

Fondamenti di Programmazione - CdL in MATEMATICA

II Prova di verifica del 25/5/2010

Scrivere **in stampatello** COGNOME, NOME e MATRICOLA su ogni foglio consegnato

N.B.: Negli esercizi di programmazione, viene valutata anche la leggibilità del codice proposto. Laddove è utilizzato, il tipo `boolean` è definito da `typedef enum {false, true} boolean;`

n. eserc.	1	2	3	4
punt. tot	6	7	8	11

ESERCIZIO 1 (6 punti)

- Costruire l'automa minimo equivalente a quello dato dalla seguente tabella di transizione:

	0	1
→ q0	q1	q2
q1	q0	q3
q2	q4	q3
q3	q1	q5
★ q4	q4	q5
q5	q4	q3

- Scrivere la corrispondente grammatica regolare.

ESERCIZIO 2 (7 punti)

Senza usare costrutti iterativi, né strutture di appoggio come liste o array, si scriva un programma che legga una sequenza di interi $a_0, a_1, \dots, a_i, \dots$, terminata da -1 e stampi la sequenza derivata $b_0, b_1, \dots, b_i, \dots$, definita nel modo seguente:

- $b_0 = a_0$,
- $b_i = (a_{i-1} + a_i)$ per $i > 0$

ESERCIZIO 3 (8 punti)

In un *quadrato magico* (vedi esempio), la somma dei numeri presenti in ogni riga, in ogni colonna e in entrambe le diagonali dà sempre lo stesso numero, detto *somma magica*.

8	1	6
3	5	7
4	9	2

Assumendo che N sia una costante naturale, si definisca una funzione con prototipo

```
boolean ismagic (int mat[][N])
```

che stabilisca se la matrice `mat` rappresenta o meno un quadrato magico.

ESERCIZIO 4 (11 punti)

Si vuole modellare un'asta su base mensile mediante una lista concatenata, in cui ogni elemento rappresenta un'offerta con:

- un campo relativo alla cifra
- un campo che riporta il giorno del mese relativo alla consegna
- un campo numerico che rappresenta *univocamente* l'offerente.

Le offerte vengono inserite seguendo l'ordine di consegna.

- Definire i tipi opportuni per rappresentare un'asta.
- Un'offerta è *valida* se la cifra è superiore alla base d'asta b e il giorno della consegna è inferiore al giorno s di scadenza dell'asta. Scrivere una procedura *ricorsiva* che, dati un'asta, la base d'asta e il giorno di scadenza, elimini dalla lista tutte le offerte non valide.
- Scrivere una funzione *iterativa* che data un'asta (che si suppone contenere solo offerte valide), restituisca il puntatore all'offerta vincitrice, ovvero quella in cui la cifra è più alta e, a parità di cifra, quella consegnata prima (si ipotizza che non possano esistere casi di ex-aequo).