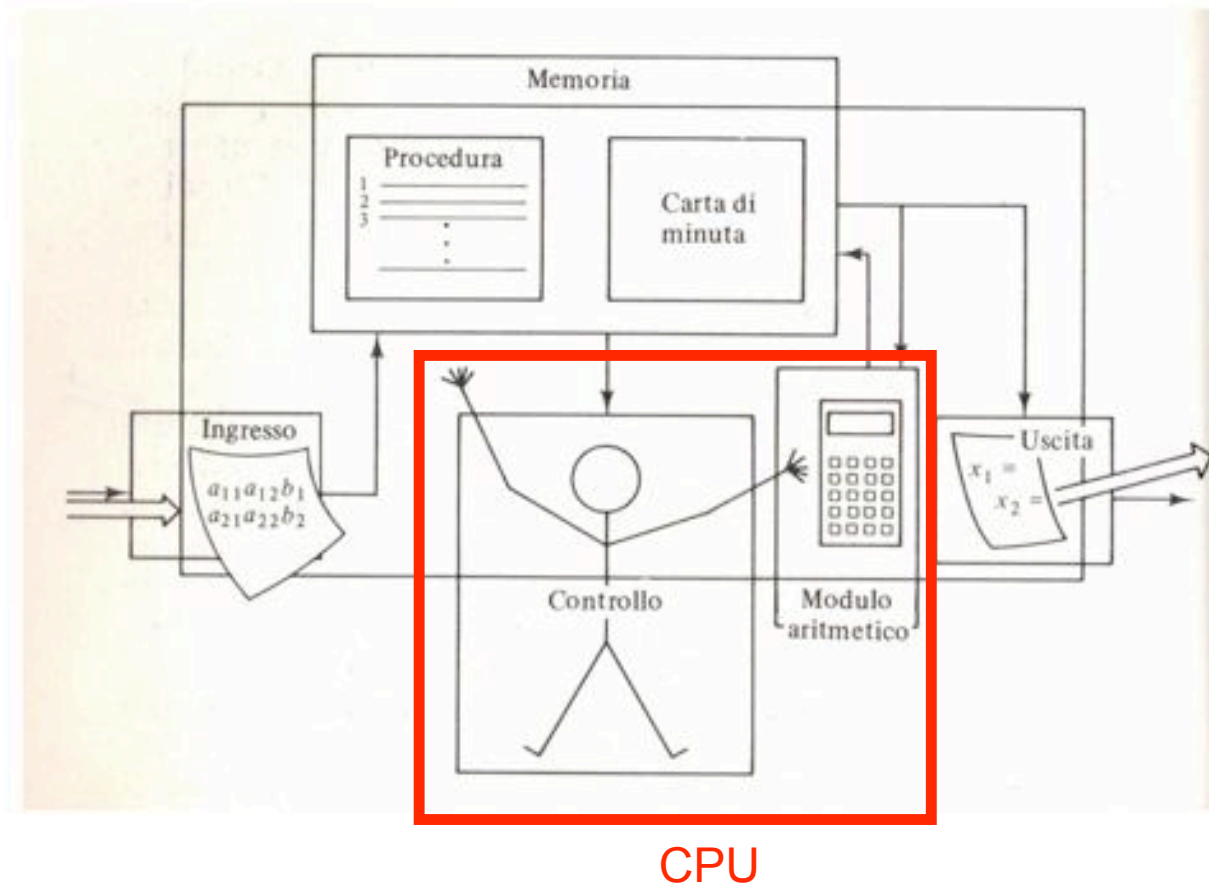
- I programmi sono scritti in linguaggi di

In questo corso affronteremo, dato un problema, 1) come determinare la sequenza di passi elementari che devono essere eseguiti per risolvere il problema, 2) come codificare tale sequenza in C, 3) come verificare che la soluzione trovata e la sua codifica siano corrette.

Componenti di un elaboratore

- Un computer consiste di un
 - processore
 - Controlla le operazioni del computer ed effettua il processing dei dati
 - memoria
 - Memorizza dati e programmi
 - dispositivi di I/O
 - Consentono la comunicazione tra il calcolatore e l'utente (Input: mouse, tastiera,...; Output: schermo, stampante,...)
 - bus di sistema
 - Consentono la comunicazione tra le varie componenti HW dell'elaboratore

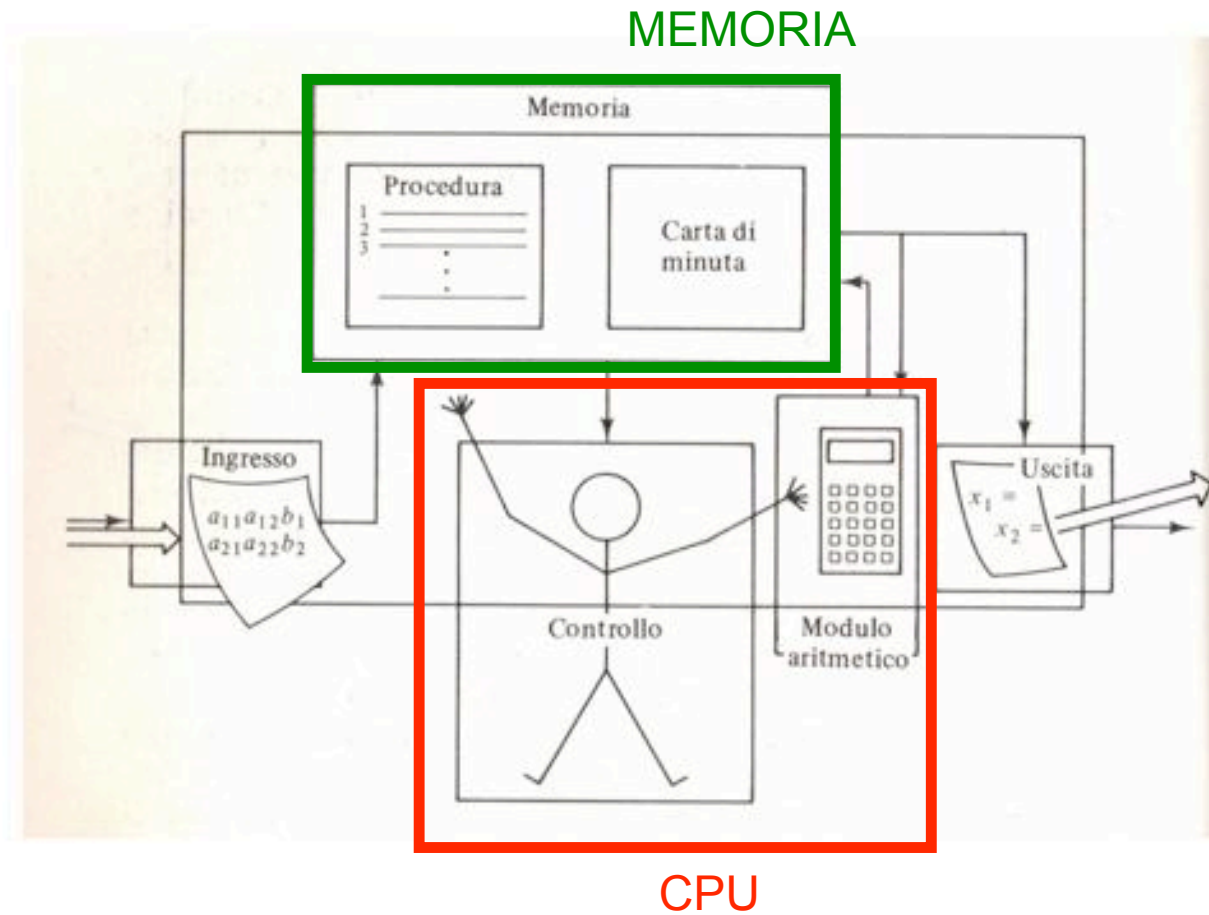
Componenti di un elaboratore



Componenti di un elaboratore

- Un computer consiste di un
 - processore
 - Controlla le operazioni del computer ed effettua il processing dei dati
 - memoria
 - Memorizza dati e programmi
 - dispositivi di I/O
 - Consentono la comunicazione tra il calcolatore e l'utente (Input: mouse, tastiera,...; Output: schermo, stampante,...)
 - bus di sistema
 - Consentono la comunicazione tra le varie componenti HW dell'elaboratore

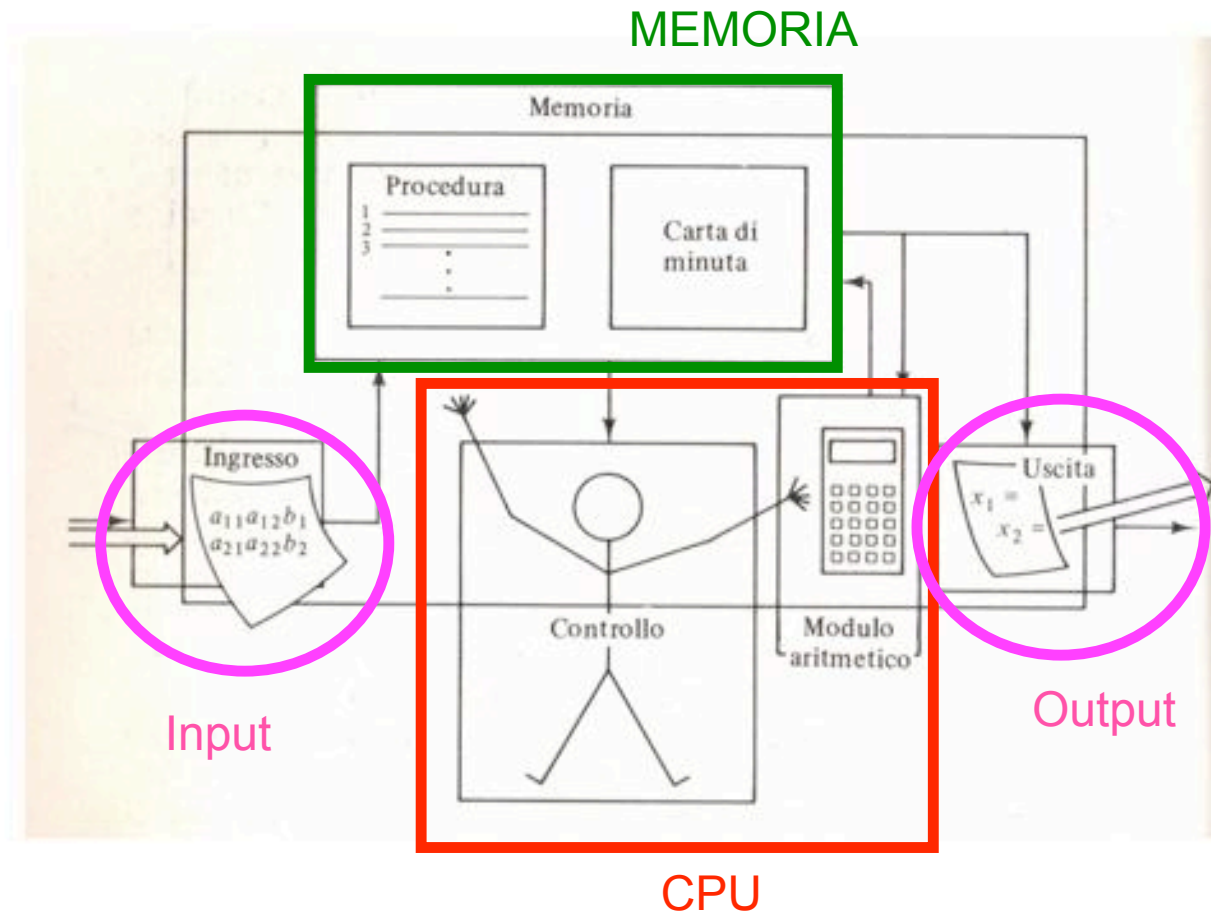
Componenti di un elaboratore



Componenti di un elaboratore

- Un computer consiste di un
 - processore
 - Controlla le operazioni del computer ed effettua il processing dei dati
 - memoria
 - Memorizza dati e programmi
 - dispositivi di I/O
 - Consentono la comunicazione tra il calcolatore e l'utente (Input: mouse, tastiera,...; Output: schermo, stampante,...)
 - bus di sistema
 - Consentono la comunicazione tra le varie componenti HW dell'elaboratore

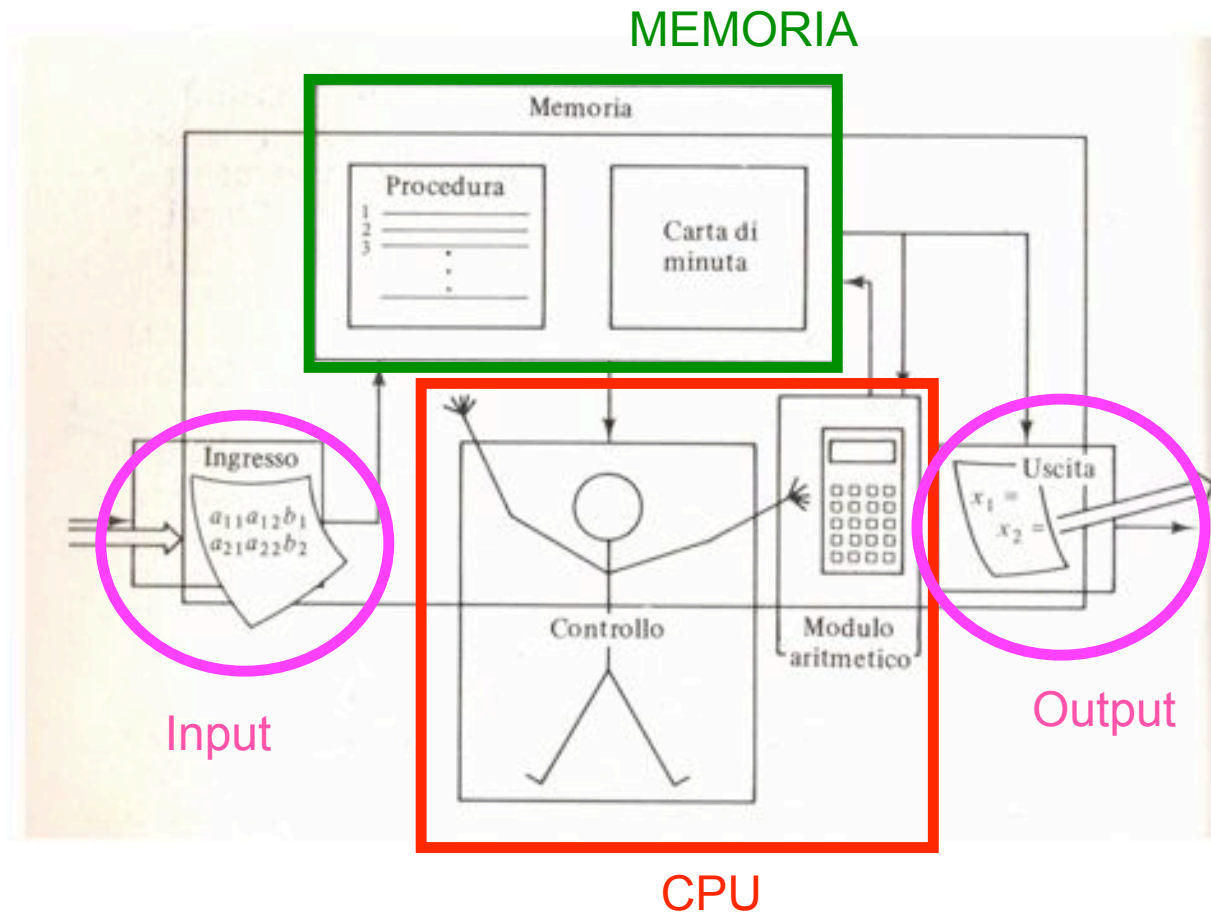
Componenti di un elaboratore



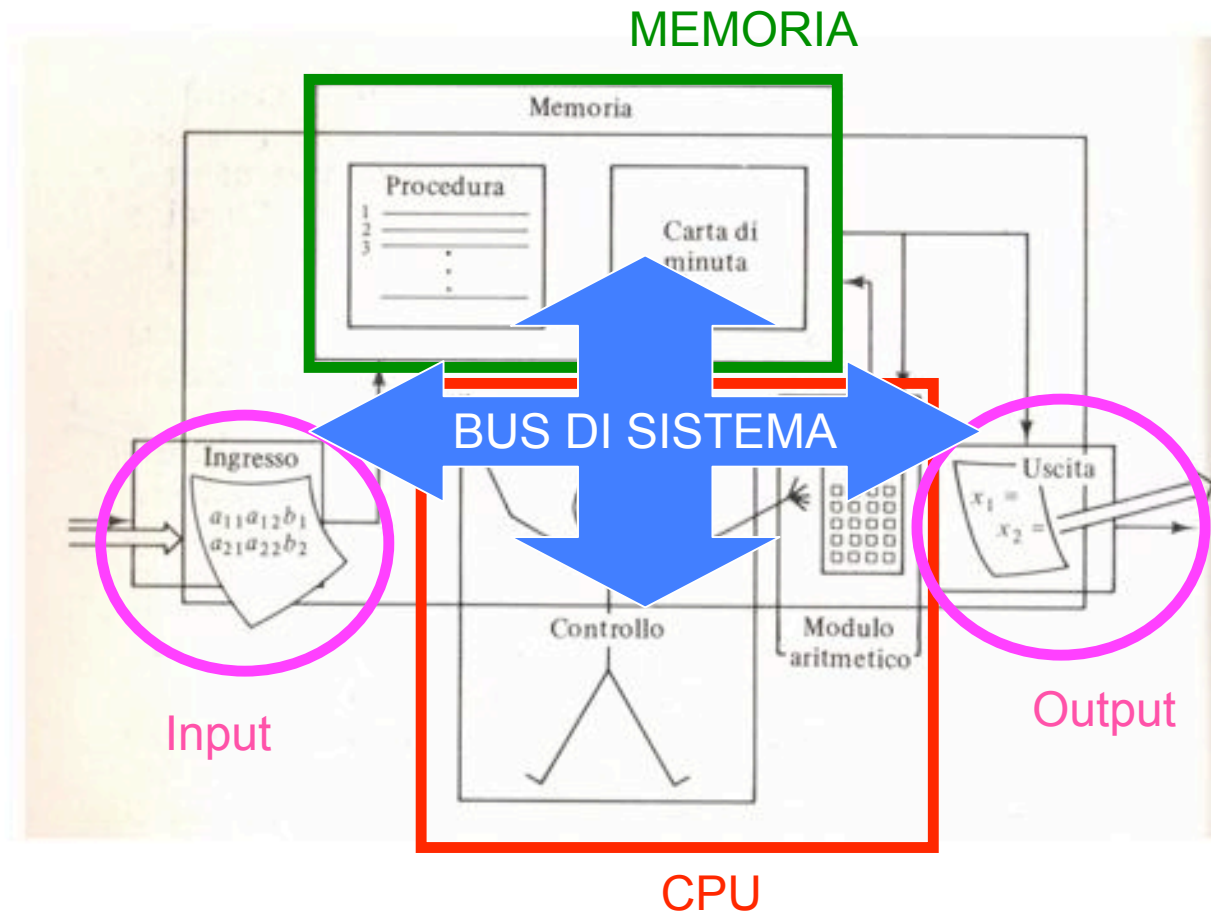
Componenti di un elaboratore

- Un computer consiste di un
 - processore
 - Controlla le operazioni del computer ed effettua il processing dei dati
 - memoria
 - Memorizza dati e programmi
 - dispositivi di I/O
 - Consentono la comunicazione tra il calcolatore e l'utente (Input: mouse, tastiera,...; Output: schermo, stampante,...)
 - bus di sistema
 - Consentono la comunicazione tra le varie componenti HW dell'elaboratore

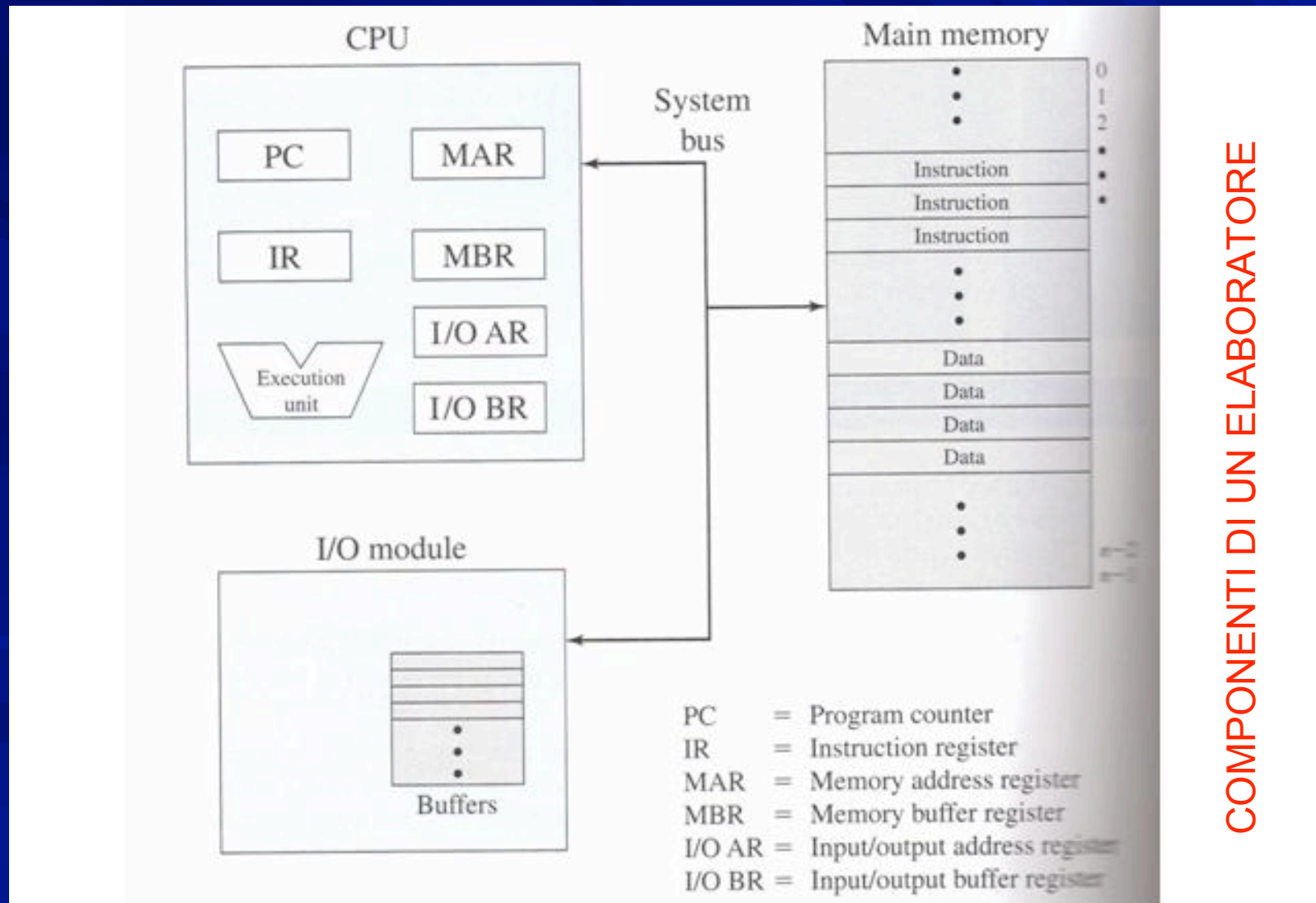
Componenti di un elaboratore



Componenti di un elaboratore

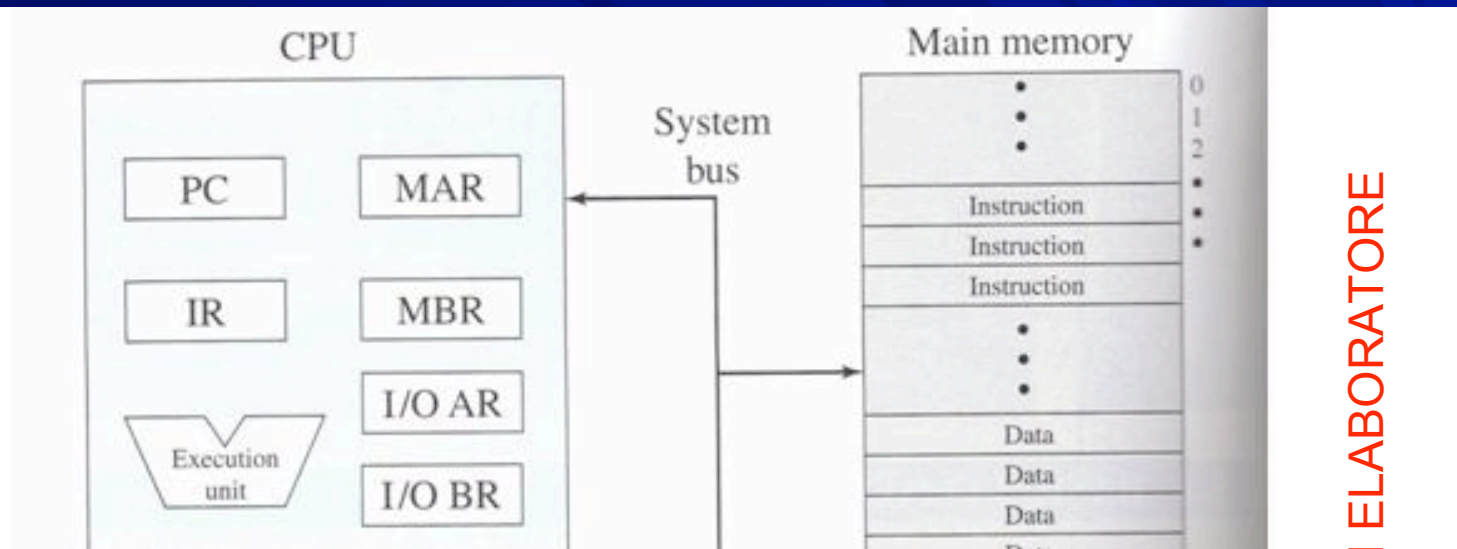


Componenti di un elaboratore



COMPONENTI DI UN ELABORATORE

Componenti di un elaboratore



Un processore include

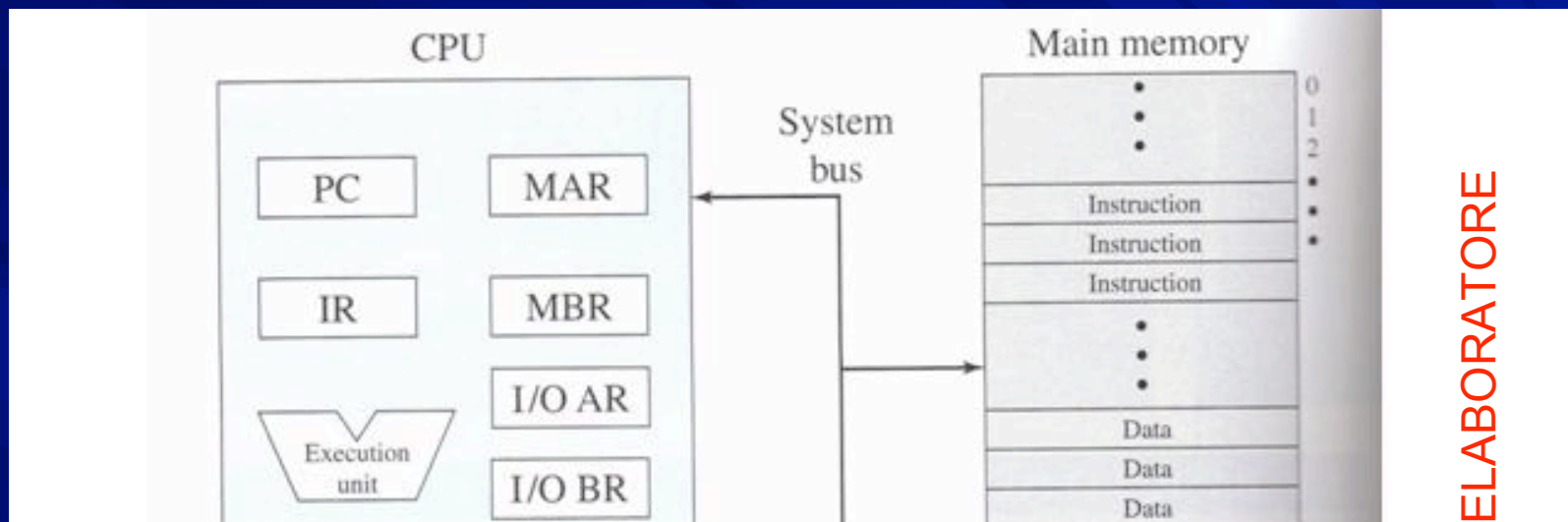
- ALU (Unità Aritmetico-Logica)

Sottosistema che effettua le operazioni sui dati (aritmetiche – addizione, sottrazione,...- e logiche –AND, OR,...-). Ha linee di ingresso (input), di controllo (per specificare il tipo di operazione da effettuare) e linee dati di uscita.

- Capacità di controllo

- Registri (memoria di dimensioni limitate e rapido accesso)

Componenti di un elaboratore

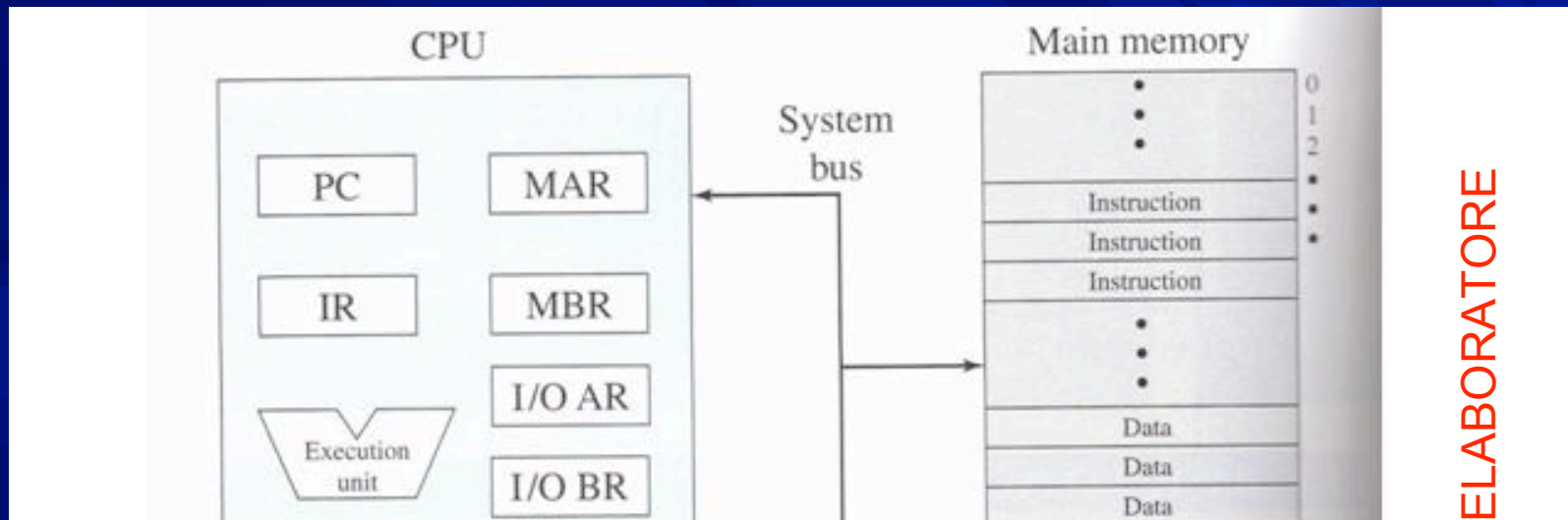


ELABORATORE

Registri:

- Program counter (memorizza l'indirizzo della prossima istruzione del programma da eseguire)
- Instruction Register (memorizza l'ultima istruzione caricata)
- Registri per il trasferimento dati da/verso la memoria
 - *MAR (memory address register): specifica l'indirizzo di memoria da/verso cui avverrà la prossima lettura/scrittura
 - *MBR (memory buffer register): contiene i dati che devono essere scritti in memoria/che vengono letti dalla memoria

Componenti di un elaboratore



Registri:

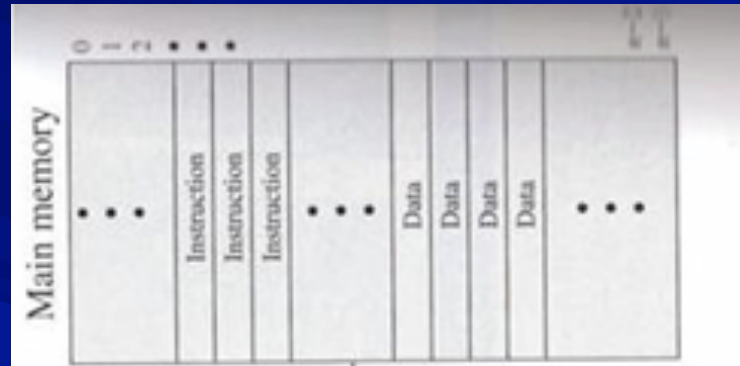
- Program counter (memorizza l'indirizzo della prossima istruzione del programma da eseguire)
- Registri per lo scambio di informazioni con dispositivi di I/O
 - *I/OAR specifica un determinato dispositivo di I/O
 - *I/OBR usato per scambiare dati tra un modulo di I/O e il processore
- Data register
 - Utilizzati per effettuare più velocemente operazioni sui dati

Processore

- Un processore include
 - ALU (Unità Aritmetico-Logica)
 - Sottosistema che effettua le operazioni sui dati (aritmetiche –addizione, sottrazione,...- e logiche –AND, OR,...-). Ha linee di ingresso (input), di controllo (per specificare il tipo di operazione da effettuare) e linee dati di uscita.
 - Capacità di controllo
 - Registri (memoria di dimensioni limitate e rapido accesso)
 - Program counter (memorizza l'indirizzo della prossima istruzione del programma da eseguire)
 - Instruction Register (memorizza l'ultima istruzione caricata)
 - Registri per il trasferimento dati da/verso la memoria
 - MAR (memory address register): specifica l'indirizzo di memoria da/verso cui avverrà la prossima lettura/scrittura
 - MBR (memory buffer register): contiene i dati che devono essere scritti in memoria/che vengono letti dalla memoria
 - Registri per lo scambio di informazioni con dispositivi di I/O
 - I/OAR specifica un determinato dispositivo di I/O
 - I/OBR usato per scambiare dati tra un modulo di I/O e il processore
 - Data register
 - Utilizzati per effettuare più velocemente operazioni sui dati

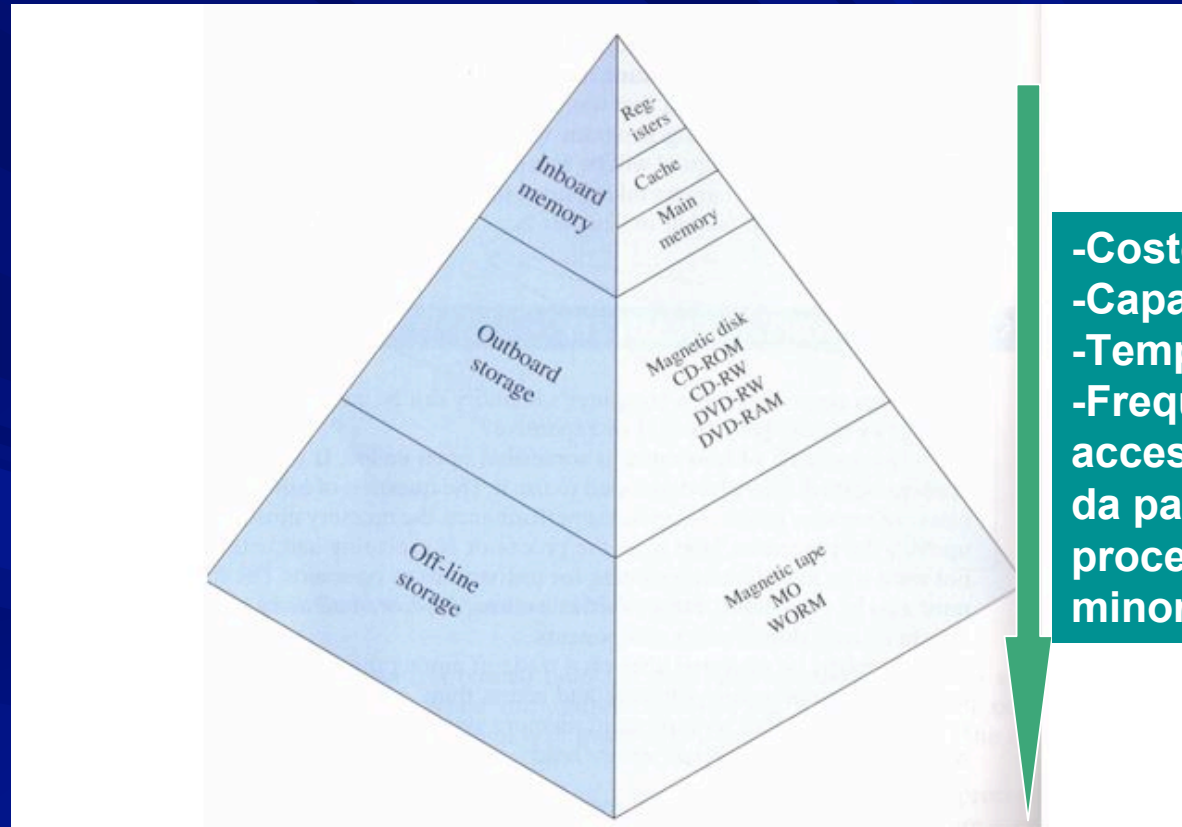
Memoria

- Un modulo di memoria consiste in un insieme di locazioni di memoria. Le locazioni di memoria hanno associati indirizzi crescenti



- Tipicamente la memoria è volatile (il contenuto della memoria viene perso quando viene effettuato uno shut down del computer). La memoria principale o primaria (es. RAM) è volatile. Esiste memoria secondaria (es: disco rigido) che invece è non volatile.
- Esistono diversi tipi di memoria con diversi trade-off tra costo, capacità e tempo di accesso
 - Tempo di accesso più veloce → costo maggiore per bit
 - Capacità maggiore → accesso più lento MA costo più basso

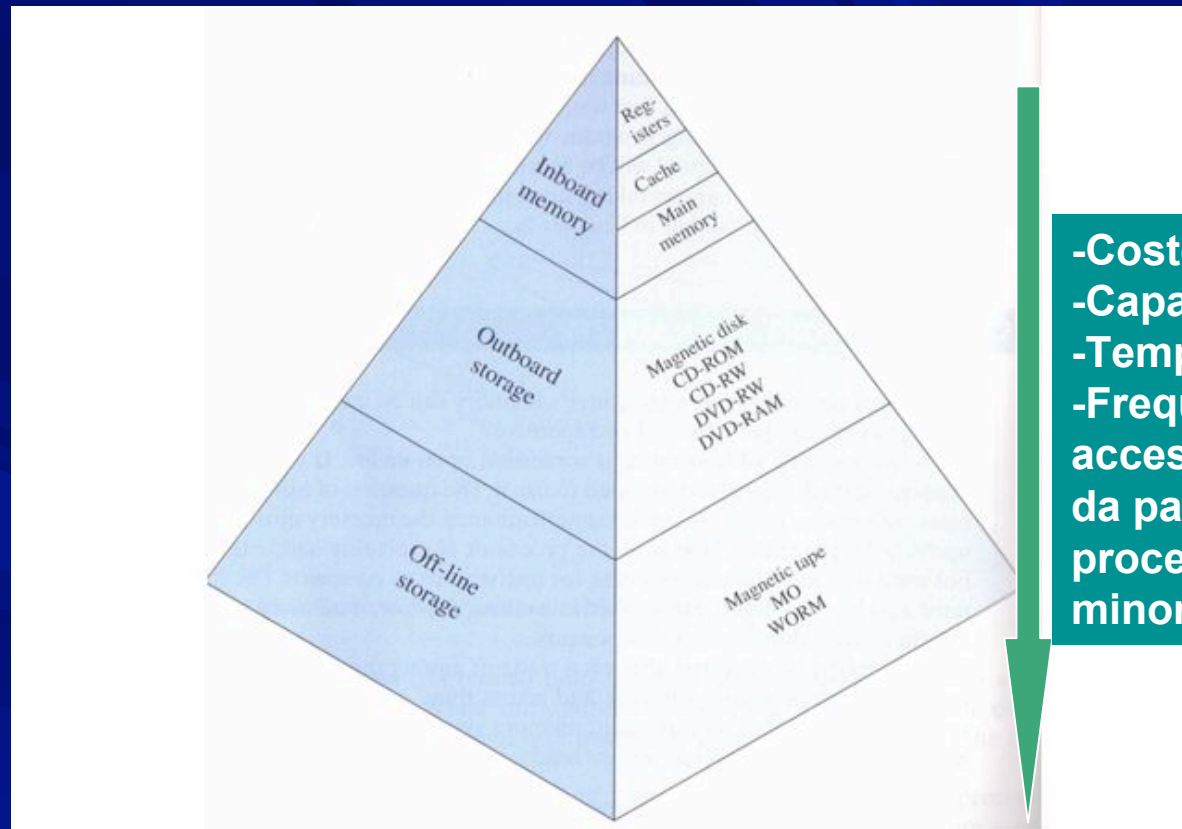
Gerarchia di memoria



-Costo/bit minore
-Capacità >
-Tempo di accesso >
-Frequenza di
accesso
da parte del
processore
minore

■ Si cerca un dato prima in livelli alti della gerarchia (ad esempio in cache). Se il dato si trova nei livelli alti della gerarchia (cosa che avviene una frazione di volte denominata Hit Ratio) allora il tempo di accesso è rapido. Altrimenti si paga il tempo di accesso maggiore associato ai livelli bassi della gerarchia. Segreto: la Hit Ratio deve essere alta

Gerarchia di memoria



-Costo/bit minore
-Capacità >
-Tempo di accesso >
-Frequenza di
accesso
da parte del
processore
minore

M1: Sarà possibile specificare che alcune variabili devono essere memorizzate nei registri. Tuttavia i compilatori di solito sono in grado anche più del programmatore di ottimizzare quali variabili dovrebbero essere memorizzate nei registri.

Gerarchia di memoria

■ Un esempio

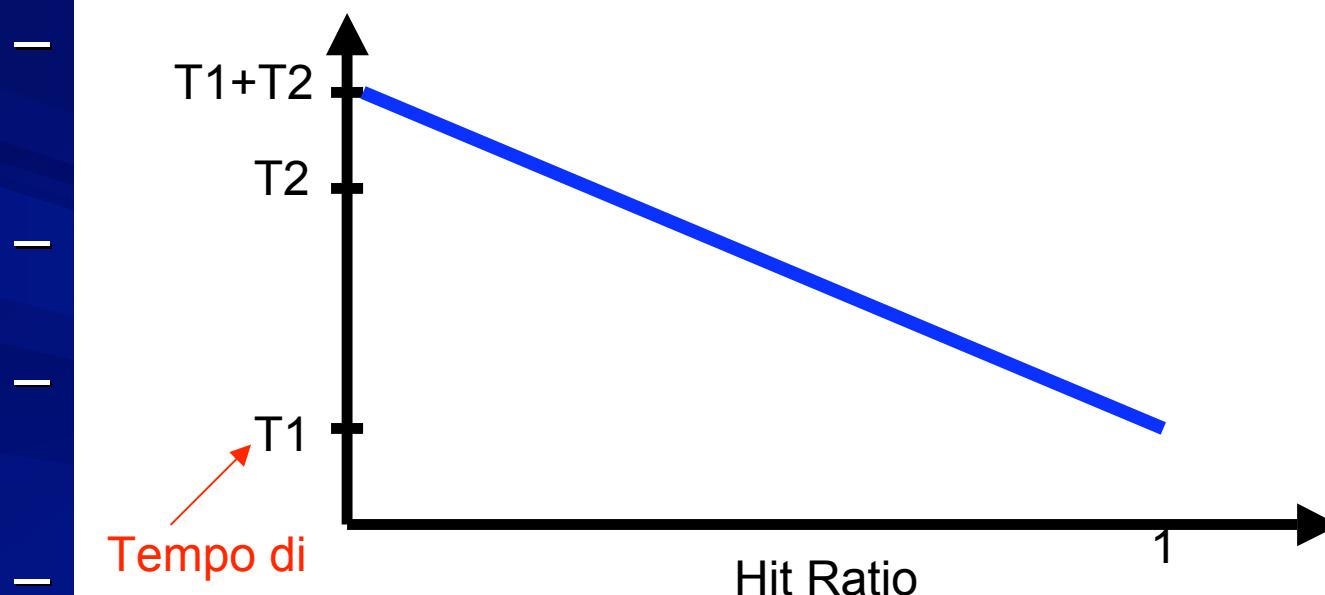
- Supponiamo che ci siano due livelli di memoria. Il primo livello è in grado di memorizzare 1000 bytes ed ha un tempo di accesso di $0.1\mu s$
- Il secondo livello è in grado di memorizzare 100.000 bytes ed ha un tempo di accesso di $1\mu s$
- Il processore cerca prima il dato nel livello 1 di memoria. Se il dato è nel livello 2 si paga un ulteriore ritardo per trasferire dal livello 2 al livello 1.
- Supponiamo che il 95% delle volte il dato venga trovato al livello 1 (Hit Ratio = 0.95). Il tempo medio di accesso è dato da:

$$(0.95)(0.1\mu s) + (0.05)(0.1\mu s + 1\mu s) = 0.15\mu s$$

Gerarchia di memoria

Il segreto è nella Hit Ratio !!

■ Un esempio

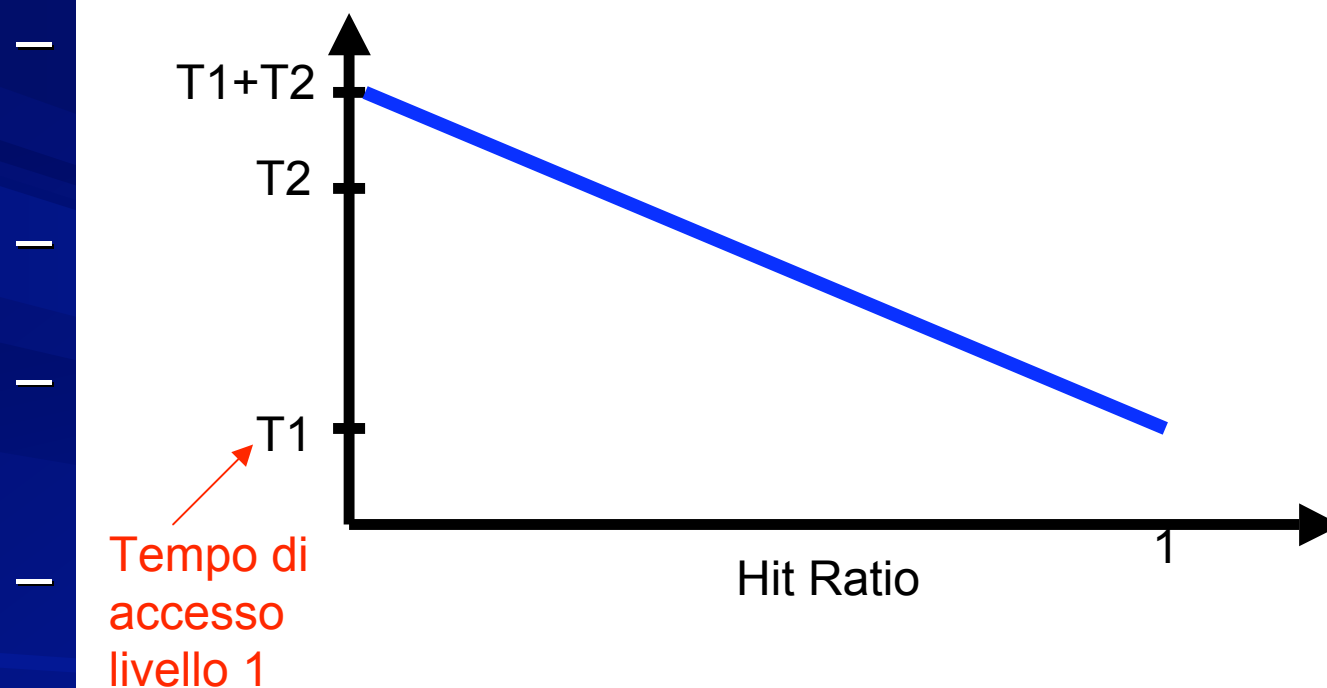


Segreto: principio di località di riferimento ← Durante l'esecuzione di un programma le locazioni di memoria accedute dal processore (dati/istruzioni) tendono ad essere correlate
(
→ È possibile proattivamente passare dati tra i vari livelli della gerarchia in modo che siano nei livelli alti della gerarchia i dati e le istruzioni che con alta probabilità verranno acceduti dal processore in questo momento → hit ratio elevata

Gerarchia di memoria

Il segreto è nella Hit Ratio !!

■ Un esempio

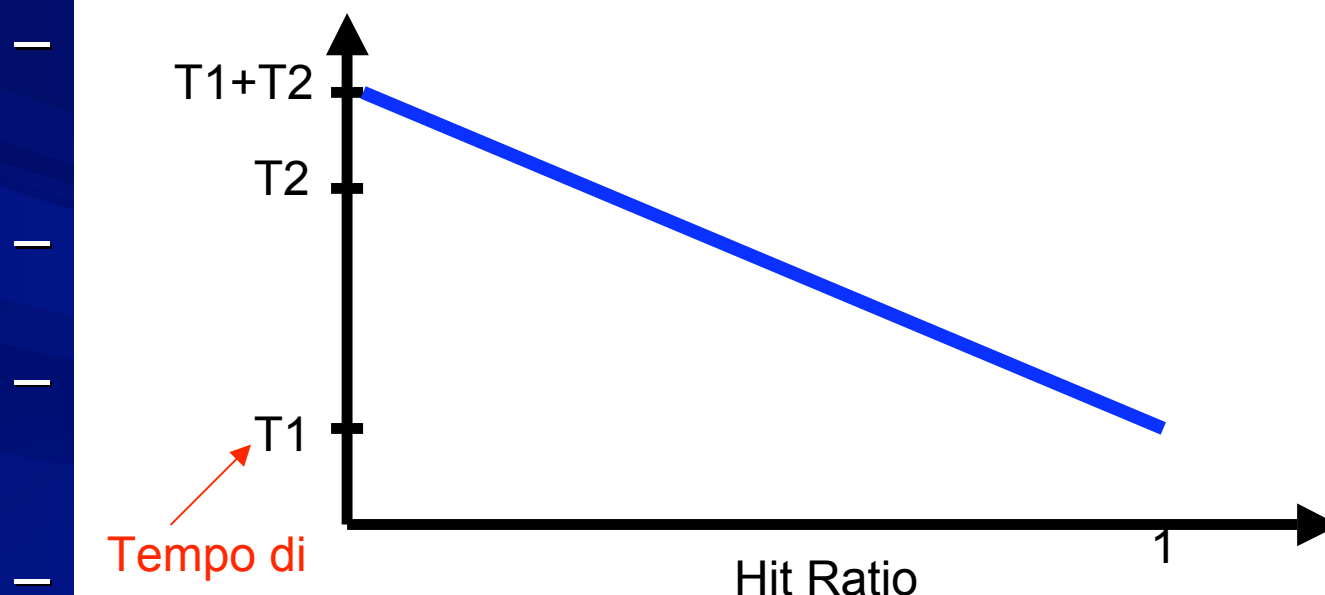


$$(0.95) (0.1\mu s) + (0.05)(0.1\mu s + 1\mu s) = 0.15\mu s$$

Gerarchia di memoria

Il segreto è nella Hit Ratio !!

■ Un esempio



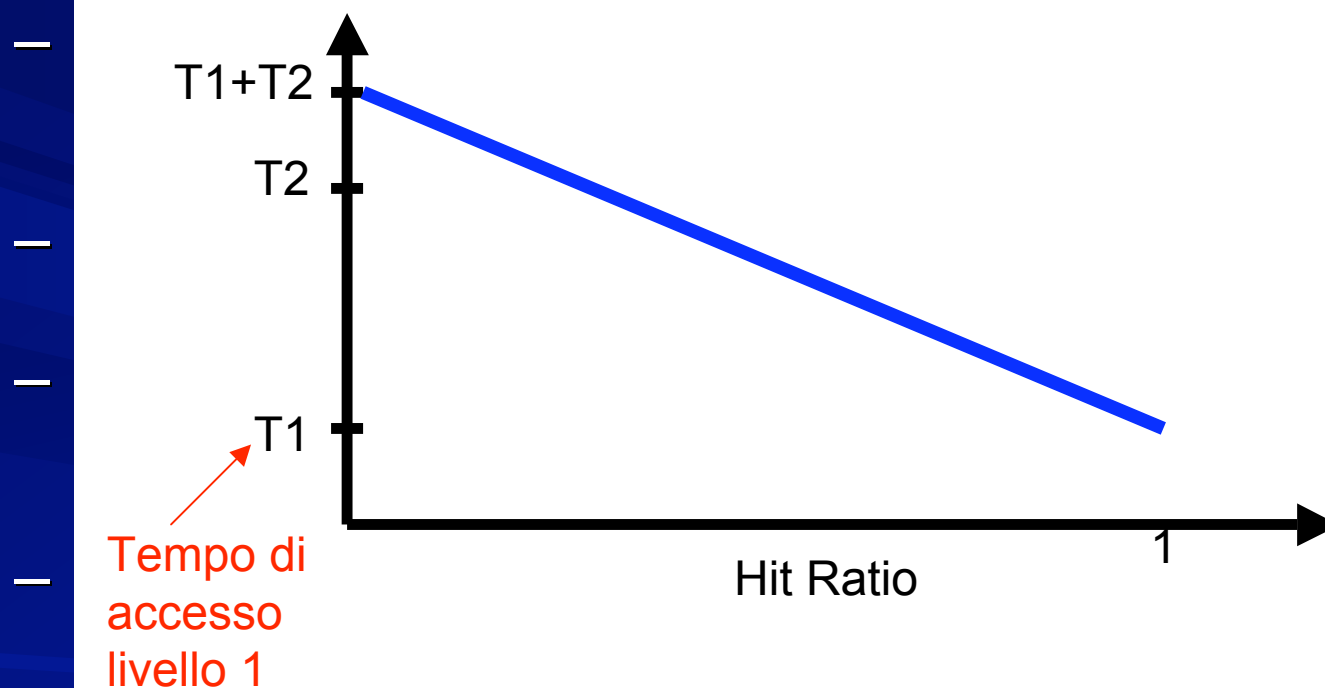
Segreto: principio di località di riferimento ← Durante l'esecuzione di un programma le locazioni di memoria accedute dal processore (dati/istruzioni) tendono ad essere correlate
(
→ È possibile proattivamente passare dati tra i vari livelli della gerarchia in modo che siano nei livelli alti della gerarchia i dati e le istruzioni che con alta probabilità verranno acceduti dal processore in questo momento → hit ratio elevata

esso
oce !

Gerarchia di memoria

Il segreto è nella Hit Ratio !!

■ Un esempio

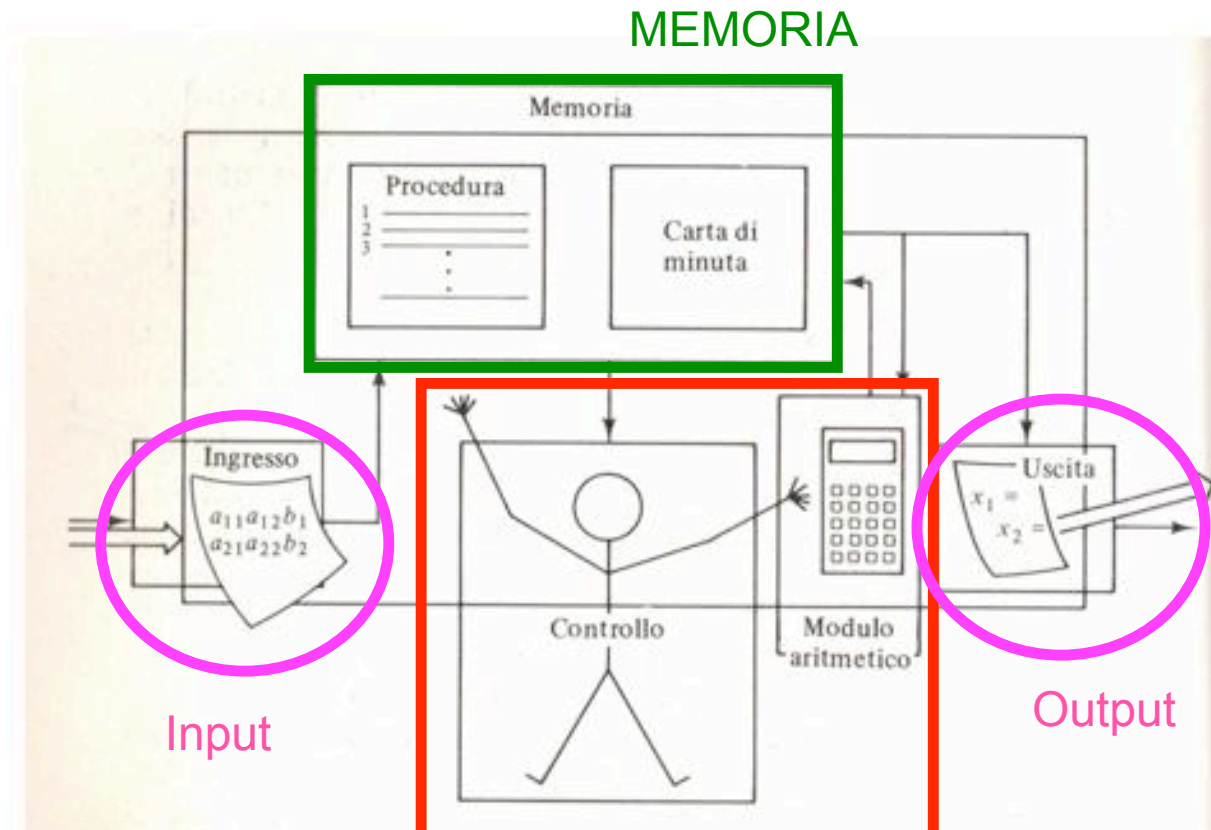


$$(0.95)(0.1\mu s) + (0.05)(0.1\mu s + 1\mu s) = 0.15\mu s$$

Prof. Chiara Petrioli, Corso di Programmazione 1, a.a. 2006/2007

Vicino al tempo di accesso
Della memoria più veloce !

Componenti di un elaboratore



Scriviamo un programma. Il nostro programma deve essere caricato in memoria principale per poter essere eseguito.

Memoria deve poter essere allocata per le variabili usate dal programma (es. x_1, x_2 , e/o variabili temporanee usate per fare conti)

I valori di $a_{11}, a_{12}, \dots$ devono anche essere memorizzate in memoria e essere inserite da input → si comincerà quindi ad eseguire il programma

Esecuzione di una istruzione

- Il processore legge l'istruzione dalla memoria (fetch) /* program counter dice dove trovare la prossima istruzione; Instruction Register conterra' l'istruzione letta*/
- Esegue quindi l'istruzione
 - consideriamo il caso di un computer semplificato in cui ogni istruzione occupa una parola di memoria (16 bit)
 - PC (program counter) ha memorizzato il valore 300
 - Processore contiene un unico registro dei dati chiamato accumulatore, di 16 bit (sia dati che istruzioni lunghi per ipotesi 16 bit)
 - Formato delle istruzioni: 4 bit per tipo di istruzione (opcode), 12 bit restanti per altre info per poter eseguire l'istruzione

Esecuzione di una istruzione

- Il processore legge l'istruzione dalla memoria (fetch) /* program counter dice dove trovare la prossima istruzione; Instruction Register conterra' l'istruzione letta*/
- Esegue quindi l'istruzione
 - consideriamo il caso di un computer semplificato in cui ogni istruzione occupa una parola di memoria (16 bit)
 - PC (program counter) ha memorizzato il valore 300
 - Processore contiene un unico registro dei dati chiamato accumulatore, di 16 bit (sia dati che istruzioni lunghi per ipotesi 16 bit)
 - Formato delle istruzioni: 4 bit per tipo di istruzione (opcode), 12 bit restanti per altre info per poter eseguire l'istruzione

Esecuzione di una istruzione

- Il processore legge l'istruzione dalla memoria (fetch) /* program counter dice dove trovare la prossima istruzione; Instruction Register conterra' l'istruzione letta*/
- Esegue quindi l'istruzione
 - consideriamo il caso di un computer semplificato in cui ogni istruzione occupa una parola di memoria (16 bit)
 - PC (program counter) ha memorizzato il valore 300
 - Processore contiene un unico registro dei dati chiamato accumulatore, di 16 bit (sia dati che istruzioni lunghi per ipotesi 16 bit)

Azioni che possono essere effettuate nell'eseguire una istruzione

Processore-Memoria: dei dati possono essere trasferiti dal processore alla memoria o dalla memoria al processore

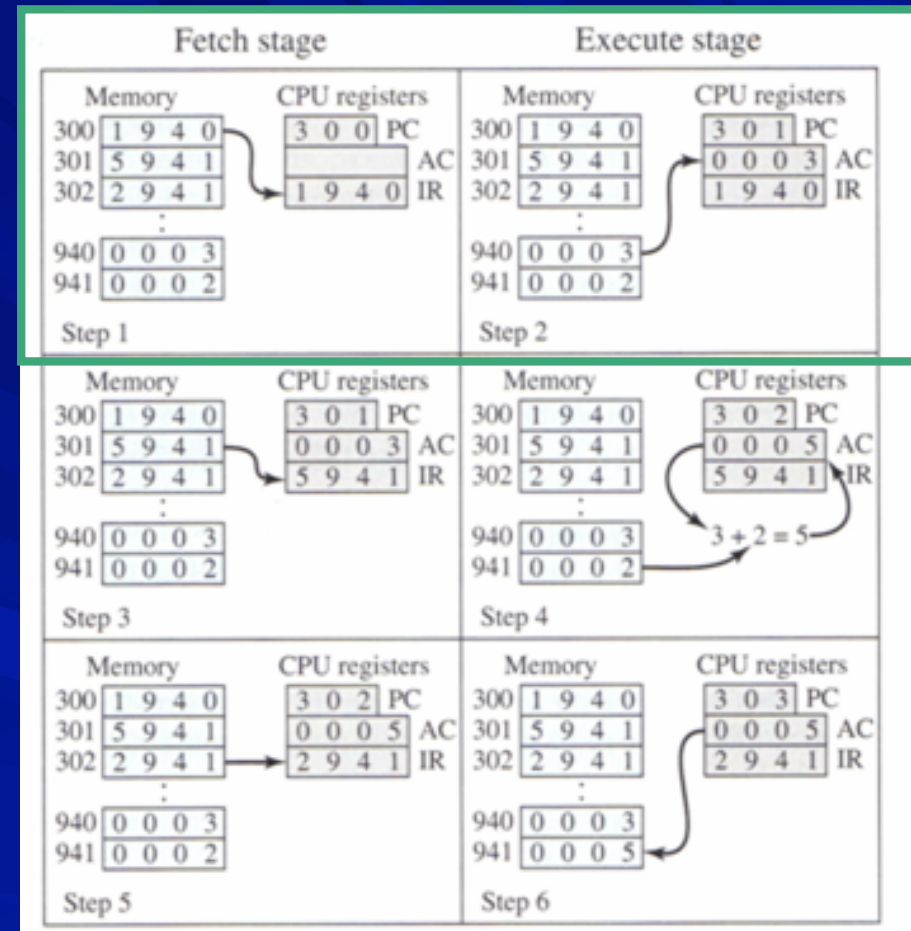
Processore-IO: dei dati possono essere trasferiti da o verso una periferica

Processing dei dati: il processore può dover eseguire operazioni aritmetiche o logiche sui dati

Controllo: ad esempio il flusso di esecuzione puo' essere modificato

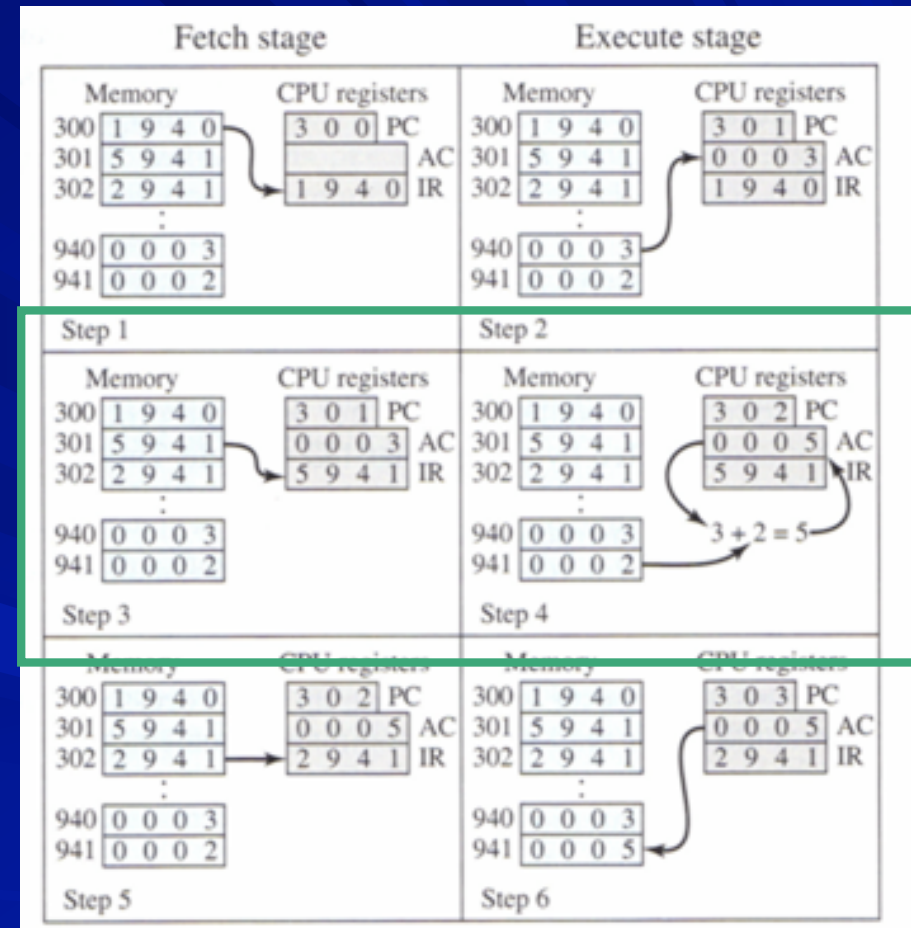
Un esempio...

- PC=300
- Si fa il fetch dell'istruzione memorizzata nella locazione di memoria 300. Tale istruzione (1940 in esadecimale) viene caricata nell'IR.
- Per fare queste operazioni si usano il MAR e il MBR come registri intermedi
- Istruzione 1940
 - 1 (primi 4 bit) indica che carichiamo un nuovo valore dalla memoria nell'accumulatore
 - L'indirizzo di memoria da cui prendere il valore è 940
- Passo 2 esegue tale istruzione



Un esempio...

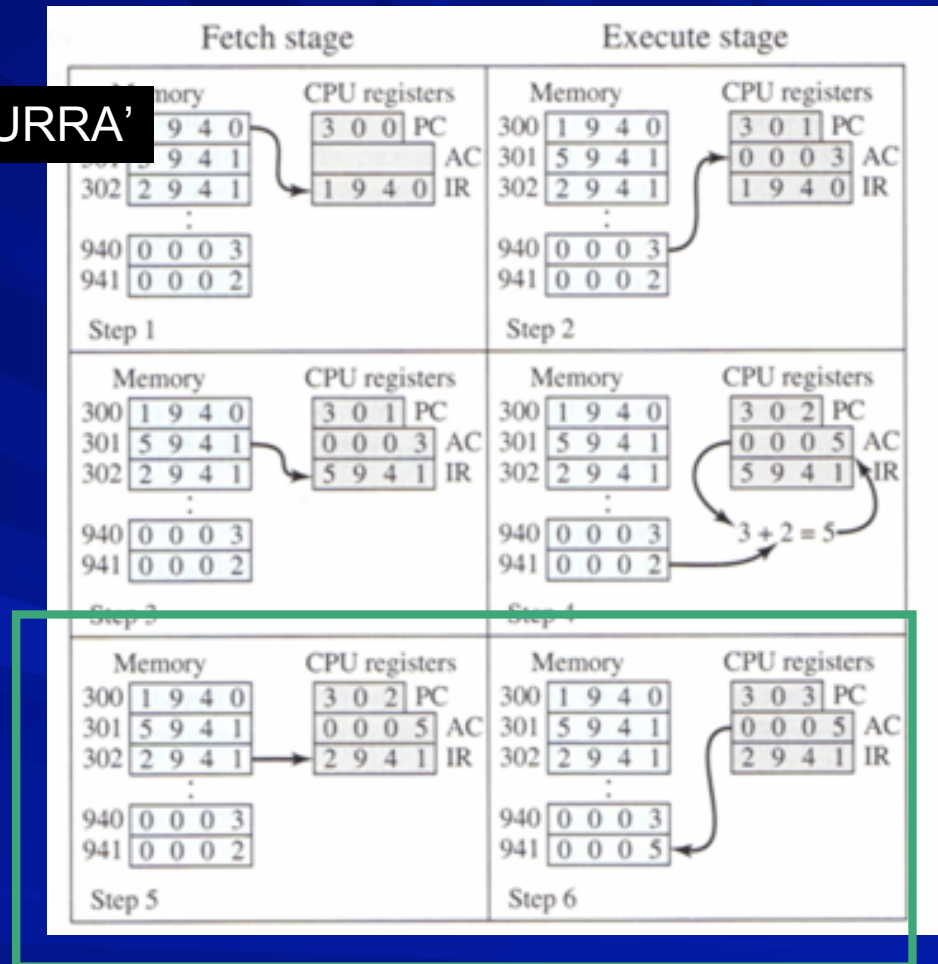
- PC=301
- Si fa il fetch dell'istruzione memorizzata nella locazione di memoria 301. Tale istruzione (5941 in esadecimale) viene caricata nell'IR.
- Per fare queste operazioni si usano il MAR e il MBR come registri intermedi
- Istruzione 5941
 - 5 (primi 4 bit) indica che il vecchio contenuto dell'AC e la locazione di memoria specificata devono essere sommate (ed il risultato memorizzato nell'AC)
 - L'indirizzo di memoria da cui prendere il valore è 941
- Passo 2 esegue tale istruzione



Un esempio...

- PC=302
- Si fa il fetch dell'istruzione memorizzata nella locazione di memoria 302. Tale istruzione (2941 in esadecimale) viene caricata nell'IR.
- Per fare queste operazioni si usano il MAR e il MBR come registri intermedi
- Istruzione 2941
 - 2 (primi 4 bit) indica che il contenuto dell'AC deve essere copiato nella locazione di memoria specificata
 - L'indirizzo di memoria in cui memorizzare il valore dell'AC è 941
- Passo 2 esegue tale istruzione

URRA'



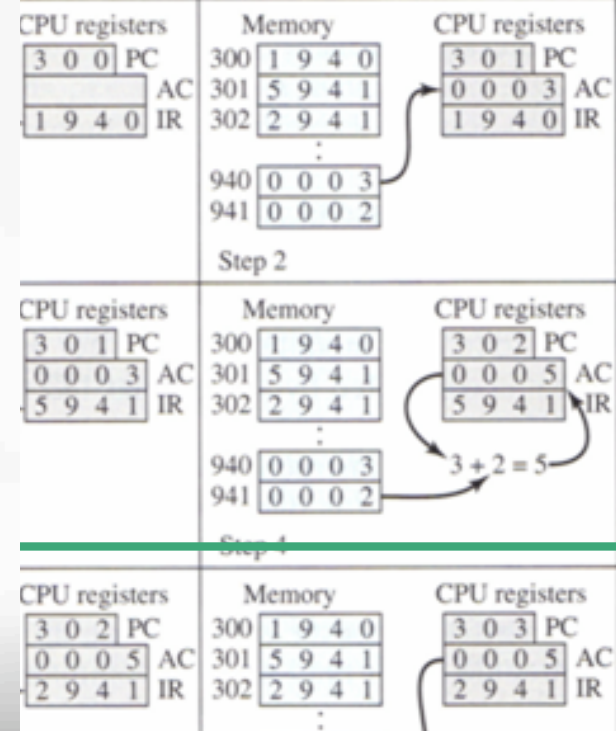
Un esempio...

- PC=302
- Si fa il fetch dell'istruzione memorizzata nella locazione di memoria 302. Tale istruzione (2941 in esadecimale) viene copiata nell'IR.
- Per fare queste operazioni si usano il MAR e il MB come registri intermedi
- Istruzione 2941
 - 2 (primi 4 bit) indica il contenuto dell'AC deve essere copiato nella locazione di memoria specificata



Fetch stage

Execute stage



Che cosa abbiamo fatto? Siano x e y due variabili memorizzate nelle locazioni 940 e 941. Abbiamo semplicemente eseguito $y = x + y$;
Complicato!! Dobbiamo programmare conoscendo così da vicino i dettagli della macchina?
No. Si programma in questo modo in linguaggio macchina. Per fortuna il C è un linguaggio ad alto livello...

Compilatore

- L'elaboratore è in grado di comprendere ed eseguire direttamente programmi scritti in linguaggio macchina
- Il linguaggio di programmazione che studieremo, il C, è un linguaggio di alto livello. Diversamente dal linguaggio macchina, difficile da usare, e che richiede una comprensione approfondita dell'HW, i linguaggi ad alto livello consentono di esprimere in modo del tutto naturale una serie di passi elementari che devono essere compiuti dal programma. (complessità crescente del SW richiede linguaggi di alto livello efficaci e buona programmazione)



- Occorre un traduttore che una volta scritto in un linguaggio ad alto livello un programma traduca questo programma in linguaggio macchina. Tale compito di traduzione è svolto dal compilatore del linguaggio (es. da gcc, compilatore C).

Linguaggi di alto livello vs. linguaggio macchina (un esempio)

Task: Sommare straordinari a paga base per ottenere lo stipendio lordo

- Linguaggio macchina (dipende dal tipo di computer ed è difficile da programmare)

+1300042774

+1400593419

+ 1200274027

- Assembly language

LOAD BASEPAY

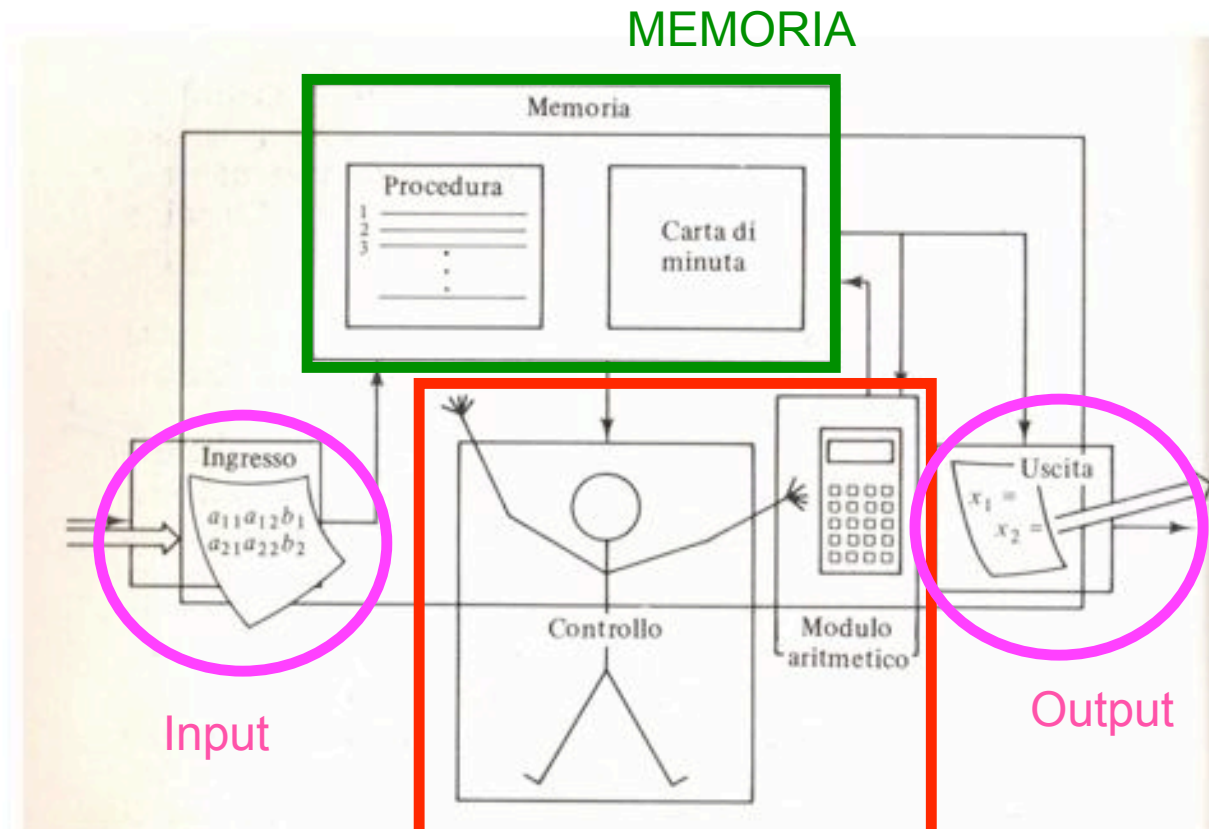
ADD OVERPAY

STORE GROSSPAY

- Linguaggio ad alto livello

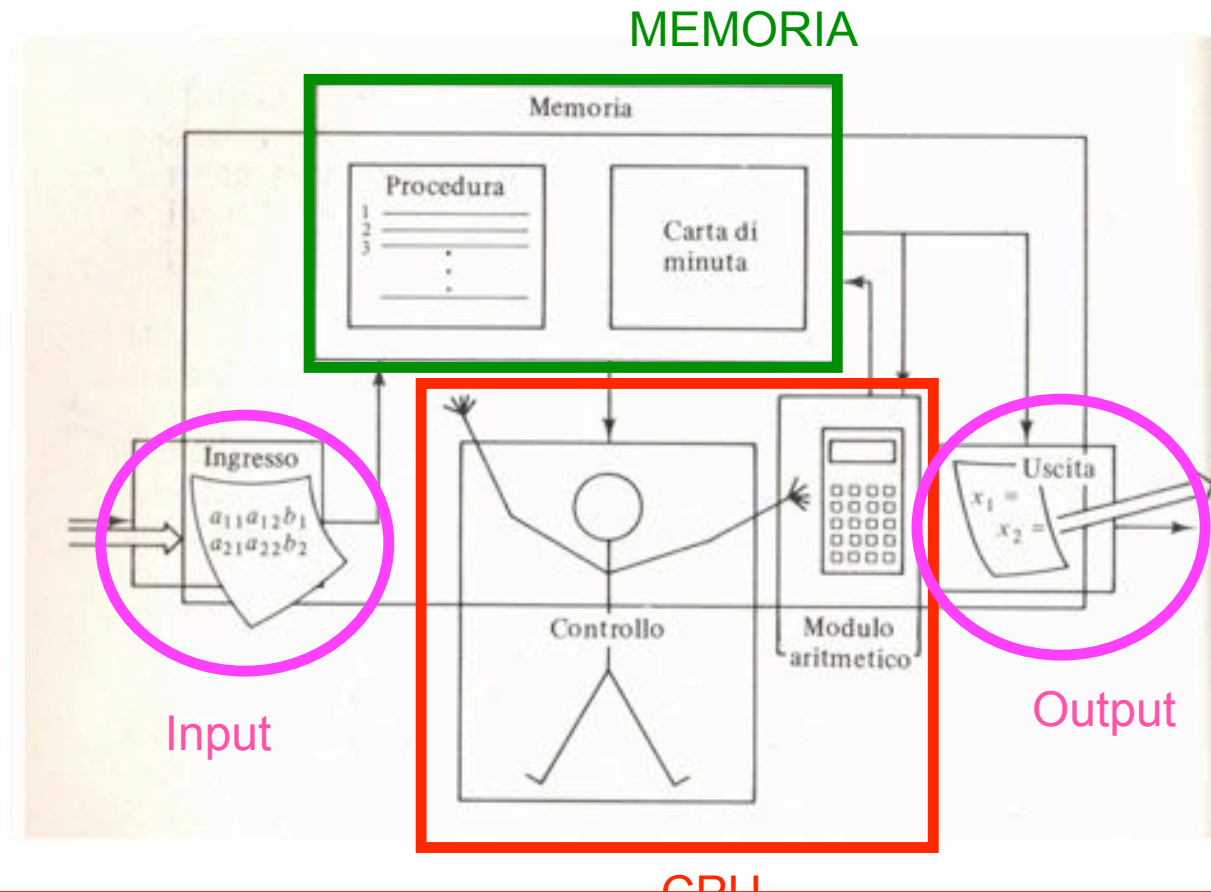
grossPay = basePay + overTimePay;

Componenti di un elaboratore



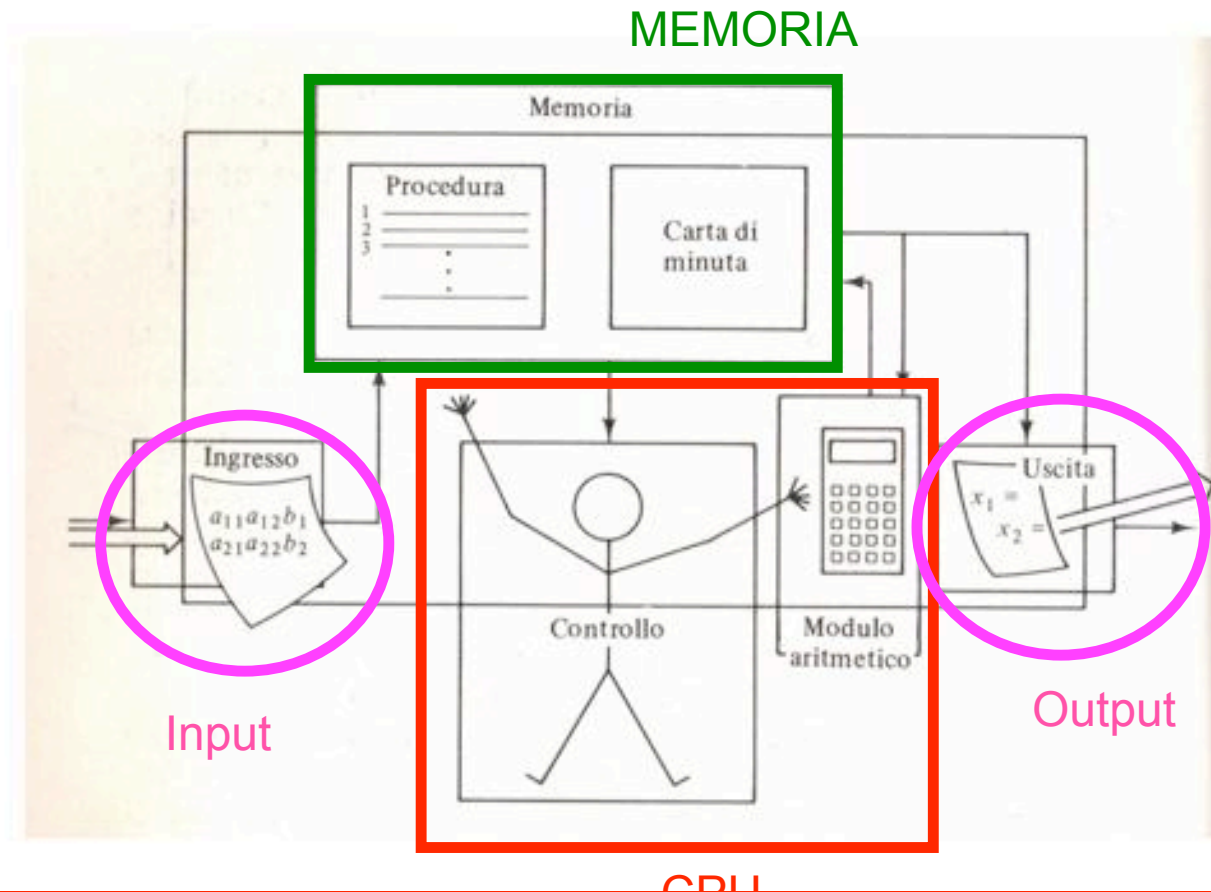
Scriviamo un programma. Il nostro programma deve essere caricato in memoria principale per poter essere eseguito. L'utente scrive il programma in un linguaggio ad alto livello. Prima dell'esecuzione del programma il programma viene compilato ottenendo un eseguibile in linguaggio macchina, che il calcolatore comprende... Si comincia quindi ad eseguire il programma...

Componenti di un elaboratore



Al termine dell'esecuzione del programma tale programma stamperà se il sistema di equazioni lineari ha una soluzione, è inconsistente o ha infinite soluzioni. Se ha una soluzione il valore sarà memorizzato in x_1 e x_2 e questo valore verrà fornito in output all'utente (su schermo o stampato tramite stampante, o stampato su file,...)

Componenti di un elaboratore



Al termine dell'esecuzione del programma tale programma stamperà se il sistema

Le risorse di un calcolatore possono essere condivise da più utenti o da più programmi. Chi decide quale programma eseguire, quali risorse dedicare all'esecuzione del programma (ad esempio dove memorizzare le variabili del programma etc.?)

SU FILE,...)

Sistema Operativo



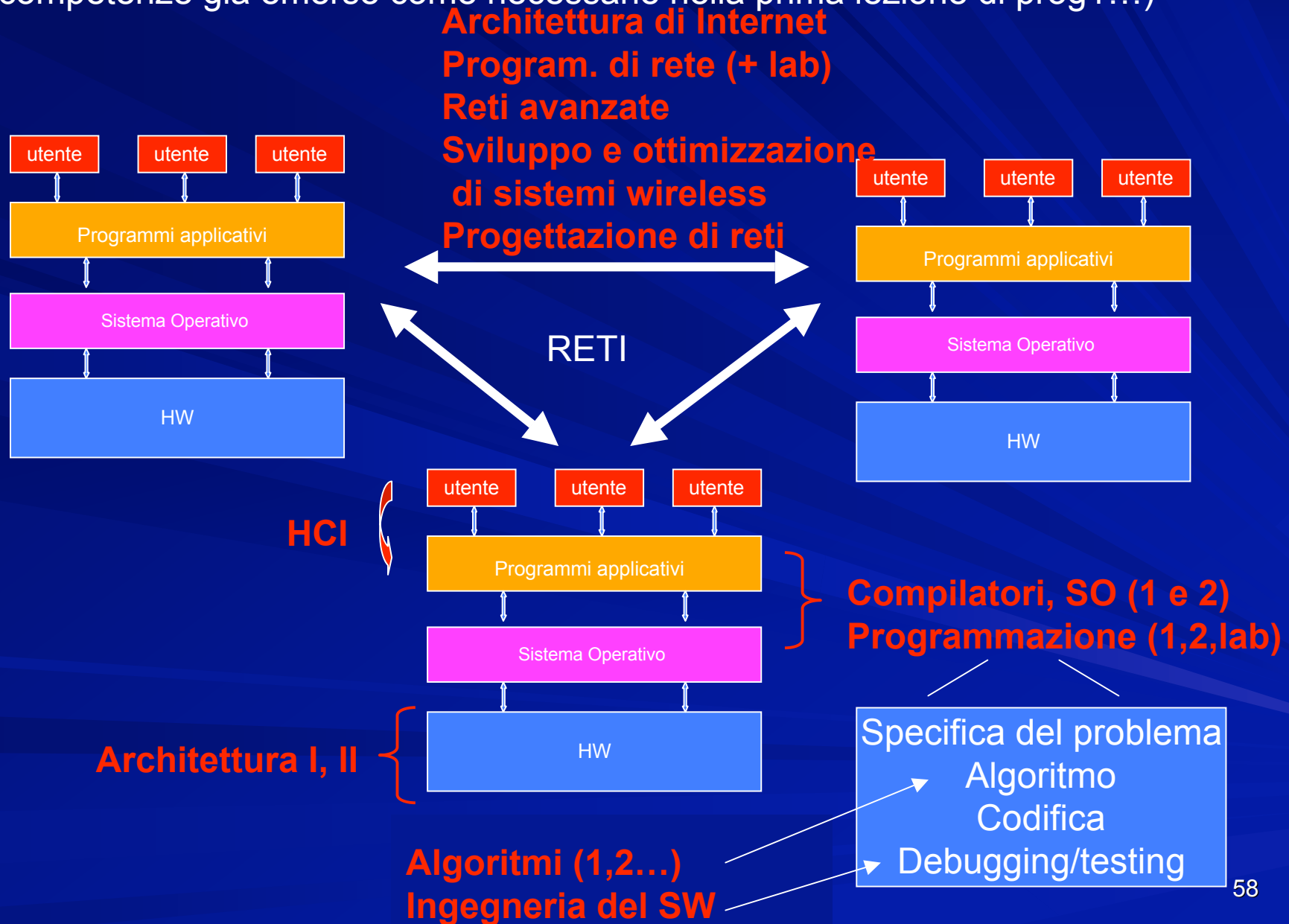
Vedrete il laboratorio come si lavora da utente su Linux, il gcc (compilatore C della GNU) e vi (editor di testo)

Dai mainframe all'ubiquitous computing

- Dimensioni e costo decrescente dell'HW hanno fatto sì che si passasse dai mainframe ai personal computer prima (anni 80') e che l'evoluzione consentisse di inserire capacità di calcolo (e comunicazione) all'interno di tutti gli oggetti che ci circondano (ubiquitous computing)
- Gli elaboratori non sono più stand-alone ma possono comunicare tramite reti che li interconnettono (rendendo possibile una computazione distribuita)

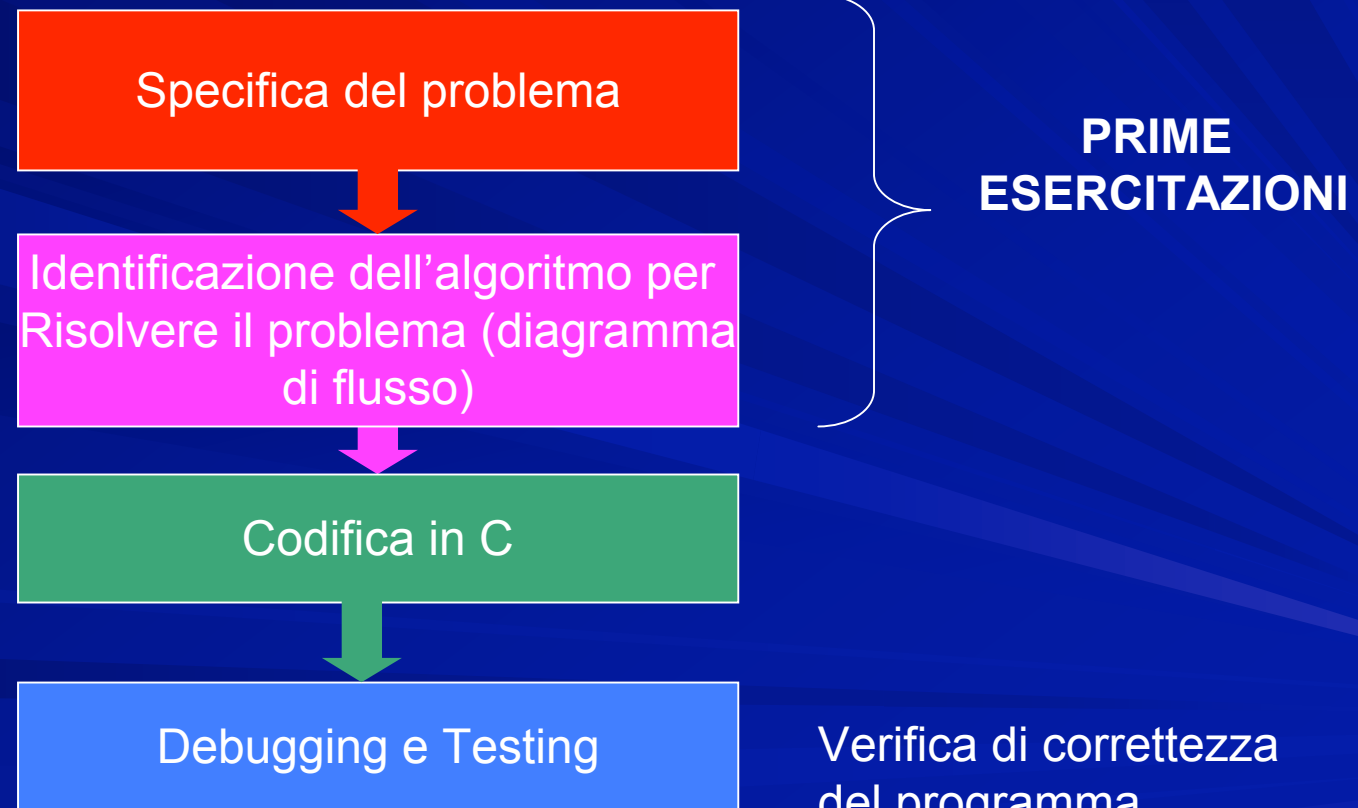
...Struttura del corso di laurea

(competenze già emerse come necessarie nella prima lezione di prog1...)



Programma di questo corso

■ Imparare a programmare in linguaggio C



Informazioni generali (1)

- Docente: Prof.ssa Chiara Petrioli
 - Lezioni in AULA 5 di matematica il giovedì e venerdì
 - Ricevimento il mercoledì studio n 311, terzo piano, Via Salaria 113, ore 11.30-13.30
 - **per sapere di più sulla mia attività di ricerca mia pagina personale** <http://reti.dsi.uniroma1.it/eng/petrioli/bio.html>
 - Esercitatore: Dott. Ivano Salvo
 - Lezioni in AULA 5 di matematica il lunedì (tipicamente)
- Tutore: Dott. Michele Mastrogiovanni
 - Laboratorio a Via Salaria 113, piano seminterrato.

Lunedì e Giovedì mattina. Il tutore arriverà alle ore 9.30. Vi dividerete in due turni. Questa settimana un unico turno giovedì mattina. In via eccezionale il laboratorio comincia alle ore 10.00

Una digressione...chi è e cosa fa un docente universitario

■ Struttura dei corsi di laurea



- I docenti universitari sono tipicamente Dottori di Ricerca, selezionati tra i migliori del loro corso
- Svolgono attività di ricerca (proposta di soluzioni innovative nel loro ambito di ricerca, valutazione di tali soluzioni, disseminazione tramite articoli scientifici pubblicati in conferenze e journal)
Conferenze e journal di riferimento sono quelle dell'ACM e dell'IEEE
← avete libero accesso alla digital library dell'ACM e IEEE dalle macchine del dipartimento

Una digressione...chi è e cosa fa un docente universitario

- Svolgono servizi per la comunità scientifica internazionale (es: organizzazione di conferenze, editorial board di journal,...)
- Possono essere a capo di gruppi di ricerca che coinvolgono vari collaboratori (dottorandi di ricerca, postdoc) ← per finanziare tali gruppi possono effettuare attività di R&D in collaborazione con centri di ricerca industriale, o in progetti finanziati dalla Comunità Europea
- Svolgono attività didattica (due corsi l'anno, tesi triennali, tesi quinquennali, advising di dottorandi di ricerca)
- Svolgono attività per il dipartimento/corso di laurea (partecipazione a commissioni, sedute di laurea, COADI, Consigli di Dipartimento, Collegi di Dottorato)
- Carriera tipica:
 - (Laurea)
 - Dottorato di Ricerca
 - Postdoc
 - Ricercatore
 - Professore Associato
 - Professore Ordinario



strutturati

Prof. Chiara Petrioli, Corso di
Programmazione 1, a.a. 2006/2007

Una digressione...chi è e cosa fa un docente universitario

- Svolgono servizi per la comunità scientifica internazionale (es: organizzazione di conferenze, editorial board di journal,...)
- Possono essere a capo di gruppi di ricerca che coinvolgono vari collaboratori (dottorandi di ricerca, postdoc) ← per finanziare tali gruppi possono effettuare attività di R&D in collaborazione con centri di ricerca europei

- Svolgono attività di ricerca a quinquennale

- Svolgono attività di insegnamento a commissione e nei Collegi di Ingegneria

- Carriera tipica

- (Laurea)
- Dottorato
- Postdoc

- Ricercatore
- Professore Associato
- Professore Ordinario



Diversamente dagli stereotipi un docente universitario
è tipicamente una persona molto impegnata
→rispettate gli orari di ricevimento

MA

→cercate anche di imparare il piu' possibile dalle
Interazioni con persone che sono
esperti riconosciuti a livello internazionale in un
ambito (vero chiaramente soprattutto per i
corsi specialistici)

strutturati

Prof. Chiara Petrioli, Corso di
Programmazione 1, a.a. 2006/2007

Informazioni generali (2)

■ Libri di testo

- Deitel & Deitel “C: How to Program-fourth edition”, Prentice Hall, 2004 (meglio abituarsi a studiare sui testi originali)

■ Materiale di consultazione

- A mano a mano disponibile sul sito. Prime lezioni: Lucidi, cap. 2 del Preparata di architetture (prime 5-6 pagine), Stallings di Sistemi Operativi, cap 1 sull'organizzazione di un calcolatore.

■ Sito del corso:

- andate su twiki.di.uniroma1.it

Laurea triennale, prog1 AD

Informazioni generali (3)

■ Esame:

- Scritto (domande/ esercizi).
 - Due esoneri previsti durante le interruzioni del calendario accademico (primo esonero: 6 novembre)
- Verbalizzazione dello scritto oppure scritto + orale (orale necessario per la lode o per verbalizzare voti superiori al 28)
- Homework bisettimanali, da fare possibilmente in laboratorio (fino a 5 punti extra). Chi non dovesse fare gli Homework dovrà sostenere un esame di laboratorio.
- Numero degli appelli: 3

■ Un messaggio esplicito: seguite le lezioni in aula, studiate ed esercitatevi al calcolatore, sfruttate tutte le opportunit  (esercitatori, tutor, laboratorio, esoneri + ricevimento) per apprendere, chiarirvi i dubbi e passare immediatamente l'esame. Non rimanete indietro!