

A questo punto introduciamo le regole di transizione. Facciamo attenzione a distinguere tra il simbolo sintattico di una operazione (bop in grassetto rosso) e il simbolo dell'operazione nella macchina sottostante (bop normale).. di questo non ci interessa l'implementazione, ma solo che lo possiamo usare per interpretare il simbolo del linguaggio. Qui eseguiamo astrazione. Partiamo dalle regole necessarie per valutare espressioni aritmetiche.

$$\mathcal{E}_1: m \text{ op } n \rightarrow p \quad \text{se} \quad m \text{ op } n = p, \\ m, n, p \in \mathcal{N}$$

La prima regola è un assioma, che semplicemente valuta l'espressione sintattica contenente un operatore aritmetico, nel valore che esso rappresenta. Ad esempio,  $3+5$  è una scrittura sintattica dove il simbolo 3 rappresenta il valore 3, il simbolo 5 rappresenta il simbolo 5, e quindi interpretando il simbolo della somma come addizione restituiamo il simbolo 8 che rappresenta il valore 8.

$$\mathcal{E}_2: \frac{e \rightarrow e'}{e \text{ op } e_0 \rightarrow e' \text{ op } e_0}$$

La seconda regola è invece una regola induttiva di valutazione. Questa regola mi dice che se gli operandi non sono valori primitivi allora dobbiamo valutarli. In particolare, questa regola mi dice che prima valutiamo l'espressione a sinistra dell'operatore. La regola successiva stabilisce invece che, nel momento in cui l'operando a sinistra è un valore allora posso iniziare a valutare l'operando a destra.

Chiaro che quando anche l'operatore a destra è un valore allora ricadiamo nell'assioma e possiamo restituire il valore finale.

$$\mathcal{E}_3: \frac{e \rightarrow e'}{m \text{ op } e \rightarrow m \text{ op } e'}$$