

# Schemi di Traduzione

(variante delle SDD)

Def.

Una "Syntax-Directed Translation scheme" è una grammatica in cui le produzioni sono arricchite da frammenti di codice (azioni semantiche) che vengono eseguiti nel momento in cui tutti i simboli alla loro sinistra sono stati riconosciuti.

## DIFFERENZE TRA SDD E SDT

### SDD

- Specificano come determinare il valore degli attributi
- Valutate in ordine implicito determinato dal grafo delle dipendenze
- Richiedono la costruzione de l'albero sintattico annotato

### SDT

- Possono anche contenere codice arbitrario
- Eseguite in ordine esplicito determinato dalla posizione nel corpo delle prod.
- Eseguite da sx a dx, possono essere integrate nel parser minimo

## CONVERSIONE DA SDD L-ATTRIBUITE A SDT

Per ogni prod  $A \rightarrow X_1 X_2 \dots X_n$

- ① Aggiungi un'azione semantica che calcola il valore degli attributi ereditati subito prima di  $X_i$
- ② In fondo alla produzione aggiungi un'azione semantica che calcola il valore degli attributi sintetizzati da  $A$

## SDT E PARSING RICORSIVO DISCENDENTE

- Il parser ha una procedura per ogni variabile della grammatica che riconosce la stringhe generata da  $A$  nella grammatica
- La procedura  $A$  ha tanti argomenti quanti sono gli attributi ereditati da  $A$  e restituisce tanti valori quanti sono gli attributi sintetizzati da  $A$
- La procedura  $A$  usa il simbolo corrente e gli insiemni guide per scegliere la produzione  $A \rightarrow \alpha_1 | \alpha_2 | \dots | \alpha_n$  da usare per rischiudere  $A$
- Per ogni simbolo o azione semantica  $X$  nel corpo della produzione scelta:
  - Se  $X$  è un simbolo terminale il metodo controlla che il simbolo corrente sia  $X$ , if true fa avanzare il lexer al simbolo successivo, else segnala syntax error
  - Se  $X$  è una variabile invoca la procedura  $X$  passandola come argomenti i suoi attributi ereditati e raccogliendo in variabili locali gli attributi sintetizzati restituiti da  $X$
  - Se  $X$  è un'azione semantica la eseguo

esempio:

$$\begin{array}{ll} S \rightarrow \epsilon & \{S.n = \emptyset\} \\ S \rightarrow aS.b & \{S.n = S_1.n + 1\} \end{array}$$

Stringhe in frame a^n b^n

```
private int S() {
    switch (peek()) {
        case 'a': // S → aSb
        { check('a');
            int S_n = S();
            check('b');
            return S_n + 1; }

        case 'b': // S → ε
        case '$':
            return 0;

        default:
            throw error("S");
    }
}
```

Produzioni	Azioni semantiche
$T \rightarrow F$	$\{T'.e = F.s\}$
$T'$	$\{T.s = T'.s\}$
$T' \rightarrow *F$	$\{T'_1.e = T'.e \times F.s\}$
$T'_1$	$\{T'.s = T'_1.s\}$
$T' \rightarrow \epsilon$	$\{T'.s = T'.e\}$
$F \rightarrow n$	$\{F.s = n.v\}$

```
private int T() {
    switch (peek()) {
        case 'n': { // T → FT'
            int F_s = F();
            int TT_s = TT(F_s);
            return TT_s;
        }
        default:
            throw error("T");
    }
}
```

```
private int TT(int TT_e) {
    switch (peek()) {
        case '*': // T' → *FT'
        { check('*');
            int F_s = F();
            int TT_s = TT(TT_e * F_s);
            return TT_s; }

        case '+':
        case '$': // T' → ε
            return TT_e;
        default:
            throw error("T'");
    }
}

private int F() {
    switch (peek()) {
        case 'n': // F → n
        { int F_s = peek() - '0';
            check('n');
            return F_s; }

        default:
            throw error("F");
    }
}
```

## TRADUZIONE ON-THE-FLY

- gli attributi principali possono essere eliminati "emettendo il voto" gli elementi ausiliari nel punto in cui devono comparire

### DEF DI ATTRIBUTO SINTETIZZATO PRINCIPALE

- Il suo voto include sempre la concatenazione dei valori dello stesso attributo per tutte le variabili nel corpo di ogni re produzione, oltre ad eventuali elementi ausiliari
- La concatenazione rispetta sempre l'ordine delle variabili nel corpo delle produzioni

### esempio di attributo principale

$$E \rightarrow E_1 + T \quad \{E.\text{post} = E_1.\text{post} \parallel T.\text{post} \parallel "+" \}$$

### SDT con accumulo della traduzione

Produzioni	Azioni semantiche
$S \rightarrow E$	$\{\text{print}(E.\text{post})\}$
$E \rightarrow E_1 + T$	$\{E.\text{post} = E_1.\text{post} \parallel T.\text{post} \parallel "+" \}$
$E \rightarrow T$	$\{E.\text{post} = T.\text{post}\}$
$T \rightarrow T_1 * F$	$\{T.\text{post} = T_1.\text{post} \parallel F.\text{post} \parallel "*"\}$
$T \rightarrow F$	$\{T.\text{post} = F.\text{post}\}$
$F \rightarrow (E)$	$\{F.\text{post} = E.\text{post}\}$
$F \rightarrow n$	$\{F.\text{post} = n.v\}$

- $E.\text{post} = E$  in forma postfissa
- $E.\text{post}$  è un attributo principale

### Traduzione "on-the-fly"

Produzioni	Azioni semantiche
$S \rightarrow E$	
$E \rightarrow E_1 + T$	$\{\text{print}("+" )\}$
$E \rightarrow T$	
$T \rightarrow T_1 * F$	$\{\text{print}("*")\}$
$T \rightarrow F$	
$F \rightarrow (E)$	
$F \rightarrow n$	$\{\text{print}(n.v)\}$

- nessun attributo!