



Relazione progetto OCaml del corso di Programmazione 2

INSEGNANTE: FRANCESCA LEVI

DAVIDE CHEN | MATRICOLA N° 544795 | CORSO B

Capitolo 1

1.1 OVERVIEW

Il progetto consiste nella progettazione e realizzazione di un interprete per un semplice linguaggio di programmazione funzionale didattico (come da lezione) esteso alla manipolazione sui dizionari, ovvero una collezione di coppie <chiave, valore>.

1.2 DETTAGLI DI IMPLEMENTAZIONE

Un dizionario è espresso dal tipo **Dictionary of dict** dove *dict* è un tipo che può assumere il valore *Empty* oppure il valore *Item of id * exp * dict*. Questo significa che il dizionario viene rappresentato da una concatenazione dei valori del tipo *dict*. Ad esempio il dizionario:

$$\{ \text{"mele": } 430, \text{ "banane": } 312 \}$$

Può essere definito nel seguente modo:

$$\text{Dictionary}(\text{Item}(\text{mele}, \text{Eint}(430), \text{Item}(\text{"banana"}, \text{Eint}(312), \text{Empty})))$$

L'interprete è fatto in modo che non esistono chiavi duplicati nel dizionario, di conseguenza l'inserimento (**Insert**) di un nuovo elemento in un dizionario può essere fatto se e solo se l'elemento non esiste già. In caso di inserimento di una chiave già presente l'interprete restituirà un *errore* alla sua prima occorrenza, altrimenti scorrerà tutta il dizionario e inserirà l'elemento nell'ultima posizione.

Per l'operazione di rimozione (**Delete**) avviene con un ragionamento simile controllando ogni chiave del dizionario eliminando l'elemento solo se trova un riscontro positivo, altrimenti scorrerà su tutto il dizionario senza fare nulla.

La funzione di controllo **Has_Key** prende una chiave ed un dizionario e restituisce *true* se trova una chiave nel dizionario che è uguale alla chiave passata per argomento, *false* se arriva a fine dizionario senza aver trovato nulla.

La funzione **Iterate** è una operazione che per ogni elemento del dizionario invoca la funzione passandogli l'elemento per argomento e lo sostituisce con il valore restituito dalla funzione.

L'operazione **Fold**, oltre ad applicare la funzione agli elementi del dizionario come avvenuto con *Iterate* fa anche la somma dei nuovi valori ottenuti.

L'operazione **Filter** usa l'ausilio della funzione controllo *memberList* per controllare se le chiavi del dizionario sono presenti nella lista dei elementi da filtrare, se contiene lo inserisce nel nuovo dizionario da restituire, altrimenti fa la chiamata ricorsiva su se stessa omettendo l'ultimo elemento.

1.2 BATTERIA DI TEST

La batteria di test viene realizzato “a mano” in un secondo file, andando a creare come prima cosa un nuovo ambiente vuoto. Dopodiché segue la creazione di un dizionario vuoto e uno con valori e testa su quest’ultimo tutte le funzioni precedentemente menzionate.