

Esercitazione di Linguaggi e Compilatori

Modulo Linguaggi - - AA 2011-2012

18 Gennaio 2012

1. Verificare che il tipo della funzione ML

```
fun f g h = (+ (g(+ h 1))) h;
```

è $(\text{int} \rightarrow \text{int}) \rightarrow \text{int} \rightarrow \text{int}$ (si assuma che il tipo di $+$ sia $\text{int} \rightarrow \text{int} \rightarrow \text{int}$).

2. Si rappresenti in ML (usando `datatype`) il tipo `itree` degli alberi binari con i nodi etichettati con interi (non è detto che ogni nodo interno abbia due figli). Si scriva una funzione ML che dato un albero t di tipo `itree` e due interi x, y restituisca il nuovo albero ottenuto sottraendo il valore x dall'etichette maggiori di y .

3. Verificare che la funzione

```
fun f g h r = (+ (g(h))) (+ h r);
```

ha tipo $(\text{int} \rightarrow \text{int}) \rightarrow (\text{int} \rightarrow (\text{int} \rightarrow \text{int}))$ (si assuma che il tipo di $+$ sia $\text{int} \rightarrow \text{int} \rightarrow \text{int}$).

4. Si rappresenti in ML (usando `datatype`) il tipo `btree` degli alberi binari con i nodi etichettati con interi (non è detto che ogni nodo interno abbia esattamente due figli). Si scriva una funzione ML che dato un albero t di tipo `btree` restituisca l'altezza dell'albero.

5. Si rappresenti in ML (usando `datatype`) il tipo `btree` degli alberi binari con le foglie etichettate con booleani, tali che ogni nodo interno abbia esattamente due figli. Si scrivano due funzioni `left` e `right` che, dato un albero t di tipo `btree` restituiscano rispettivamente il sottoalbero sinistro e destro (se questi esistono), altrimenti restituiscano t .

6. Si rappresenti in ML (usando `datatype`) il tipo `itree` degli alberi binari con i nodi etichettati con interi. Si scriva una funzione ML che dato un albero t di tipo `itree` restituisca la sommatoria degli elementi pari in t .

7. Definire un tipo di dato per le espressioni aritmetiche basate su letterali, variabili, somma e prodotto. Scrivere un valutatore in ML.

8. Definire un tipo di dato per i comandi **skip**, **if**, **while**, composizione e assegnamento. Scrivere un valutatore in ML.

9. Implementare in ML le funzioni insiemistiche di

- appartenenza
- sottinsieme
- uguaglianza
- unione
- intersezione.

Si assuma che un insieme sia rappresentato mediante una lista in cui ogni elemento appare una sola volta (l'ordine non ha importanza).

10. Definire in ML una funzione `merge` che prende come argomenti due liste ordinate e restituisce la lista ordinata risultante dalla loro unione. La lista restituita può contenere elementi ripetuti.