

# DESCRIVERE I P.L.

/// **SINTASSI**  $\Rightarrow$  Grammatiche c.f. e Linguaggi Regolari

/// **SEMANTICA**  $\Rightarrow$  Associazione Significato ai Simboli, effetto simboli sulla Macchina

/// **PRAGMATICA**  $\Rightarrow$  Regole non formali di Buona Programmazione

/// **IMPLEMENTAZIONE**  $\Rightarrow$  Poiché Sono Eseguibili

① Interprete

② Compilatore

③ Ibrido

## SEMPLICE LINGUAGGIO IMPERATIVO:

/// No dichiarazioni

/// Solo BOOLEANI

/// Assegnamento, Composizione Sequenziale

/// Comando Condizionale

/// Comando Iterativo

Programma  $\Rightarrow$

Comandi Composti  $\Rightarrow$ ; Controllo

Comandi Base  $\Rightarrow$  Assegnamento, skip

Sintassi Espressioni  $\Rightarrow$  Operatori BOOLEANI

Identificatori Parole Chiave  $\Rightarrow$  x..., WHILE, IF, ...

Notazione della Grammatica  $\Rightarrow$   $:= (\rightarrow)$  e  $|$  (separatore)

**BOOLEAN - EXPRESSION:**  $B \rightarrow \text{true} \mid \text{false} \mid (\text{not } B) \mid (B \text{ and } B) \mid (B \text{ or } B)$

Elementi Terminali/Base della Grammatica

**COMANDI BASA/ATOMICI:**  $A \rightarrow x := B \mid \text{skip}$

**COMANDI COMPOSTI:**  $C \rightarrow A \mid C; C \mid \text{if } B \text{ then } C \text{ else } C \mid \text{WHILE } B \text{ do } C$

**PROGRAMMA:**  $P \rightarrow C$

---

Per rendere UNICO il valore di un EXPRESSION devo avere la possibilità di parentesizzare

**Albero di Parsing deve Essere UNICO.**

SINTASSI ~~~~~ SEMANTICA

ANALISI SEMANTICA

SEMANTICA STATICA



/// #Parametri Attuali in una chiamata a Procedura

/// Correttezza dei Tipi

Chiamati **VINCOLI SINTATTICI** ma che **NON** possono Essere Verificati **dalla Grammatica essendo CONTEXT-FREE**

## SEMANTICA:

Associazione di Significati ai Simboli del linguaggio e quindi Agli Elementi del linguaggio

SEMANTICA DENOTAZIONALE  $\Rightarrow$  Comportamento I/O dove descriviamo funzionalità  $\Rightarrow$  Cosa calcola

SEMANTICA ASSIOMATICA  $\Rightarrow$  descrive delle Proprietà e Voglio  
 $\hookrightarrow$  fisso dei Vincoli

Capire Se Vengono Mantenerate o Meno  $\Rightarrow$  descrivo Comportamento dimostrandone le Proprietà

SEMANTICA OPERAZIONALE  $\Rightarrow$  Descrive il Comportamento mediante Trasformazione di Stato (siamo nel come)

## ESEMPIO:

$z := 2;$   
 $y := z;$   
 $y := y + 1;$   
 $z := y;$

SEMANTICA DENOTAZIONALE  $\rightarrow$  I/O

$S_0 = \langle z \mapsto \perp, y \mapsto \perp \rangle$

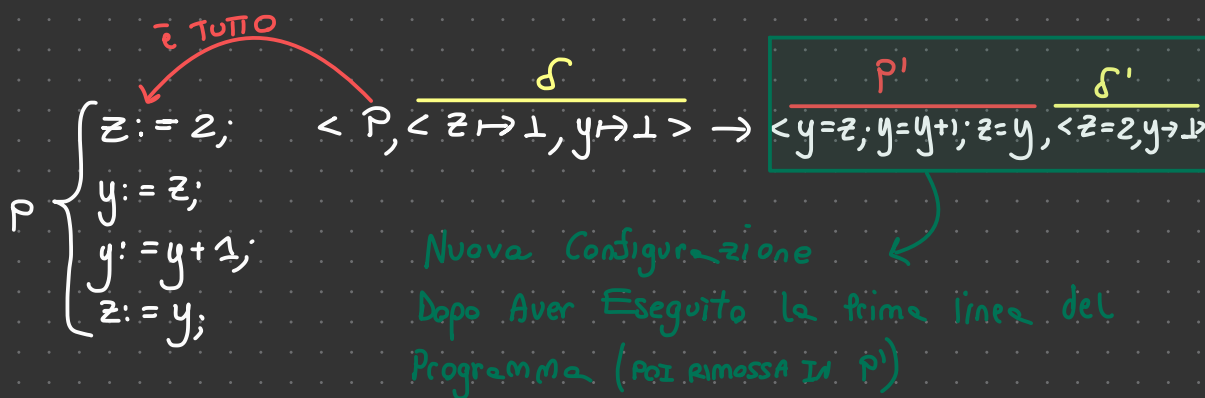
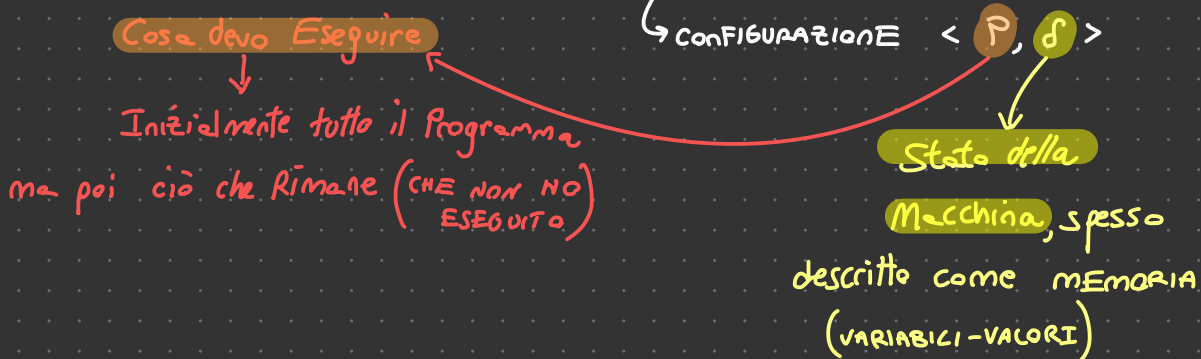
Stato Iniziale con  $z$  e  $y$  definiti ( $\perp$ )

$D(p)(s_0) = [D(z := y) \circ D(y := y + 1)] \circ$

$D(z := 2) \rightarrow$  Raggiunge Stato  
( $z \mapsto 2, y \mapsto \perp$ )

E poi via via **Inizio a Comporre le funzioni** arrivando a restituire l'OUTPUT  $\Rightarrow \langle z \rightarrow 3, y \rightarrow 3 \rangle$

**SEMANTICA OPERAZIONALE**  $\Rightarrow$  Come il Risultato Viene Costruito,  
; **SINGOLI PASSI DI CALCOLO** usando il Sistema di Transizione



Quindi è la Traccia/Sequenza di Configurazione

$\langle P, \delta \rangle \rightarrow \langle P', \delta' \rangle \rightarrow \dots \rightarrow \langle P^F, \delta^F \rangle$  *finali*

## INDUZIONE STRUTTURALE:

Come funziona dimostrazione per una proprietà  $\pi$

① Dimostriamo  $\pi$  per tutti gli Elementi Base della Categoria (TERMINALI)

Quelle produzioni in cui a dx non compaiono Simboli Terminali

$B \rightarrow \text{true}$  e  $B \rightarrow \text{false}$  Sono BASE

$B \rightarrow (\text{not } B)$   non è Terminale come simbolo

② Dimostro  $\pi$  per tutti gli Elementi Composti della Categoria, ovvero generati da produzioni dove ho il simbolo della Categoria a dx.

$B \rightarrow (\text{not } B)$  e  $B \rightarrow (B \text{ and } B)$

assumendo che  $\pi$  valga per tutti i COMPONENTI IMMEDIATI

vale - dire i Simboli/Elementi della Categoria a dx. della produzione

$B \rightarrow (\text{not } B)$   COMPONENTE IMMEDIATO

## ESEMPI:

Definisco  $X \subseteq \mathbb{N}$  per Induz. Strutt.

**BASE:**  $0 \in X$  Altri  $n$  sono esclusi per costruzione  
(non posso ottenere ad esempio 1  
come somme di 3 mai!)

**PASSO:**  $n \in X \Rightarrow (n+3) \in X$  [Tutti i multipli di 3]

Dimostro che  $X = \{n \mid n \in 3\mathbb{N}\}$

① **BASE**  $\Rightarrow 0 \in X$  e  $0 \in 3\mathbb{N}$

② **PASSO**  $\Rightarrow n \in X$  e per Ipotesi Induttiva so che  $n \in 3\mathbb{N}$  e devo dimostrare che  $(n+3) \in 3\mathbb{N}$  ma:

$n \in 3\mathbb{N}$  per def.  $\exists i. n = 3 \cdot i$  ma

$n+3 = 3i+3 = 3(i+1) \in 3\mathbb{N}$  per definizione

Date una definizione strutturale poi dimostro per Induzione

**DEFINIAMO UN INSIEME (BT) ALBERI BINARI:**

$N$  insieme dei nodi

**BASE**  $\Rightarrow n \in N \rightarrow n \in BT$  (nodi sono alberi binari)

**PASSO**  $\Rightarrow T_1, T_2 \in BT$   
 $n \in N$   $br(n, T_1, T_2) \rightarrow$  Specifico Radice  
 $br(T_1, T_2) \rightarrow \forall n \in N$

