

# PROGRAMMAZIONE 1 e LABORATORIO (A,B) - a.a. 2012/13

## Prova scritta del 6 giugno 2013

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

### ESERCIZIO 1 (punti 5)

Si definisca una grammatica libera che genera il linguaggio

$$\mathcal{L} = \Lambda^* \setminus \{a^n b^n \mid n \geq 0\}$$

sull'alfabeto  $\Lambda = \{a, b\}$ .

### ESERCIZIO 2 (punti 5)

Si considerino le seguenti definizioni di tipo:

```
type 'a unodue = Uno of 'a | Due of 'a * 'a
type 'a mseq = Single of 'a unodue | Mult of 'a unodue * 'a mseq
```

Si definisca una funzione `count` con tipo

```
count : 'a mseq -> int
```

in modo che `(count m)` restituisca il numero di elementi di tipo `'a` presenti in `m`.

### ESERCIZIO 3 (punti 5)

Dato un array `a` di dimensione `dim` sia:

$$mscart(a) = \max\{|a[i] - a[j]| \mid i, j \in [0, dim) \wedge i \neq j\}$$

Si definisca in C una funzione

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

che, dati un array `a`, la sua dimensione `dim` e un intero `d`, restituisce il valore di verità `mscart(a) < d`.

(N.B.  $|x - y|$  indica la differenza in valore assoluto tra  $x$  e  $y$ .)

Si assuma predefinito il tipo `typedef enum {false, true} boolean`.

### ESERCIZIO 4 (punti 5)

Si completi la seguente definizione

```
let foo p lis = let f x y = .... in foldr f ... lis
```

della funzione

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

in modo che `(foo p lis)` restituisca:

- il primo elemento di `lis` che soddisfa `p`, se tale elemento esiste
- il primo elemento di `lis`, altrimenti

### ESERCIZIO 5 (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, data in ingresso una lista di interi, inserisce in testa alla lista un nuovo elemento che contiene la somma di tutti gli interi della lista.

### ESERCIZIO 6 (punti 5)

Si definisca una procedura C che, dato un array di interi `a` di dimensione `dim`, sostituisce ogni elemento con la somma tra l'elemento stesso e tutti gli elementi che lo precedono in `a`.