

# Relazione secondo progetto intermedio

---

Marco Costa

545144

## 1 SCELTE PROGETTUALI

Il progetto è stato svolto partendo da un interprete OCaml con presenza di scoping statico assemblato attraverso le varie sezioni delle note fornite.

Le più importanti scelte progettuali effettuate sono le seguenti:

- Poiché la struttura presentava primitivamente type-checking dinamico sulle operazioni di base, e poiché il tipo esprimibile *EvalTree* è stato definito come:

```
type evT = (* *)
          | EvalTree of evNode
and evNode = Empty | ENode of ide * evT * evNode * evNode ;;
```

nel quale viene forzato grammaticalmente l'utilizzo di un *evNode* come figlio (a differenza di una soluzione del tipo *EvalTree of ide \* evT \* evT \* evT*) non è richiesta la scrittura di un meccanismo di typechecking dinamico ad esclusione dei *pattern matching* effettuati sui parametri.

- L'albero è polimorfo rispetto alle espressioni contenute nei nodi, tuttavia, le funzioni applicate su di esso devono essere compatibili col tipo dell'espressione.
- Per motivi di complessità non è stato esteso l'interprete per la gestione di funzioni di ordine superiore, infatti, ad esempio, l'esecuzione di *ApplyOver* su un nodo con una chiusura come espressione, fallirà.
- Dato che la funzione *Select* non è deterministica rispetto all'albero da restituire, in caso di due o più cammini determinati dagli stessi identificatori è stata data priorità al sotto-albero proveniente dal cammino maggiormente a sinistra.

## 2 TEST

Il codice può essere provato interattivamente da shell mediante comando:

```
$ ocaml
# #use "source.ml";; (*compilazione dell'interprete*)
# #use "test.ml";; (*compilazione ed esecuzione delle batterie di test*)
```

a partire dalla directory contenente il codice sorgente.