

Progetto PR2 Seconda parte

Alessandro Puccia

1 Traccia

Considerando il linguaggio didattico funzionale, si richiede di progettare un'estensione che permetta di manipolare come dati primitivi dizionari di elementi, ovvero una collezione di coppie (chiave, valore), assumendo che la chiave sia un identificatore.

2 Sintassi concreta

$\text{Dict} ::= \{ \} \mid \{ \text{Item} \}.$

$\text{Item} ::= \text{Ide} : \text{Exp}, \text{Item} \mid \text{Ide} : \text{Exp}.$

$\text{Exp} ::= \dots$

$\text{Dict} \mid \text{Select}(\text{Exp}, \text{Ide}) \mid \text{Insert}(\text{Exp}, \text{Ide}, \text{Exp}) \mid \text{Remove}(\text{Exp}, \text{Ide}) \mid \text{Clear}(\text{Exp}) \mid$
 $\text{ApplyOver}(\text{Exp}, \text{Exp}).$

3 Scelte progettuali

Il typechecker adottato è dinamico.

Nei casi in cui si voglia definire un dizionario con chiavi ripetute viene lanciata un'eccezione, se si volesse inserire una coppia con campo chiave già presente invece, viene aggiornato il campo valore della coppia all'interno del dizionario con il nuovo valore passato.

Nel caso in cui si usi la remove di una chiave, e la stessa non compaia nel dizionario passato, allora non viene sollevata un'eccezione ma semplicemente si restituisce il dizionario immutato.

4 Parte opzionale

Per la parte opzionale ho creato un nuovo file (`dinamico.ml`) che contiene `rt_eval`, l'unica differenza con `eval` riguarda le chiusure, in quanto non verrà salvato l'ambiente di definizione (vengono valutate nell'ambiente attuale).