

PROGRAMMAZIONE 1 e LABORATORIO (A,B) - a.a. 2014-2015

Prova scritta del 4/02/2016

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

ESERCIZIO 1 (6 punti)

Dato l'alfabeto $\Lambda = \{a, b, c\}$ si definisca una grammatica **regolare** che genera il linguaggio delle stringhe di Λ^+ la cui lunghezza è un multiplo di 3.

ESERCIZIO 2 (6 punti)

Si definisca in CAML, senza utilizzare ricorsione esplicita, una funzione

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

in modo che `(foo lis)` restituisca una terna `(max, n, m)` in cui `max` è il valore massimo presente nella lista, `n` è il numero di occorrenze di `max` nella lista e `m` è il numero di elementi della lista. La funzione non è definita se la lista è vuota.

N.B. Non è consentito l'utilizzo della funzione `length`.

ESERCIZIO 3 (6 punti)

Si scriva in C una procedura che, dati attraverso opportuni parametri una lista di interi ed un intero `n`, sposta in testa alla lista tutti gli elementi maggiori di `n`.

Si suppongano predefiniti i tipi

```
struct el {int info; struct el *next;};

typedef struct el ElementoDiLista;
typedef ElementoDiLista *ListaDiInteri;
```

ESERCIZIO 4 (6 punti)

Si scriva una funzione C

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

che, dati due array di interi `a`, `b` e le loro dimensioni `dima`, `dimb`, restituisce il valore di verità della seguente formula

$$\exists i \in [0, dima). ((\forall j \in [0, dimb). a[i] \neq b[j]) \wedge \#\{k \mid k \in [0, dima) \wedge a[i] = a[k]\} = 1)$$

Si ricorda che, dato un insieme finito A , $\#A$ denota il numero di elementi di A .

ESERCIZIO 5 (6 punti)

Dato il tipo degli alberi binari

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

definire una funzione

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

in modo che `(leafsdepth bt n)` restituisca il numero di foglie dell'albero che occorrono a profondità `n`.