

# Fondamenti di Programmazione - CdL in MATEMATICA

## I Prova di verifica del 22/4/2013

num. eserc.	1	2	3	4	5
punt. tot	6	6	2	7	9

**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 e provochino l'uscita forzata.

### ESERCIZIO 1 (6 punti)

Sia  $L$  il linguaggio delle stringhe  $w$  di  $\Sigma = \{a, b\}$ , tali che ogni sottostringa  $x$  di  $w$  che inizia e termina con  $b$  contiene un numero pari di  $a$ . Fornire l'automa a stati finiti deterministico che riconosce  $L$ .

### ESERCIZIO 2 (6 punti)

Si consideri l'automa a stati finiti non deterministico  $N$  che ha la seguente tabella di transizione.

		$a$	$b$
$I$	$q_0$	$\{q_0, q_1\}$	$\{q_0, q_2\}$
$F$	$q_1$	$\{q_1\}$	$\emptyset$
$F$	$q_2$	$\{q_2\}$	$\emptyset$

- Spiegare perché si tratta di un NFA.
- Far vedere che la stringa "abaaa" è accettata da  $N$ .
- Costruire l'automa a stati finiti deterministico equivalente  $D$ .

### ESERCIZIO 3 (2 punti)

Completare la specifica dello stato finale, relativa al seguente problema. Data una sequenza di interi di dimensione  $dim$ , calcolare il massimo e il minimo e dire se coincidono.

Stato iniziale:  $\{dim \rightsquigarrow K, c[0] \rightsquigarrow V_0, \dots\}$  con  $K > 0$   
Stato finale:  $\{ \dots$

### ESERCIZIO 4 (4+3 punti)

Scrivere in C una funzione `maxvolte`, che dato un array `v` di caratteri e la sua dimensione `dim`, restituisca la lettera maiuscola che più frequentemente degli altri è seguita dalla lettera successiva nell'ordine alfabetico. Ad esempio, se l'array contiene i caratteri B T M N M P S T M N la funzione restituirà il carattere M, che per due volte è seguito dal carattere N. Si premieranno (3 punti oltre ai 4) le soluzioni che prevedono un solo scorrimento dell'array `v`.

### ESERCIZIO 5 (9 punti)

- Scrivere una funzione C che, dato un array `SEQ` di caratteri di lunghezza `DIM` che rappresenta una sequenza di DNA e un array `seq` di caratteri di lunghezza `dim` che rappresenta una sequenza più corta (`dim < DIM`) di DNA, controlla se la sequenza contenuta in `seq` appare come segmento (sottosequenza di caratteri adiacenti) in `SEQ`. La funzione restituisce l'indice a partire dal quale si trova il segmento, se esiste, restituisce -1 altrimenti. Ad esempio se le sequenze fossero le seguenti, allora la funzione dovrebbe restituire l'indice 4.

A	T	C	C	A	T	G	G	C	T	A
---	---	---	---	---	---	---	---	---	---	---

A	T	G	G	C	T
---	---	---	---	---	---

- Scrivere una funzione simile che, con gli stessi ingressi, controlla quante volte il segmento appare nella sequenza. La funzione restituisce proprio il numero di volte, ad esempio se le sequenze fossero le seguenti, allora la funzione dovrebbe restituire 2.

A	T	C	C	A	T	G	G	C	T	A	A	T	G	G	C	T
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

A	T	G	G	C	T
---	---	---	---	---	---

Se le sequenze fossero invece le seguenti, allora la funzione dovrebbe restituire 4.

A	A	A	A	A
---	---	---	---	---

A	A
---	---