

Prova scritta dell'esame di Fondamenti di Informatica

Data: 6 aprile 2004

Durata della prova: 2 ore

Cognome e Nome		Corso		Matricola	
----------------	--	-------	--	-----------	--

TRACCIA A

Esercizio 1

Si consideri il seguente programma:

```
public class Esercizio1A {
    public static int metodoA(int x[]) {
        int c = 0;
        for (int i = 0; i < x.length/2; i++) {
            int j = i + x.length/2;
            if (( x[i] % x[j] == 0 ) || ( x[j] % x[i] == 0 )) {
                c++;
                System.out.println (x[i]);
            }
        }
        return c;
    }
    public static void main(String args[]) {
        int v[] = {3, 4, 8, 5, 2, 16};
        int k = metodoA(v);
        System.out.println ("k = "+k);
    }
}
```

Si descriva sinteticamente la funzione svolta dal metodo **metodoA** e, in particolare, si mostri l'esecuzione e cosa viene stampato nel caso in esempio, in cui $v = \{3, 4, 8, 5, 2, 16\}$.

Esercizio 2

Si scriva un metodo *costruisciVettore* che riceve in ingresso due vettori di interi **v1** e **v2**, e restituisce un vettore **v3**. In particolare, il vettore **v3** dovrà contenere gli elementi di **v1** il cui valore è maggiore del massimo tra gli elementi di **v2**. Qualora nessuno tra gli elementi di **v1** sia maggiore del massimo degli elementi di **v2**, il vettore **v3** conterrà un solo elemento, il cui valore è il massimo degli elementi di **v2**.

Ad esempio, se $v1 = [20, 9, 4, 8, 2]$ e $v2 = [2, 3, 7, 4]$ il vettore restituito è $v3 = [20, 9, 8]$, in quanto i valori 20, 9 ed 8 di **v1** sono maggiori di 7, che è il massimo tra gli elementi di **v2**.

Esercizio 3

Si realizzi una classe *Esercizio3A* che contenga almeno i seguenti metodi:

1. Un metodo *elaboraArray* che riceve una matrice quadrata di interi **M** di dimensione dispari, e restituisce un vettore **V** contenente gli elementi situati nel triangolo avente come base la prima riga e come vertice opposto l'elemento centrale di **M** (si veda l'esempio sottostante).
2. Un metodo *confronta* che riceve una matrice di interi **M** ed un intero **r**, e restituisce un booleano. In particolare, il metodo restituisce **true** se la somma degli elementi di **M** che si trovano al di sopra della riga **r** è uguale alla somma degli elementi che si trovano al di sotto della riga **r**, altrimenti restituisce **false**. Il metodo restituisce **false** anche se il numero di righe di **M** è minore di **3**, oppure se $r < 1$ o $r > M.length-2$.
3. Un metodo *elaboraMatrice* che riceve una matrice di interi **M** e restituisce una matrice **A** ottenuta da **M** eliminando la prima e l'ultima colonna.
4. Un metodo *main* nel quale si legge una matrice quadrata di interi, e si invocano opportunamente i metodi definiti ai punti 1, 2 e 3.

Esempio:

M =

3	19	13	22	7
15	0	24	2	16
4	21	9	14	23
17	1	25	8	11
6	20	12	18	5

1. *elaboraArray* (**M**) restituisce $V = [3, 19, 13, 22, 7, 0, 24, 2, 9]$.
2. *confronta* (**M**, 2) restituisce **false** perché la somma degli elementi nelle righe 0 e 1 (pari a 121) è diversa dalla somma degli elementi nelle righe 3 e 4 (pari a 123).

3. *elaboraMatrice* (**M**) restituisce **A** =

19	13	22
0	24	2
21	9	14
1	25	8
20	12	18