

# RELAZIONE SECONDO PROGETTO INTERMEDIO

Matteo Biviano – Corso B – Matricola 543933

## 1 Valutazione delle espressioni

Per la valutazione delle estensioni richieste sono state usate delle funzioni ausiliarie per semplificare la visione d'insieme dell'interprete.

- **ETree tr**: viene valutata passando l'albero *tr* e l'ambiente *r* ad una funzione ausiliaria *eval\_t*, per essere trasformata in un *evTree*.

Tutte le espressioni successive falliranno la loro valutazione nel caso in cui l'espressione *tr* non verrà interpretata come tipo albero.

- **ApplyOver(ex, tr)**: viene valutata l'espressione *tr*. L'albero risultante *t* viene visitato attraverso la funzione *apply\_o*, la quale applica la funzione denotata da *ex* al valore di ogni nodo di *t*.

- **Update(lis, ex, tr)**: viene valutata l'espressione *tr*. La funzione ausiliaria *update* applica la funzione denotata da *ex* solo al valore del nodo (o dei nodi) dell'albero *t*, identificato dal cammino *lis*. Questo viene eseguito attraverso il seguente algoritmo:

i) se *lis* contiene un solo elemento e quest'ultimo risulta essere uguale al *tag* della radice dell'albero, viene applicata *ex* al valore della stessa;

ii) se *lis* contiene più di un elemento e il primo elemento coincide con la radice dell'albero, viene applicato il procedimento (sopra esposto) ricorsivamente per i due rami di *t*, escludendo il primo elemento di *lis*;

iii) se il cammino è vuoto o se nessun nodo corrisponde al cammino indicato, non viene effettuato nessun aggiornamento.

- **Select(lis, ex, tr)**: viene valutata l'espressione *tr*. La funzione ausiliaria *select* opera in modo simile alla funzione *update*.

Nel caso in cui *lis* contenga un unico elemento, viene valutato il predicato, denotato da *ex*, sulla radice; memorizzando la valutazione appena eseguita in una variabile *g*. Nell'eventualità in cui la valutazione del predicato restituisce un valore booleano, l'algoritmo prosegue; altrimenti il programma solleva un'eccezione *Failure* con il messaggio: "*non boolean guard*". Se la variabile *g* risulta essere *true* e contemporaneamente il *tag* della radice di *t* è uguale alla testa della lista, allora viene restituito l'albero *t* (valutato).

Nel caso in cui *lis* contenga più di un elemento e il primo elemento coincide con la radice dell'albero, viene eseguito l'algoritmo ricorsivamente sui due sottoalberi corrispondenti (scartando il primo elemento della lista). Qualora i risultati siano vuoti, viene restituito l'albero vuoto, in alternativa uno dei due non-vuoti (dando precedenza al secondo).

In condizioni di lista vuota e/o albero vuoto viene restituito l'albero vuoto.

## 2 Type checker dinamico

Il type checker dinamico è stato esteso con la possibilità di riconoscere valori di tipo albero, aggiungendoli all'insieme dei possibili valori rappresentabili.

## 3 Esecuzione del codice

L'esecuzione del programma avviene attraverso il comando

```
$ ocaml < Interprete_tree.ml
```