

# Programmazione 2

Relazione secondo progetto intermedio

Gabriele Sergi

Corso A

Per il progetto in Ocaml dell'anno accademico 2018-19 vado a descrivere l'implementazione a cui ho pensato.

Inizialmente ho definito la rappresentazione astratta del dizionario come **My\_dict of el**

Dove el = Empty | DictEl of ((ide \* exp) \* el)

Dopo aver definite il dizionario, ho creato le seguenti funzioni che vado a spiegare:

- **Add(exp\*ide\*exp)** -> Restituisce un dizionario che ha in cima un nuovo elemento con chiave ide e valore secondo exp, e in coda il dizionario rappresentato dal primo exp.  
Permette l'inserimento di duplicati;
- **Show(exp\*ide)** -> Restituisce il valore dell'elemento che ha come chiave ide all'interno di exp. Se non viene trovata nessuna chiave che corrisponde ad ide, lancia un errore;
- **Modify(exp\*ide\*exp)** -> Restituisce un nuovo dizionario nel quale il valore della chiave corrispondente ad ide all'interno del dizionario rappresentato dal primo exp, viene sostituito con il valore del secondo exp.  
Come prima in caso di ricorrenze multiple della chiave, modifica il valore della prima mente lancia un errore nel caso in cui non ne trovi neanche una;
- **Remove(exp\*ide)** -> Restituisce un dizionario senza la prima occorrenza dell'elemento che ha come chiave ide, se non esiste alcun elemento con quella chiave all'interno di exp viene restituito un errore;
- **Clear(exp)** -> Restituisce un dizionario vuoto, dato che i dizionari sono immutable non modifica la rappresentazione precedente;
- **ApplyOver(exp\*exp)** -> data una funzione, questa viene applicata al dizionario nel caso in cui tutti gli elementi abbiano un valore dello stesso tipo che deve corrispondere a quello della funzione.  
Nel caso in cui ci siano dei tipi differenti oppure che i tipi non corrispondano a quelli richiesti dalla funzione, viene lanciato un errore.

Per testare queste funzioni, ho creato dei test che permettono la verifica dei meccanismi di base appena descritti.

Nel caso dei test dedicati alla funzione di ApplyOver, ho esteso l'interprete con un nuovo tipo Estring e ho aggiunto un'operazione di concatenazione di stringhe allo scopo di dimostrare come si puo' operare anche su questo tipo.

Ho mantenuto un Type-Checking dinamico ed uno Scoping statico.