

Relazione OCAML Dictionary Interpreter [Corso A]

Ziccolella Francesco Giuseppe

Matricola : 549220

L'ambiente è definito come una lista polimorfa (`string * 't`) list e il type checking dei tipi viene eseguito dinamicamente. I casi di test sono inclusi nel file `mydict.ml`

Considerando le funzioni chieste nella traccia ho esteso alle espressioni “exp” del linguaggio dei costruttori che potessero eseguire ciò che è stato chiesto:

- **Dict of (ide * exp) list** (definisco il dizionario come una lista di coppie ide, value)
- **Filt_ of ide list**, per gestire una lista di ide e usarla per la funzione Filter
- **Check exp * ide * exp**, per controllare se gli elementi del dizionario sono stati modificati seguendo le funzioni scritte successivamente

Remove of exp * ide

- **Update of exp * ide * exp**, per poter aggiornare il valore di una etichetta con un nuovo valore
- **Remove of exp * ide**, elimina l'elemento con la key = ide
- **Has_key of exp * ide**, mi controlla se un elemento del dizionario ha la key che sto cercando
- **Clear of exp**, rende vuoto il dizionario selezionato
- **Iterate of exp * exp**, applica una funzione exp ad un dizionario exp
- **Filter of exp * (ide list)**, estraggo dal dizionario gli elementi con le key presenti nel filtro (lista di identificatori)
- **Fold of exp * ide * exp**, *Applica una funzione sequenzialmente e aggiorna i valori delle chiavi

L'implementazione del dizionario è basata su una lista di coppie (key,value) utilizzando le liste native di OCaml, poichè erano comode da implementare.

Rep Invariant :

- Ogni elemento nella lista è univoco rispetto la key
- la Iterate può essere applicata solo se il tipo degli elementi è uguale e la funzione è applicabile al tipo in comune

Il valore è inizialmente exp e viene valutato tramite eval nell'ambiente corrente