

## Prova scritta dell'esame di Fondamenti di Informatica

Data: 6 aprile 2004

Durata della prova: 2 ore

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

### TRACCIA B

#### Esercizio 1

Si consideri il seguente codice:

```
public static int metodo1( boolean vett[] ){
    if ((vett.length%2 !=0) || (vett.length < 2))
        return -1;
    else{
        int k=0, p=0;
        while( k < vett.length/2 ){
            if((vet[k] == vet[vett.length-k-1])&&(k%2==0))
                p++;
            k++;
        }
        return p;
    }
}
```

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 **vett**=[false, false, true, false, false, true, false, false]. Specificare il contenuto dell'intero **p** restituito.

#### Esercizio 2

Si scriva un metodo *generaVettore* che riceve in ingresso due vettori di interi **v1** e **v2** di uguale dimensione, e restituisce un vettore **v3** contenente un sottoinsieme degli elementi di **v1**. In particolare, il generico elemento **v1[i]** sarà inserito in **v3** se **v1[i]** è maggiore di tutti gli elementi in **v2** di indice maggiore di **i**. Qualora nessuno tra gli elementi di **v1** rispetti la condizione suddetta, il vettore **v3** conterrà un solo elemento il cui valore è **-1**.

Ad esempio, se **v1**= [ 7 , 10 , 9 , 2 ] e **v2**= [ 2, 8, 1, 4 ] il vettore restituito è **v3** = [ 10, 9 ], in quanto **10** è maggiore di **1** e **4**, e **9** è maggiore di **4**.

#### Esercizio 3

Si realizzi una classe *Esercizio3B* che contenga almeno i seguenti metodi:

1. un metodo *estraiArray* che riceve in ingresso una matrice quadrata **A** e restituisce un vettore contenente gli elementi che si incontrano percorrendo la diagonale secondaria di **A** dall'alto verso il basso, fino al primo **0** (si veda l'esempio);
2. un metodo *estraiMatrice* che riceve in ingresso una matrice quadrata **A** ed un intero **k** e restituisce la matrice ottenuta eliminando dalla matrice **A** le prime **k** righe e le prime **k** colonne;
3. un metodo *controlla* che riceve in ingresso una matrice quadrata **A** e restituisce **true** se **A** non contiene elementi di valore pari posti su righe di indice dispari e **false** altrimenti;
4. un metodo *main* in cui viene letta da input una matrice quadrata **A** di interi, e vengono opportunamente invocati i metodi definiti ai punti 1, 2 e 3.

Esempio:

$$\mathbf{A} = \begin{bmatrix} 5 & 2 & 11 & 9 & 12 \\ 9 & 15 & 10 & 7 & 1 \\ 14 & 21 & 19 & 2 & 2 \\ 8 & 0 & 13 & 9 & 7 \\ 10 & 17 & 12 & 10 & 3 \end{bmatrix}$$

$$1. \text{estraiArray}(A) \longrightarrow [ 12, 7, 19, 0 ]$$

$$2. \text{estraiMatrice}(A,2) \longrightarrow \begin{bmatrix} 19 & 2 & 2 \\ 13 & 9 & 7 \\ 12 & 10 & 3 \end{bmatrix}$$

$$3. \text{controlla}(A) \longrightarrow \text{false} \text{ (10 sulla riga 1)}$$