

Ordenación en Java

Ordenación en Java según diversos criterios

Situación inicial

- Conocemos al menos un algoritmo para ordenar arrays de enteros (Selección en este caso)
- Tenemos una clase con varios atributos sobre los que podemos establecer una ordenación (Clase Persona, ordenar por edad o por altura)
- Queremos ordenar un array de la clase anterior según un criterio u otro

Algoritmo de Selección

- Su funcionamiento es el siguiente:
 - Buscar el mínimo elemento de la lista
 - Intercambiarlo con el primero
 - Buscar el siguiente mínimo en el resto de la lista
 - Intercambiarlo con el segundo
- Y en general:
 - Buscar el mínimo elemento entre una posición i y el final de la lista
 - Intercambiar el mínimo con el elemento de la posición i

Ordenamiento por selección. (2015, 8 de diciembre). Wikipedia, La enciclopedia libre. Fecha de consulta: 09:28, diciembre 14, 2016 desde https://es.wikipedia.org/w/index.php?title=Ordenamiento_por_selecci%C3%B3n&oldid=87656620.

Implementación en Java

- ```
public static void ordenarSeleccion(int[] array){
 for(int i=0;i<array.length-1;i++){
 //array.length-1 porque al llegar al último, ya estará ordenado
 int minimo=array[i];
 int posicionMinimo=i;
 for(int j=i+1;j<array.length;j++){
 if(array[j]<minimo){
 minimo=array[j];
 posicionMinimo=j;
 }
 }
 if(posicionMinimo!=i){ //Si el mínimo no estaba en i, intercambiar
 int temp=array[posicionMinimo];
 array[posicionMinimo]=array[i];
 array[i]=temp;
 }
 }
}
```

# Clase Persona

```
public class Persona {
 private String nombre;
 private int edad;
 private int altura;

 Persona(String nombre,int edad,int altura){
 //Constructor típico
 }
 //Getters para los tres atributos.

 public String datosPersona(){
 return String.format("%5s %3daños %3dcm",
 this.nombre,this.edad,this.altura);
 }
}
```

# Programa Principal

```
public class OrdenarPersonas {
 public static void main(String[] args) {
 Persona[] listaPersonas={
 new Persona("Ana",30,150),
 new Persona("Bea",20,160),
 new Persona("Clara",10,170),
 new Persona("Delia",60,190),
 new Persona("Elena",50,180)
 };
 for(Persona p:listaPersonas){ System.out.println(p.datosPersona()); }
 ordenarPorEdad(listaPersonas);
 for(Persona p:listaPersonas){ System.out.println(p.datosPersona()); }
 ordenarPorAltura(listaPersonas);
 for(Persona p:listaPersonas){ System.out.println(p.datosPersona()); }
 }
}
```

# Modificaciones en la Ordenación

- ```
public static void ordenarSeleccion(int[] Persona[] array){  
    for(int i=0;i<array.length-1;i++){  
        //array.length-1 porque al llegar al último, ya estará ordenado  
        int tipoAComparar minimo=array[i] array[i].getter;  
        int posicionMinimo=i;  
        for(int j=i+1;j<array.length;j++){  
            if(array[j] array[j].getter<minimo){  
                minimo=array[j] array[j].getter;  
                posicionMinimo=j;  
            }  
        }  
        if(posicionMinimo!=i){ //Si el mínimo no estaba en i, intercambiar  
            int Persona temp=array[posicionMinimo];  
            array[posicionMinimo]=array[i];  
            array[i]=temp;  
        }  
    }  
}
```

ordenarPorEdad

```
private static void ordenarPorEdad(Persona[] array) {  
    for (int i = 0; i < array.length - 1; i++) {  
        int minimo = array[i].getEdad();  
        int posicionMinimo = i;  
        for (int j = i + 1; j < array.length; j++) {  
            if (array[j].getEdad() < minimo) {  
                minimo = array[j].getEdad();  
                posicionMinimo = j;  
            }  
        }  
        if (posicionMinimo != i) {  
            Persona temp = array[posicionMinimo];  
            array[posicionMinimo] = array[i];  
            array[i] = temp;  
        }  
    }  
}
```