Espressioni

Definiamo un linguaggio L come un insieme di stringhe. Per descrivere la sintassi di linguaggi formali (la grammatica), usiamo la **BNF (Backus-Naur Form)**, con questa sintassi:

$$\text{<simbolo> ::= \_espressione\_}$$

Esempio

Conssideriamo la grammatica:

$$M, N ::= 5|7|M + N|M * N$$

Le espressioni che rispettano questa grammatica sono del tipo:
- "5" o "7"
- Un'espressione del tipo $M + N, M * N$ che rispetta a sua volta la grammatica.

Introduciamo una funzione $\text{eval} : L \to \mathbb{N}$ che valuta le espressioni del linguaggio:
- $\text{eval}(5) = 5$
- $\text{eval}(7) = 7$
- $\text{eval}(M + N) = \text{eval}(M) + \text{eval}(N)$
- $\text{eval}(M * N) = \text{eval}(M) * \text{eval}(N)$

Notiamo che nell'esempio precedente l'algebra (L, eval) non è induttiva, infatti una stringa $5 + 7 * 5$ potrebbe essere stata generata in due modi diversi: $(5 + 7) * 5$ e $5 + (7 * 5)$.

Possiamo però considerare $5, 7, +, *$ come costruttori dell'algebra e in questo modo $(5 + 7) * 5 \neq 5 + (7 * 5)$, si potrebbe quindi dimostrare come $(L, 5, 7, +, *)$ è un'algebra induttiva.

Linguaggio Exp

In questo semplice linguaggio indichiamo le espressioni con

$$M, L, N, \ldots ::= 0 \mid 1 \mid \ldots \mid x \mid y \mid z \mid \ldots \mid M + N \mid \text{let x = M in N}$$

Quando usiamo let x = M in N stiamo assegnando un valore alla variabile x all'interno dell'espressione N. Al di fuori di quell'espressione x avrà altri significati.

Ad esempio:
- $\text{let x = 3 in x + x}$ vale 6
- let x = 2 in 10 vale 10

Funzione Free

La funzione free: $\text{Exp} \to P(\text{Var})$, prende in input un'espressione e restituisce l'insieme delle variabili libere, ovvero quelle che non hanno un valore assegnato e sono quindi inutili al calcolo dell'espressione.

Esempi:
- $\text{free}(0) = \{\}$
- $\text{free}(k) = \{\}$ con k una qualsiasi costante
- $\text{free}(x) = \{x\}$
- $\text{free}(M + N) = \text{free}(M) \cup \text{free}(N)$
- $\text{free(let x = M in N) = free(M)} \cup \{\text{free(N) - }\{\text{x}\}\}$

Sintassi e Semantica Astratta

Assumiamo che siano un dati un insieme *Var* di variabili ed un insieme di costanti entrambi numerabili:

- Utilizziamo $x, y, \ldots$ per indicare le variabili
- $k_1, k_2, \ldots$ per le costanti
- M, N per i termini del linguaggio

L'insieme di tutti questi termini è definito induttivamente dalla sintassi astratta:

$$k ::= 5|40|\ldots$$
$$M ::= k|x|M+N|\text{let x} = M \text{ in } N$$

L'operatore *let* ha segnatura:

$$\text{let} : \text{Var} \times \text{Exp} \times \text{Exp} \to \text{Exp}$$

Questo operatore (termine) rappresenta un segmento di codice che definisce la variabile locale x, la inizializza al valore dell'espressione M all'interno del corpo N che può contenere o no dei riferimenti ad x. *Ad esempio:*

$$\text{let } x = 3 + 2 \text{ in } x + x = 10$$

Nell'esempio precedente la variabile x compare due volte in N, si dice che ci sono duie **occorrenze** della variabile, per la x che invece compare subito dopo il *let* si parla di **dichiarazione**.

Se però prendiamo ad esempio l'espressione:

$$\text{let } x = 3 \text{ in } x + \text{let } x = 2 \text{ in } x + x$$

Quante occorrenze di x troviamo? **Dipende**.

L'espressione contiene due variabili con lo stesso nome e dobbiamo quindi capire quante volte compare ciascuna di esse ad esempio specificando meglio la struttura dell'espressione attraverso l'uso di parentesi:

$$\text{let } x = 3 \text{ in } (x + ((\text{let } x = 2 \text{ in } x) + x))$$

In questo caso ci aspettiamo una valutazione di 8 e:

- La x con valore 3 ha 2 occorrenze
- La x con valore 2 ha 1 occorrenza

Espressioni Alfa-Equivalenti

Due espressioni si dicono **alfa-equivalebti** se sono identiche a meno di ridenominazione di variabili legate, ad esempio:

$$\text{let x} = 1 \text{ in } x + 1$$

è alfa equivalente a:

$$\text{let y} = 1 \text{ in } y + 1$$

Domini Semantici

La semantica di *Exp* viene rappresentata usando la nozione di **ambiente**, un ambiente è una funzione **parziale** (non necessariamente definita su tutto il dominio) che associa dei valori ad un insieme finito di variabili. Indichiamo gli ambienti come insiemi di coppie, per esempio l'ambiente E dove z vale 3 e y vale 9 lo scriviamo come $\{(z, 3), (y, 9)\}$.

Il dominio di un ambiente è sempre un sottoinsieme finito di *Var*, in questo caso il dominio è $\{x, y\}$

Concatenazione di Ambienti

Dati due ambienti E_1, E_2, la loro concatenazione indicata da $E_1 E_2$ il cui dominio è $\mathrm{dom}(E_1) \cup \mathrm{dom}(E_2)$ è definita come:

$$(E_1 E_2)(x) = \begin{cases} E_2(x) \text{ se } x \in \mathrm{dom}(E_2) \\ E_1(x) \text{ altrimenti} \end{cases}$$

Ha quindi la precedenza il dominio più a destra.

La **semantica operazionale** di *Exp* è una relazione:

$$\rightsquigarrow \subseteq \mathrm{Env} \times \mathrm{Exp} \times \mathrm{Val}$$