

Prova scritta dell'esame di Fondamenti di Informatica

Data: 25 marzo 2006

TRACCIA B

Esercizio 1

Il metodo genera un nuovo vettore y della stessa lunghezza del vettore x su cui opera una operazione di reverse degli elementi appartenenti alla prima e alla seconda meta' del vettore.

Per $v = \{1, 5, 9, 5, 2, 3, 7\}$

il metodo eseguirà la seguente traccia di iterazioni:

$i=0$ $k[3]=v[6]=7$
 $k[2]=v[0]=1$

$i=1$ $k[4]=v[5]=5$
 $k[1]=v[1]=5$

$i=2$ $k[5]=v[4]=2$
 $k[0]=v[2]=9$

$i=3$ esce dal for

Poiché $x.length \% 2 \neq 0$ restituisce true che indica che la lunghezza del vettore x è dispari,

$k[6]=v[3]=5$

il vettore k risultato è $\{9, 5, 1, 7, 3, 2, 5\}$

Esercizio 2

public class Esercizio2TB{

```
    public static int[] generaVettore(int[] v1){
        int[] v2 = new int[v1.length];
        for(int i=0; i<v1.length; i++){
            double somma = 0;
            for(int j=0; j<=i; j++){
                somma += v1[j];
            }
            double media = somma / (i+1);
            if(media <= v1[i])
                v2[i] = (int)media;
            else{
                int s = 0;
                for(int j=0; j<v1.length; j++){
                    if(j<i)
                        s -= v1[j];
                    if(j>i)
                        s += v1[j];
                }
                v2[i] = s;
            }
        }
        return v2;
    }
```

```
    public static void main(String[] args){
        int[] v1 = {20, 9, 4, 8, 2};
        int[] v2 = generaVettore(v1);
        for(int i=0; i<v2.length; i++)
            System.out.print(v2[i] + " ");
        System.out.println();
    }
```

}

Esercizio 3

```
import corejava.*;

public class Esercizio3TB{

    //METODO 3.1
    public static int[] elaboraArray(int[][] M){
        int n = M.length;
        int[] v = new int[4*(n/2) - 2];
        int shift = n/2 - 1;
        int pos = 0;
        for(int i=n/2; i<n; i++){
            v[pos] = M[0][i];
            v[pos + shift] = M[i - n/2][n-1];
            v[pos + 2*shift + 1] = M[i][0];
            v[pos + 3*shift + 1] = M[n-1][i - n/2];
            pos++;
        }

        return v;
    }

    //METODO 3.2
    public static boolean verifica(int[][] M, int p){
        int cnt = 0;
        for(int i=0; i<M.length; i++)
            for(int j=0; j<M[0].length; j++)
                if(M[i][j] % p == 0)
                    cnt++;

        if(cnt >= p)
            return true;
        else
            return false;
    }

    //METODO 3.3
    public static int[][] elaboraMatrice(int[][] M){
        int n = M.length;
        int[][] A = new int[n][n-2];
        int r = 0;
        int c;
        for(int i=0; i<n; i++){
            c = 0;
            for(int j=0; j<n; j++)
                if(j != i && j != (n-1-i) ){
                    A[r][c] = M[i][j];
                    c++;
                }
            r++;
        }
        return A;
    }

    //METODO 3.4
```

```

public static void main(String[] args){

    int[][] M = leggiMatrice();
    int[] v = elaboraArray(M);
    stampaMatrice(M);
    System.out.println("Il vettore restituito dal primo metodo e' ");
    stampaVettore(v);

    int p = Console.readInt("Dammi il numero di cui trovare i multipli: ");
    if(verifica(M,p))
        System.out.println("La matrice contiene almeno "+p+" multipli di "+p+" ");
    else
        System.out.println("La matrice non contiene almeno "+p+" multipli di "+p+" ");

    int[][] A = elaboraMatrice(M);
    System.out.println("La matrice restituita dal terzo metodo e' ");
    stampaMatrice(A);

}

//legge una matrice quadrata di interi
public static int[][] leggiMatrice(){
    int n = Console.readInt("Dammi la dimensione (pari) della matrice: ");
    int[][] M = new int[n][n];
    for(int i=0; i<n; i++)
        for(int j=0; j<n; j++)
            M[i][j] = Console.readInt("M["+i+""]["+j+"] = ");
    return M;
}

//stampa una matrice di interi
public static void stampaMatrice(int[][] M){
    int n = M.length;
    int m = M[0].length;
    for(int i=0; i<n; i++){
        for(int j=0; j<m; j++)
            System.out.print(M[i][j] + " ");
        System.out.println();
    }
}

//stampa un vettore di interi
public static void stampaVettore(int[] v){
    int n = v.length;
    for(int i=0; i<n; i++)
        System.out.print(v[i] + " ");
    System.out.println();
}
}

```