

Esercitazione 1



Planning & Scheduling Team
ISTC-CNR

Esercitazione

- Progettare e realizzare una classe “Box” che rappresenti una scatola specificandone la larghezza, l’altezza e la profondità.
- Aggiungere un metodo per il calcolo del volume della scatola
- Scrivere un metodo per ordinare un vettore di classi “Box” basato sull’algoritmo di selection sort



Planning & Scheduling Team
ISTC-CNR

La classe Box

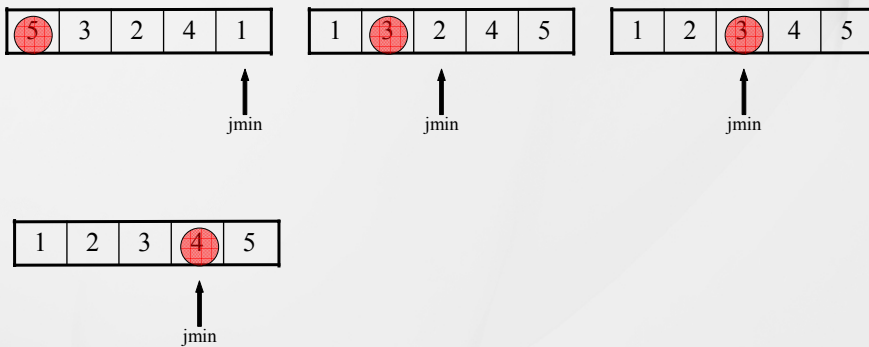
```
class Box
{
    double width;
    double height;
    double depth;

    Box(double w, double h, double d) //costruttore con parametri
    {
        width=w;
        height=h;
        depth=d;
    }

    double volume()
    {
        return width*height*depth;
    }
}
```

Ordinamento di un vettore

Selection Sort



Ordinamento di un vettore

Selection Sort

```
public static void selectionSort(int[] a)
{
    int n = a.length;

    for (int i=0; i<n-1; i++)
    {
        // trova il piu' piccolo elemento da i a n-1
        int jmin = i;
        for (int j=i+1; j<n; j++)
        {
            if (a[j]<a[jmin])
                jmin = j;
        }

        // scambia gli elementi i e jmin
        int aux = a[jmin];
        a[jmin] = a[i];
        a[i] = aux;
    }
}
```

Complessità: doppio ciclo $\rightarrow O(n^2)$



Planning & Scheduling Team
ISTC-CNR

Soluzione

```
import java.util.Vector;
import java.util.Enumeration;

class myVector{

public static void main(String args[])
{
    scatola box1=new box(5,6,2);
    scatola box2=new box(10,3,4);
    Vector v=new Vector();
    v.addElement(box1);
    v.addElement(box2);

    System.out.println(selectionSort(v));

}

public static Vector selectionSort(Vector v){
    for(int i=0; i<v.size()-1; i++)
    {
        int jmin=i;
        for(int j=i+1; j<n; j++)
        {
            if(((Box)v.elementAt(j)).volume()<((Box)v.elementAt(jmin)).volume())
                jmin=j;
        }

        Box temp=((Box)v.elementAt(jmin));
        v.setElementAt(v.elementAt(i),jmin);
        v.setElementAt(temp,i);
    }

    Vector v2=new Vector();
    for(int i=0; i<v.size(); i++) v2.addElement(v.elementAt(i));
    return v2;
}
}
```



Planning & Scheduling Team
ISTC-CNR