

PROGRAMMAZIONE 1 e LABORATORIO (A,B) - a.a. 2012-2013

II Verifica scritta del 19/12/2012

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

ESERCIZIO 1 (6 punti)

Dato il tipo degli alberi binari

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

si definisca in CAML una funzione `check` con tipo

```
check : 'a btree -> int -> bool
```

in modo che `(check bt n)` restituisca `true` se sono al più `n` i nodi di `bt` con due figli, `false` altrimenti.

ESERCIZIO 2 (6 punti)

Scrivere in C una funzione

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

che, dati due array `a` e `b` di dimensione `dima` e `dimb` rispettivamente, verifica che ogni elemento di `a` occorra in `b` un numero dispari di volte.

ESERCIZIO 3 (6 punti)

Scrivere in C una procedura

```
void azzera (int a[], int dim)
```

che azzera ogni elemento dell'array minore della somma degli elementi che lo precedono, fino al primo elemento uguale a 0.

Dato ad esempio il seguente array `vet1`

-1	6	8	7	3	0	5	3
----	---	---	---	---	---	---	---

la chiamata `azzera(vet1,8)` deve modificare `vet1` come segue

0	6	8	0	0	0	5	3
---	---	---	---	---	---	---	---

Dato invece il seguente array `vet2`

2	1	8	20	3
---	---	---	----	---

la chiamata `azzera(vet2,5)` deve modificare `vet2` come segue

2	0	8	20	0
---	---	---	----	---

ESERCIZIO 4 (6 punti)

Date le seguenti definizioni:

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

```
typedef struct el ElementoDiLista;
```

```
typedef ElementoDiLista *ListaDiInteri;
```

scrivere in C una procedura che, data in ingresso attraverso un opportuno parametro una lista di interi, antepone ad ogni elemento negativo un elemento nullo.

ESERCIZIO 5 (6 punti)

Senza utilizzare ricorsione esplicita, definire in CAML una funzione

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

in modo che `(foo lis p x)` restituisca la coppia `(w,n)` dove `n` è il numero di elementi di `lis` che soddisfa `p` e `w` è il primo elemento di `lis` che soddisfa `p`, oppure la coppia `(x,0)` se nessun elemento di `lis` soddisfa `p`.