

# Fondamenti di Programmazione - CdL in MATEMATICA

## I Appello del 1/7/2013

num. eserc.	1	2	3	4
punt. tot	7	7	6	10

### 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 (come, ad esempio, `continue`, `break` e istruzioni di `return` all'interno di cicli che ne provochino l'uscita forzata).
- Non è consentito l'uso di variabili statiche.
- Laddove è utilizzato, il tipo `boolean` è definito da `typedef enum {false, true} boolean`.

### ESERCIZIO 1 (7 punti)

- Si fornisca il **minimo** automa a stati finiti deterministico che riconosce il seguente linguaggio  $L$ :

$$L = \{w \in \{a, b\}^* \mid \text{il numero di occorrenze di } ab \text{ in } w \text{ è pari}\}$$

- Si scriva l'**espressione regolare** ottenuta dall'automata con l'algoritmo di eliminazione di stati, nella parte in cui si elimina tutti gli stati tranne lo stato iniziale.
- Si fornisca il **minimo** automa a stati finiti deterministico che riconosce il seguente linguaggio  $L'$ :

$$L = \{w \in \{a, b, c\}^* \mid \text{il numero di occorrenze di } ab \text{ in } w \text{ è pari}\}$$

### ESERCIZIO 2 (7 punti)

Scrivere una funzione **ricorsiva** che, legga una sequenza di interi da input, terminata da 0 e che restituisca il numero di tutti gli elementi che sono uguali alla somma dell'elemento precedente e di quello successivo nella sequenza. Ad esempio se la sequenza è 2 7 5 4 8 4 0, la funzione deve restituire 2, dato che 7 = 2 + 5 e che 8 = 4 + 4.

### ESERCIZIO 3 (6 punti)

Scrivere una funzione **iterativa** che, dato un array `a` di dimensione `dim`, restituisce il valore di verità della seguente formula:

$$\exists i. i \in [1, dim - 1) \wedge \sum_{j=0}^{i-1} a[j] = \sum_{j=i+1}^{dim-1} a[j]$$

### ESERCIZIO 4 (10 punti)

Si vuole modellare una lista di caratteri alfabetici.

- Definire i tipi opportuni per rappresentare liste di questo tipo.
- Scrivere in C una procedura **iterativa** che, dati in ingresso un carattere minuscolo `x` e una lista di caratteri minuscoli, sposta in *ultima* posizione il primo elemento della lista, se esiste, che nell'ordine alfabetico viene dopo `x`.
- Supponendo che la lista abbia la proprietà che tutte le lettere minuscole precedono quelle maiuscole, e che entrambe le categorie di lettere siano ordinate alfabeticamente, scrivere una procedura che, dati in ingresso un carattere `x` e una lista di caratteri, inserisca il carattere `x` nella lista preservando la proprietà di ordine dato.