

# PARADIGMI DI PROGRAMMAZIONE:

**FUNZIONALE:** Basati su funzioni Matematiche, Base Teorica Vicina All' Utente ma distante dall'Architettura su cui il Programma Viene Eseguito

[Compongo funzioni Diverse Per fare il Calcolo]

**ITERATIVI:** Nasce dall'Architettura di Von Neumann  $\Rightarrow$  Esecuzione di una Serie di Istruzioni Su Celle di Memoria.

Hanno Come Obiettivo L'EFFICIENZA

## CONCETTI FONDAMENTALI

/// Funzione Matematica è l'ELEMENTO PRIMITIVO

/// Variabile è un PLACEHOLDER e NON CELLA DI MEMORIA,

$$f(x) = x^2 + 2x + 1$$

Sostituisco le OCCORRENZE

/// Funzione è una Mappa di 2 Insiemi

$$\lambda\text{-Calcolo} \Rightarrow f(x) = x^2 + 2x + 1 \Rightarrow f = \lambda x. x^2 + 2x + 1$$

↪ Separo il nome dal PARAMETRO

/// LISP ha Alla Base il  $\lambda$  Calcolo

/// Programma è una composizione di Funzione

Ho Solo Categoria delle ESPRESSIONI, comandi non mi Servono  
Per MODIFICARE la MEMORIA

Applicazione  $\Rightarrow$  ee di 2  $\lambda$ -Termini

Associa a SINISTRA con Precedenza Sull' ASTRAZIONE ?

Astrazione  $\Rightarrow \lambda x.e$

Funzioni hanno una COMPOSIZIONE che va In direzione  
opposta a Quella SEQUENZIALE, infatti:

$C; d \Rightarrow$  Prima C e poi D

$f(g(x)) \Rightarrow$  Prima  $g()$  e dopo  $f()$  con Risultato  $g()$

## LISP:

Ad Alto Livello Creato con Notazione Matematica

Orientato Alle Espressioni, TUTTO è un ESPRESSIONE (anche i Comandi)

Tutto Quello dopo l'Apice NON Viene Interpretato

# PARADIGMA LOGICO:

calcolo dei Predicati

Computazione Come DIMOSTRAZIONE, Attraverso **Logica Simbolica**

È dichiarativo e NON procedurale, Vengono date solo le proprietà del RISULTATO

OGGETTI  $\Rightarrow$  ~~///~~ COSTANTE  
~~///~~ VARIABLE

## PROLOG:

Dichiaro Fatti e Relazioni Sugli OGGETTI  
CONOSCENZA

Simili ai LINGUAGGI ad Accesso ai DataBase

PREDICATO al Quale DO SIGNIFICATO

CONOSCENZA è Tutto ciò che NOI descriviamo anche se si contrappone alla REALTÀ (ASSUME TUTTO VERO)

La Risoluzione Si Basa Su UNIFICAZIONE ed ISTANZIAMENTO

Processo MATCHING

Valori Temporanei alle Var