

Scrivere una classe Docente che rappresenti le seguenti informazioni relative ad un docente: nome, cognome, codice ed età, e che contenga il costruttore parametrizzato ed i metodi getCodice, getCognome e getEta che restituiscono rispettivamente il codice, il cognome e l'età del docente.

Scrivere poi una classe Università, che rappresenti un insieme di docenti universitari tramite un array di tipo Docente, e che contenga il costruttore parametrizzato ed un metodo etaMinima che restituisce la minima età tra i docenti universitari.

Soluzione:

```
class Docente{
    private String nome;
    private String cognome;
    private int codice;
    private int eta;
    public Docente(String n, String c, int cod, int e){
        nome=n;cognome=c;codice=cod;eta=e;
    }
    public Docente(Docente d){
        nome=d.getNome();
        cognome=d.getCognome();
        codice=d.getCodice();
        eta=d.getEta();
    } //costruttore di copia non richiesto ma implementato perchè usato nella //classe Università
    public String getNome(){
        return nome;
    } //non richiesto ma implementato perchè usato nel costruttore di copia
    public String getCognome(){
        return cognome;
    }
    public int getCodice(){
        return codice;
    }
    public int getEta(){
        return eta;
    }
}

class Università{
    private Docente[] docenti;
    public Università(Docente[] d){
        docenti=new Docente[d.length];
        for(int i=0;i<d.length;i++)
            docenti[i]=new Docente(d[i]);
    }
    public int etaMinima(){
        int min=docenti[0].getEta();
        for(int i=1;i<=docenti.length;i++)
            if(docenti[i].getEta()<min) min=docenti[i].getEta();
        return min;
    }
}
```

Esercizio 2 (esercizio 9.2 del libro di testo). Implementare un metodo statico di nome `filtra` che prende come parametro un array `a` di numeri interi e un intero positivo `k`, e che restituisce un nuovo array `b` di numeri interi contenente i soli elementi di `a` che sono divisibili per `k`. Si faccia in modo che l'array `b` abbia una dimensione uguale al numero di elementi che deve contenere (non devono cioè esservi posizioni inutilizzate).

Esercizio 2 – svolgimento.

```
public static int[] filtra(int[] a, int k){
    // conta gli elementi da inserire nell'array restituito
    // e ne memorizza il numero in numElementi
    int numElementi = 0;
    for (int i=0; i<a.length; i++)
        if (a[i]%k == 0)
            numElementi++;

    // crea l'array b da restituire e vi memorizza gli elemen
    // dell'array a che sono divisibili per k
    int[] b = new int[numElementi];
    int j = 0; // la prossima posizione libera nell'array b
    for (int i=0; i<a.length; i++)
        if (a[i]%k == 0){
            b[j] = a[i];
            j++;
        }
    return b;
}
```

Scrivere una classe Libro che rappresenti le seguenti informazioni relative ad un libro: titolo, autore, prezzo, e che contenga il costruttore parametrizzato ed i metodi getTitle, getAutore e getPrezzo che restituiscono rispettivamente il titolo, l'autore e il prezzo del libro.

Scrivere poi una classe Libreria, che rappresenti un insieme di libri tramite un array di tipo Libro, e che contenga il costruttore parametrizzato ed un metodo trova che accetta in ingresso un autore a e intero k e restituisce il numero di libri contenuti nella libreria aventi autore a e prezzo superiore a k.

Soluzione:

```
class Libro{
    private String titolo;
    private String autore;
    private int prezzo;
    public Libro(String t, String a, int p){
        titolo=t;autore=a;prezzo=p;

    }
    public Libro(Libro l){
        titolo=l.getTitle();
        autore=l.getAutore();
        prezzo=l.getPrezzo();
    } //costruttore di copia non richiesto ma implementato perchè usato nella classe Libreria
    public String getTitle(){
        return titolo;
    } //non richiesto ma implementato perchè usato nel costruttore di copia
    public String getAutore(){
        return autore;
    }
    public int getPrezzo(){
        return prezzo;
    }
}

class Libreria{
    private Libro[] libri;
    public Libreria(Libro[] l){
        libri=new Libro[l.length];
        for(int i=0;i<l.length;i++)
            libri[i]=new Libro(l[i]);
    }
    public int Trova(String a, int k){
        int cont=0;
        for(int i=0;i<=libri.length;i++)
            if(libri[i].getAutore().equals(a)&&libri[i].getPrezzo()>k) cont++;
        return cont;
    }
}
```