

PROGRAMMAZIONE 1 e LABORATORIO (A,B) - a.a. 2009-2010

Prova scritta del 7 settembre 2010

Scrivere **IN STAMPATELLO** COGNOME, NOME e CORSO su ogni foglio consegnato

ESERCIZIO 1 (punti 5)

Dato l'alfabeto $\Lambda = \{a, b\}$ si definisca una grammatica che genera un linguaggio \mathcal{L} che soddisfa la seguente proprietà:

$$\emptyset \subset \{ab^m \mid m > 0\} \subset \mathcal{L} \subset \{a^n b^m \mid n, m > 0\} \subset \Lambda^*$$

ESERCIZIO 2 (punti 5)

Dato il tipo degli alberi binari visto a lezione

```
type 'a btree = Empty | Node of 'a * 'a btree * 'a btree
```

si definisca una funzione

```
extract_leafs : 'a btree -> 'a list
```

in modo che `(extract_leafs bt)` restituisca la lista che contiene i valori di tutte le foglie, presi **da destra verso sinistra**.

ESERCIZIO 3 (punti 5)

Si definisca in C una funzione

```
boolean check (int a[], int dim)
```

che restituisce `true` se vale la seguente proprietà

$$\exists i \in [2, dim - 2]. (\forall j \in [i - 2, i + 2]. j \neq i \Rightarrow a[i] > a[j])$$

e restituisce `false` altrimenti (si supponga dato il tipo `typedef enum {false, true} boolean;`).

ESERCIZIO 4 (punti 5)

Senza utilizzare ricorsione esplicita, si definisca in CAML una funzione

```
foo : 'a -> 'a list -> bool
```

in modo che `(foo x lis) = true` se e solo se `x` compare nella lista `lis`.

ESERCIZIO 5 (punti 5)

Un *multiinsieme* è una collezione di elementi in cui ciascun elemento può occorrere più volte. Ad esempio, $\{a, b, a, a, c, b\}$ è un multiinsieme di caratteri in cui a occorre tre volte, b due volte e c una volta. Un multiinsieme può essere rappresentato in C mediante una lista di coppie: ciascuna coppia nella lista rappresenta l'elemento del multiinsieme e il numero di sue occorrenze nel multiinsieme. Siano date allora le seguenti definizioni per la rappresentazione di multiinsiemi di caratteri

```
struct el { char c; int occ; struct el *next; }  
typedef struct el ElementoMultiinsieme;  
typedef ElementoMultiinsieme *Multiset;
```

Si scriva in C una funzione che, dato un multiinsieme di caratteri m , restituisca il numero di caratteri che occorrono più di una volta in m .

ESERCIZIO 6 (punti 5)

Definire in C il tipo delle liste di elementi di un generico tipo `T` e scrivere una procedura che, data una lista di elementi di tipo `T`, metta il primo elemento in ultima posizione. **Nota bene:** la procedura non può effettuare assegnamenti su campi o variabili di tipo `T`, dal momento che su di esso potrebbe non essere definita l'operazione di assegnamento.