

# PROGRAMMAZIONE 1 e LABORATORIO (A,B) - a.a. 2013/14

## Prova scritta del 9 gennaio 2014

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

### ESERCIZIO 1 (punti 5)

Definire una grammatica libera che genera il seguente linguaggio sull'alfabeto  $\Lambda = \{a, b\}$

$$\mathcal{L} = \{a^n a b^{n+1} \mid \alpha \in \Lambda^*, n > 0\}$$

### ESERCIZIO 2 (punti 5)

Definire un automa **deterministico** che riconosce il seguente linguaggio sull'alfabeto  $\Lambda = \{a, b\}$

$$\mathcal{L} = \{a^{2n+1}(ba)^{2k} \mid n, k \geq 0\}$$

**N.B.:** la notazione  $(ba)^x$  indica la stringa  $ba$  ripetuta  $x$  volte.

### ESERCIZIO 3 (punti 5)

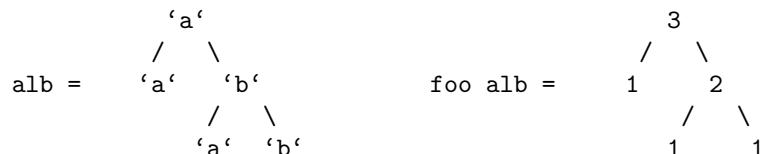
Dato il tipo degli alberi binari

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

si definisca in CAML una funzione `foo` con tipo

```
foo : 'a btree -> int btree
```

in modo che `(foo bt)` sia un albero di interi con la stessa struttura di `bt` in cui ogni nodo contiene il numero di occorrenze del corrispondente nodo in `bt` nel sottolabero di cui è radice in `bt`. Ad esempio:



### ESERCIZIO 4 (punti 5)

Si definisca in C una funzione

```
int check (int a[], int dima, int b[], int dimb)
```

che restituisce il valore di verità della seguente formula:

$$\exists k \in [0, dima]. ((\forall i \in [0, k]. \exists j \in [0, dimb]. a[i] = b[j]) \wedge (\forall i \in [k, dima]. \forall j \in [0, dimb]. a[i] \neq b[j]))$$

### ESERCIZIO 5 (punti 5)

Senza utilizzare ricorsione esplicita, definire in CAML una funzione

```
foo : 'a list -> 'a -> 'a * int
```

in modo che `(foo lis x)` restituisca la coppia `(y, n)` dove `n` è il numero di elementi di `lis` uguali a `x` e `y` è l'ultimo elemento di `lis` diverso da `x`, se tale elemento esiste, `x` stesso altrimenti.

### ESERCIZIO 6 (punti 5)

Date le seguenti definizioni:

```
struct el {int info; struct el *next;};
typedef struct el ElementoDiLista;
typedef ElementoDiLista *ListaDiElementi;
```

scrivere in C una procedura che, dati in ingresso attraverso opportuni parametri una lista di interi ed un intero `x`, elimina dalla lista i primi `x` elementi maggiori di 0 (se la lista contiene meno di `x` elementi maggiori di 0 la procedura li deve eliminare tutti).