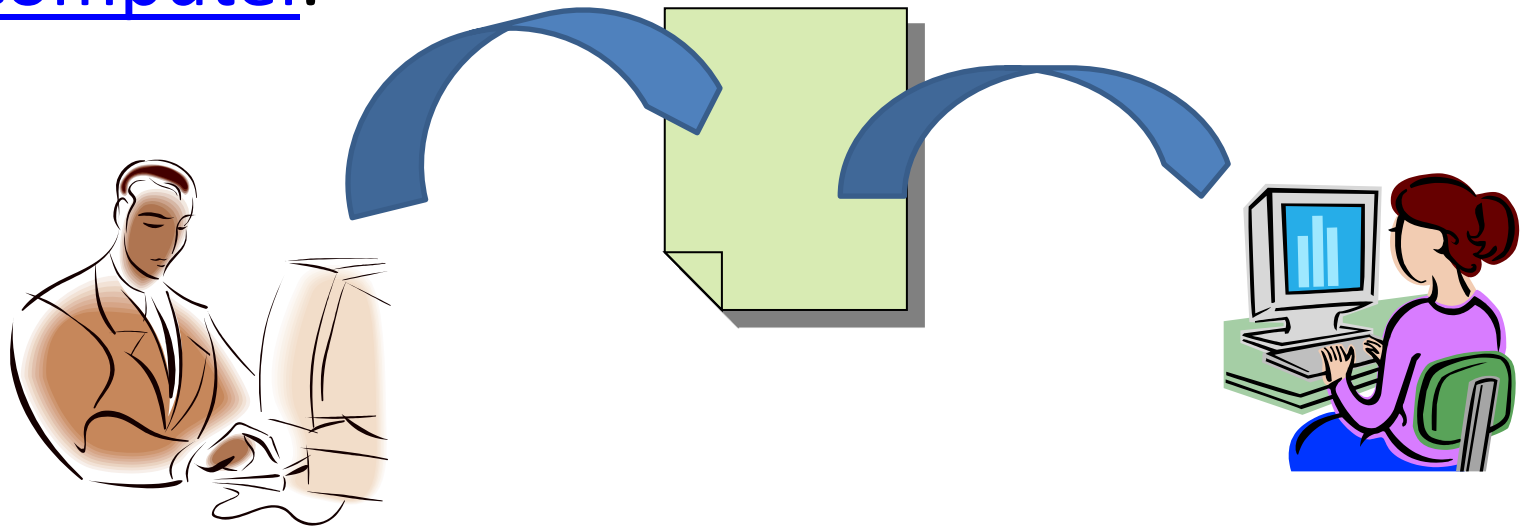
Linguaggi di Programmazione

Linguaggi di Programmazione

Programmazione. Insieme delle attività e tecniche svolte per creare un programma (*codice sorgente)* da far eseguire ad un computer.



Che lingua comprende un Computer?

Il **linguaggio macchina** o **codice macchina** è dato dall'insieme di "parole" che si possono creare con l'alfabeto binario: comprende due soli simboli, generalmente indicati con 0 e 1 (bit).

Ogni modello di processore è in grado di comprendere un proprio particolare linguaggio macchina.

000001000000011

Linguaggi di Programmazione

Il linguaggio di programmazione è un linguaggio formale, dotato di un *vocabolario*, di una *sintassi* e di una *semantica* ben definiti.

- ***A Basso Livello: Assembly***
- ***Ad alto livello: c++,Java, PHP,…***

Di seguito con il termine ***linguaggio di programmazione*** *ci riferiremo solo a quelli di alto livello*

Linguaggi di Programmazione

Si vuole scrivere un programma che sommi la *paga base* e la *paga straordinaria* di un impiegato per calcolare la *paga lorda*.

MACCHINA	ASSEMBLY	ALTO LIVELLO
+1300042774 +140593419 +1200274027	Load *pagabase* Add *straordinario* Store *pagalorda*	pagaLorda = pagaBase + Straordinario

.

Linguaggi di Programmazione

CONCETTI CHIAVE

Variabile: un dato o un insieme di dati, noti o ignoti, già memorizzati o da memorizzare; ad essa corrisponde un certo numero di *locazioni di memoria* che vengono *allocate* (cioè riservate) per contenere i dati; essere hanno anche un *tipo* (stringhe di testo, numeri, liste,ecc.);

Istruzione: un comando, una funzione, oppure una regola descrittiva; ogni volta che una istruzione viene eseguita lo stato interno del calcolatore cambia;

Espressione: una combinazione di *variabili* e *costanti* unite da *operatori;* alcuni tipi di espressioni sono quelle matematiche;

Strutture di controllo: consentono di intervenire sul flusso di un programma alterandolo in base al risultato o valutazione di una *espressione*.

Linguaggi di Programmazione

- Linguaggio intermedio tra linguaggio macchina e linguaggio naturale
- Consente di descrivere gli algorirmi in modo rigoroso (non ambiguo)
- Non è comprensibile dal processore

Linguaggi di Programmazione

COMPILAZIONE

un programma chiamato *compilatore* si occupa della traduzione del nostro programma. *Questa avviene una sola volta*.

Durante la compilazione le istruzioni in linguaggio di alto livello vengono tradotte in istruzioni macchina; ha così origine un file nel linguaggio macchina locale, che costituisce un eseguibile.

C++ , Java

Linguaggi di Programmazione

INTERPRETAZIONE

Tutte le volte che si vuole eseguire un programma questo viene interpretato.

Un programma apposito, *l'interprete*, durante l'esecuzione del programma traduce ogni istruzione di alto livello in istruzione macchina. In questo caso si ha una miglior portabilità, ma peggiori prestazioni.

PHP

Linguaggi di Programmazione

Esempio di Programma (massimo di 3 numeri)

```
int calcolaMassimo(int num1, int num2, int num3)
{
  int valoreMassimo := num1;
  if(num2 > num3)
  {
    if(num2 > num1)
      valoreMassimo := num2;
  }
  else
  {
    if(num3 > num1)
      valoreMassimo := num3;
  }
  return valoreMassimo;
}
```

Altri Linguaggi: HyperText Markup Language (HTML)

HyperText Markup Language (**HTML**) (*linguaggio di descrizione per ipertesti*) è il linguaggio solitamente usato per i documenti ipertestuali disponibili nel WEB.

Il linguaggio HTML mette a disposizione delle etichette, *tag*, che consentono di descrivere le caratteristiche e la struttura dele testo (colore, link, paragrafo)

Un browawe scarica il contenuto HTML ed eventuali documenti collegati e li elabora, ossia ne interpreta il codice, al fine di generare la visualizzazione della pagina desiderata.

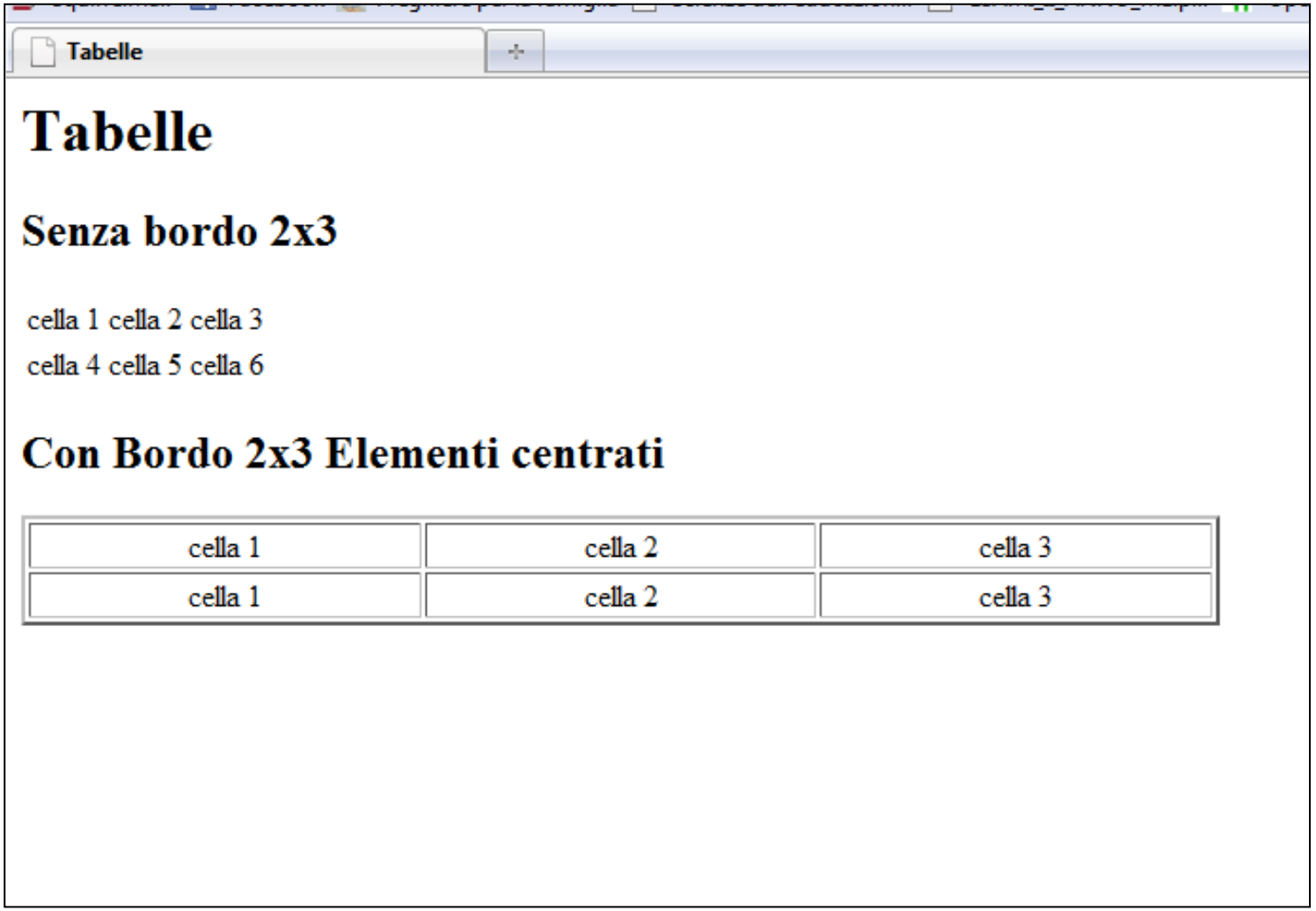
Altri Linguaggi:
HyperText Markup Language (HTML)

HTML non è un Linguaggio di Programmazione ma un <u>Linguaggio di Markup</u>

non prevede alcuna:

- Variabili
- Strutture dati
- Funzioni
- Strutture di controllo

Altri Linguaggi:
HyperText Markup Language (HTML)



Altri Linguaggi: HyperText Markup Language (HTML)

Codice Sorgente FILE

```
<HTML>
 <HEAD>
   <TITLE> Tabelle </TITLE>
</HEAD>
<BODY>
<H1>Tabelle</H1>
<H2>Senza bordo 2x3</H2>
<TABLE> <TR>
<TD>cella 1</TD> <TD>cella 2</TD> <TD>cella 3</TD>
</TR> <TR>
<TD>cella 4</TD> <TD>cella 5</TD> <TD>cella 6</TD>
</TR>
</TABLE>
<H2> Con Bordo 2x3 Elementi centrati </H2>
                                    … … …
```

Rete Di Calcolatori

Più computer connessi tra di loro che sono in grado di scambiare Dati e condividere Risorse.

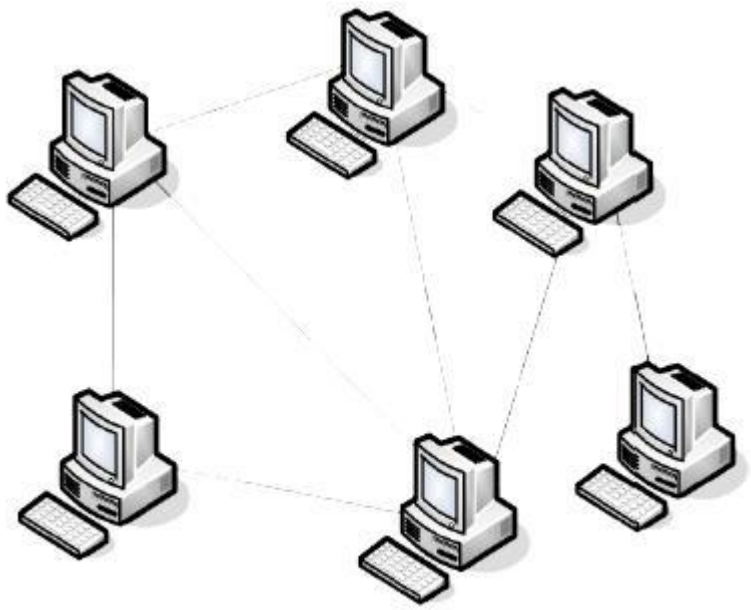
Classificazione di una Rete per Architettura:

- **Peer to Peer**
- **Client-Server**

Rete di Calcolatori

Architettura Peer to Peer

In questa tipologia di Architettura si ha una vera e propria Anarchia. **Ogni computer, oltre a poter fungere sia da Client che da Server non ha nulla a che fare con gli altri e non segue perciò alcuna Politica di Rete.** La forza di queste Reti è sicuramente la semplicità che permette una manutenzione nulla ed un alta flessibilità per accedere. Gli svantaggi si incominciano a vedere per Reti di grandi dimensioni.



Rete di Calcolatori

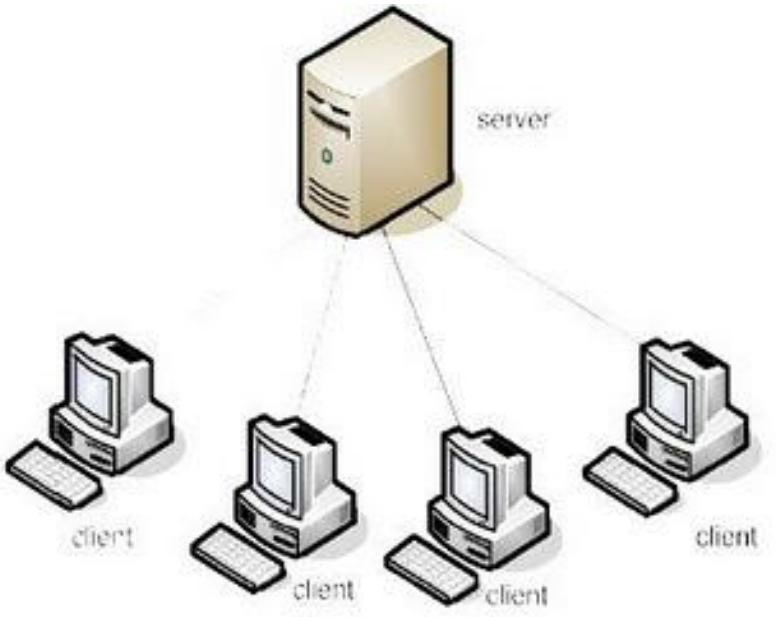
Architettura Client Server

Il Server è colui che elabora i Dati

Il client è colui che effettua le richieste

Ogni computer può essere sia Client che Server

La presenza di un *server* permette ad un certo numero di *client* di condividerne le risorse, lasciando che sia il *server* a gestire gli accessi alle risorse per evitare conflitti



Applicazioni Client-Server

Client

Il software *client* in genere è di limitata complessità, limitandosi normalmente ad operare come interfaccia verso il *server*.

Itermine *client* indica una componente che accede ai servizi o alle risorse di un'altra componente, detta *server*.

ESEMPI: posta elettronica, browser,…

Applicazioni Client-Server

Server

Il software *server*, oltre alla gestione logica del sistema, deve implementare tutte le tecniche di gestione degli accessi, allocazione e rilascio delle risorse, condivisione e sicurezza dei dati o delle risorse.

Esempio

Un *server* di posta elettronica è paragonabile ad un qualunque ufficio postale. Gli utilizzatori per accedere via *client* alla loro cassetta di posta elettronica devono essere stati autorizzati. In modo analogo un utente deve possedere la chiave della cassetta sita presso un ufficio postale dalla quale vuole prelevare la corrispondenza.

Linguaggi di Programmazione: PHP

- **PHP** (*Hypertext Preprocessor, preprocessore di ipertesti)* è un linguaggio di programmazione interpretato, con licenza open source nato per la realizzazione di pagine *web dinamiche (*Una **pagina web** il cui contenuto è, in tutto o in parte, generato sul momento dal server e può essere quindi diversa ogni volta che viene richiamata).
- Attualmente è utilizzato principalmente per sviluppare applicazione web lato sever ma può essere usato anche per altri tipi di applicazione
- L'elaborazione di codice PHP sul server produce codice **HTML** da inviare al browser dell'utente che ne fa richiesta.

Programmare in PHP

ALGORITMO

Dichiara *base* come numero

Dichiara *altezza* come numero

Leggi *base*

Leggi *altezza*

Dichiara *area* come numero

*area = base * altezza*

Stampa *area*

PHP

```
<?php

$base = (int)$_POST['base'];
$altezza = (int)$_POST['altez'];
$area = $base*$altezza;
echo $area;

?>
```

Programmare in PHP
Area di un rettangolo

ALGORITMO

Dichiara *base* come numero

Dichiara *altezza* come numero

Leggi *base*

Leggi *altezza*

Dichiara *area* come numero

*area = base * altezza*

Stampa *area*

PHP

```
<?php

$base = (int)$_POST['base'];
$altezza = (int)$_POST['altez'];
$area = $base*$altezza;
echo $area;

?>
```

Programmare in PHP
somma di due numeri

ALGORITMO

Dichiara *numero1* come numero

Dichiara *numero2* come numero

Leggi *numero1*

Leggi *numero2*

Dichiara *somma* come numero

somma= numero1 + numero2

Stampa *somma*

PHP

```
<?php

$numero1 = $_POST['num1'];
$numero2 = $_POST['num2'];
$somma = $numero1+$numero2
echo $somma;

?>
```

Programmare in PHP
Massimo di due numeri

ALGORITMO

Dichiara *numero1* come numero

Dichiara *numero2* come numero

Leggi *numero1*

Leggi *numero2*

Se *numero1 >= numero2*

    Stampa *numero1*

Altrimenti

    Stampa *numero2*

PHP

```
<?php

$numero1 = $_POST[‘num1'];
$numero2 = $_POST[‘num2'];
if ($numer1 >= $numero2)
   echo $numero1;
else
   echo $numero2;
?>
```