

Esercizio 2

In una Classe EserMatrici2 scrivere un metodo costruisciVettore che riceve in ingresso una Matrice quadrata d'interi e un intero k e restituisce un vettore d'interi v.

Ogni elemento di v deve soddisfare le seguenti condizioni:

- 1) Deve essere dato dalla somma degli elementi della diagonale secondaria e dalla somma degli elementi di una riga.
- 2) La seconda somma deve essere maggiore di k, altrimenti non può essere inserita nel vettore.

Scrivere un metodo main che esegue costruisciVettore e visualizza il risultato.

Esempio:

```
matrice = [7 , 4 , 2 , 1 , 1]
           [1 , 0 , 1 , 5 , 2]
           [7 , 0 , 3 , 4 , 5]
           [1 , 2 , 13 , 0 , 1]
           [6 , 2 , 3 , 4 , 5]
```

Ris = [36, 34, 37]

Possibile soluzione:

```
import java.util.Arrays;

public class EserMatrici2 {

    public static int[] costruisciVettore(int[][]matrice, int k){
        int somme = 0;
        int sommaDiagonale = 0;
        for(int i = 0;i<matrice.length;i++){
            sommaDiagonale+=matrice[matrice.length-1-i][i];
        }

        int[]temp = new int[matrice.length*2];
        for(int i = 0; i<matrice.length;i++){
            int tempSomma = 0;
            for(int j = 0; j<matrice.length;j++){
                tempSomma = tempSomma + matrice[i][j];
            }
            if (tempSomma > k){
                temp[somme] = tempSomma+sommaDiagonale;
                somme++;
            }
        }
        int[]v = Arrays.copyOf(temp, somme);
        return v;
    }

    public static void main(String[]args){
        int [][] matrice = {{7 , 4 , 2 , 1 , 1},{1 , 0 , 1 , 5 , 2},
            {7 , 0 , 3 , 4 , 5},{1 , 2 , 13 , 0 , 1},{6 , 2 , 3 , 4 , 5}};
        int[] ris = costruisciVettore(matrice,15);
        System.out.print("[ ");
        for(int i = 0;i< ris.length;i++){
            System.out.print(ris[i]+" ");
        }
        System.out.println("]");
    }
}
```

Esercizi svolti sulle Matrici

Ragionamento:

Mi creo una variabile `somme` che fungerà da contatore. `Somme`, infatti, conta il numero di righe per cui vale la seconda condizione.

Fatto ciò, mi calcolo la somma dei valori presenti sulla diagonale secondaria, impostando la riga della matrice a `matrice.length - 1 - i` e la colonna ad `i`.

Mi creo un vettore temporaneo che ha come lunghezza 2 volte il numero di righe della matrice (anche se impostando come lunghezza il numero di righe dovrebbe funzionare lo stesso).

Dopo di che, inizio a scandire il vettore e per ogni riga calcolo la somma dei suoi elementi. Se tale somma soddisfa la seconda condizione, aggiungo a tale valore la somma della diagonale secondaria, inserisco il risultato nel vettore temporaneo e incremento il contatore “`somme`”.

Infine mi creo il vettore finale che ha come lunghezza il numero di righe che soddisfa la seconda condizione.

Per praticità utilizzo il metodo `copyOf` di `Arrays`, il quale permette di creare un nuovo vettore a partire da quello originale specificando la nuova dimensione (in questo caso la nuova dimensione è “`somme`”).

Infine restituito il nuovo vettore.

Il metodo `main`, oltre a creare la matrice e a ottenere il vettore risultato, stampa il vettore risultato a video.