Prova scritta dell'esame di Fondamenti di Informatica

Data: 12 aprile 2003 **Durata della prova: 2 ore**

Cognome e Nome		*Corso*		*Matricola*	

Traccia E

Esercizio 1

Si consideri il seguente codice:

```
public static int metodoE (int n){
     int s = 0;
     for (int i=1; i<=n; i++){
       boolean p = true;
       for (int j=2; j<i; j++)
         if(i%j == 0)
           p = false;
        if(p){
         System.out.println(i);
         s += 1;
        }
     }
     return s;
}
```

Si descriva sinteticamente la funzione svolta dal metodo e, in particolare, se ne mostri l'esecuzione nel caso in cui il valore del parametro sia **n**=10. Specificare inoltre il contenuto dell'intero **s** restituito.

Esercizio 2

Scrivere un metodo *compvett* che riceve tre array **v1** e **v2** e **v3** di interi e ritorna un array **v4** della stessa lunghezza di **v3** in cui l'elemento di posizione **i** è calcolato come segue:

- se **v3[i]** è maggiore o uguale a 0, **v4[i]** sarà uguale al prodotto degli ultimi **v1[i]** elementi di **v2;** cioè il valore contenuto in **v1[i]** indica quanti sono gli elementi del vettore **v2** da moltiplicare, partendo dall'ultimo; il risultato va assegnato a **v4[i].**
- se v3[i] è minore di 0, **v4[i]** sarà uguale a **v3[i]**.

Ad esempio, se **v1** = [3, 5, 4, 10], **v2** = [2, 4, 1, 3] e **v3** = [3, -1, 0, -4], allora l'array **v4** restituito sarà [12, -1, 24, -4].

Esercizio 3

Si progetti una classe GestioneMatrici con i seguenti metodi:

1) un metodo *sommaVicini* che, ricevuti in ingresso una matrice quadrata di interi *M* e una coppia di indici *i*, *j*, restituisce la somma degli elementi di *M* che si trovano intorno all'elemento in posizione *i*, *j*. Ad esempio, se il metodo viene invocato sulla matrice M sotto riportata con parametri i=1, j=2, esso restituisce la quantità 8, pari alla somma dei valori colorati in grigio (2-1+5+2-3+5-5+3). Per gli elementi che sono sul bordo, i vicini sono ovviamente quelli possibili all'interno della matrice. Ad esempio se il metodo viene invocato con i parametri i=4, j=2, esso restituirà il valore 6 (= 6 -7+3+1+3).

M=	*j* 0	1	2	3	4
	7	2	-1	5	4
i	1	3	9	2	9
	5	-5	5	-3	1
	2	-7	3	1	0
	2	6	5	3	4

M=	*j* 0	1	2	3	4
	7	2	-1	5	4
	1	3	9	2	9
	5	-5	5	-3	1
	2	-7	3	1	0
i	2	6	5	3	4

2) un metodo *nuovaMatrice* che riceve in ingresso una matrice *M* di interi e restituisce una matrice N in cui gli elementi sono così ottenuti:
   - se l'elemento M[i][j] è maggiore o uguale al valore della somma dei suoi vicini, allora N [i][j] è pari a M[i][j];
   - altrimenti, N [i][j] è pari proprio alla somma dei vicini di M[i][j].

3) un metodo *main* in cui, una volta letta una matrice quadrata di interi *M*, e ottenuta tramite il metodo nuovaMatrice una nuova matrice *N*, viene ricavata e visualizzata la matrice *D* pari a *N* - *M*.