Informática II – Prepa Tec Campus Eugenio Garza Lagüera  
Actividad 3: Algoritmos de Ordenamiento

**Sección 1:** Ordena los siguientes arreglos utilizando el algoritmo indicado. Muestra cada uno de los pasos.

**Selection Sort**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1 | 7 | 4 | 3 | 2 | 1 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 3 | 5 | 8 | 3 | -1 | -9 | 1 | 2 |

**Bubble Sort**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 9 | 6 | 1 | 10 | -1 | 3 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| -1 | -3 | 2 | 7 | 8 | 9 | 0 | 10 |

**Sección 2:**

**Problema 1:** Crea un método estático con la siguiente firma:

* public static void bubbleSort(String[] array).

El método deberá acomodar los elementos alfabéticamente, de menor a mayor; utilizando el algoritmo Bubble Sort. Para comparar dos Strings, puedes usar el método compareTo() de la clase String, que utiliza una comparación lexiconográfica.

**Ejemplo:**

* bubbleSort({“a”,”b”,”1”,”5”,”z”,”x”}) 🡪 Resultado: {“1”,”5”,”a”,”b”,”x”,”z”}
* bubbleSort({“Peter”,”Jack”,”John”}) 🡪 Resultado: {“Jack”, “John”, “Peter”}

**Problema 2:** Crea un método estático con la siguiente firma

* public static void selectionSort(int[] array).

El método deberá acomodar los elementos descendentemente, es decir, de mayor a menor; utilizando el algoritmo Selection Sort.

**Ejemplo:**

selectionSort({1,5,4,3,6,7,8}) 🡪 Resultado: {8,7,6,5,4,3,1}