

# FONDAMENTI DI PROGRAMMAZIONE - CdL in MATEMATICA

## PROVA SCRITTA DEL 17/9/2009

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. Inoltre, non è consentito l'uso di istruzioni che alterino il normale flusso dell'esecuzione all'interno di cicli che ne provochino l'uscita forzata).

Laddove è utilizzato, il tipo `boolean` è definito da `typedef enum {false, true} boolean;`

### ESERCIZIO 1 (7 punti)

- Definire un automa a stati finiti deterministici (DFA) che dato l'alfabeto  $\Sigma = \{a, b, c, \dots, z\}$  riconosca **tutte** le parole che contengono la sottostringa **man**.
- Definire un automa a stati finiti non deterministici (NFA) che dato lo stesso alfabeto riconosca le parole che contengono la sottostringa **man**.
- Attraverso la costruzione per sottoinsiemi trovare l'automa a stati finiti deterministici (DFA) corrispondente.

### ESERCIZIO 2 (7 punti)

Definire una procedura **ricorsiva** che, dato un vettore  $A$  di interi distinti di dimensione non nulla e la sua lunghezza, lo modifica facendo in modo che ogni elemento  $A[i]$  assuma il valore della somma degli elementi precedenti  $\sum_{j=0}^i A[j]$  (compreso l'elemento stesso). Nella soluzione non possono essere utilizzati vettori di supporto, né strutture dinamiche.

Dato ad esempio il vettore iniziale

|   |   |   |   |   |   |
|---|---|---|---|---|---|
| 5 | 4 | 3 | 7 | 8 | 2 |
|---|---|---|---|---|---|

la procedura lo deve modificare come segue:

|   |   |    |    |    |    |
|---|---|----|----|----|----|
| 5 | 9 | 12 | 19 | 27 | 29 |
|---|---|----|----|----|----|

### ESERCIZIO 3 (10 punti)

Definire una funzione `SudokuSimple` che, data una matrice  $9 \times 9$  di interi da 1 a 9 (0 sta per vuoto), controlli che in tutte le matrici  $3 \times 3$  non sovrapposte, gli elementi, tranne lo 0, siano tutti distinti.

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| 4 | 7 | 3 | 5 | 9 | 8 | 2 | 6 | 1 |
| 6 | 5 | 2 | 3 | 1 | 4 | 8 | 7 | 9 |
| 8 | 9 | 1 | 6 | 2 | 7 | 4 | 3 | 5 |
| 7 | 6 | 4 | 8 | 5 | 3 | 9 | 1 | 2 |
| 1 | 3 | 9 | 4 | 6 | 2 | 5 | 8 | 7 |
| 5 | 2 | 8 | 9 | 7 | 1 | 6 | 4 | 3 |
| 3 | 4 | 7 | 2 | 8 | 5 | 1 | 9 | 6 |
| 2 | 1 | 8 | 7 | 4 | 9 | 3 | 5 | 8 |
| 9 | 8 | 5 | 1 | 3 | 6 | 7 | 2 | 4 |

### ESERCIZIO 4 (7 punti)

Definire una procedura `Merge` che riceve come argomento due liste ordinate di tipo intero e restituisce una lista di tipo intera, ordinata, contenente tutti gli elementi delle due liste senza ripetizioni.