Escuela Superior de Ingeniería y Tecnología
Universidad de La Laguna

Grado en Ingeniería Informática
Algoritmos y Estructuras de Datos Avanzadas

Curso 2022-2023

Práctica 5: Implementación de métodos de ordenación

---

1. Objetivo

El objetivo de esta práctica es trabajar con los algoritmos de ordenación interna vistos en las clases de teoría [1][2] y realizar una implementación en lenguaje C++ de las técnicas de los mismos. Se utilizarán plantillas de función.

2. Entrega

Se realizará en dos sesiones de laboratorio en las siguientes fechas:

Sesión tutorada: del 10 al 14 de abril de 2023
Sesión de entrega: del 17 al 21 de abril de 2023

3. Enunciado

Se pide implementar algunos de los algoritmos de ordenación vistos en clase [1][2]. Los algoritmos a implementar serán, al menos, los siguientes:

- Inserción
- MergeSort
- Por Incrementos Decrecientes (`ShellSort`): debe permitir seleccionar la constante de reducción `alfa`, siendo `0 < alfa < 1`
- HeapSort
- RadixSort

4. Notas de implementación

1. La implementación de cada método de ordenación se realiza mediante una plantilla de función, en la que se especifica el tipo de elementos a ordenar (`Key`). La función recibe como parámetros la secuencia a ordenar de elementos de tipo `Key` y su tamaño.

```
nombre_método<Key> (vector<Key>, size)
```

2. Se implementa también una clase base abstracta `SortMethod<key>` que defina un método nulo `Sort( )`:

```
template<class Key>
class SortMethod {
      protected:
            unsigned size;
            vector<Key> seq;
      public:
            void Sort( ) = 0;

};
```

3. Para implementar los diferentes métodos de ordenación se define una familia de clases genéricas a partir de la clase base `SortMethod<Key>`. De esta forma, la implementación de cada método se realiza en la sobrecarga del método `Sort`, empleando una llamada a la plantilla de función adecuada.

4. El programa principal tiene el siguiente comportamiento:

    a. Se utiliza `Key = long` para realizar la ejecución de los métodos.

    b. Se pide al usuario el algoritmo a ejecutar y el tamaño de la secuencia (N).

    c. Se pide al usuario que seleccione entre introducir los N valores manualmente o generarlos de forma aleatoria.

    d. En el caso de elegir valores aleatorios, los números generados estarán entre el 1000 y el 9999.

    e. Se crea el método de ordenación especificado por el usuario.

    f. Se realiza la ejecución del método seleccionado mostrando por pantalla la traza cuando se haya seleccionado entrada de valores manual, en cuyo caso se debe mostrar al menos cómo queda la secuencia después de cada iteración del algoritmo.

5. Durante las sesiones de laboratorio se pueden solicitar cambios y/o ampliaciones en la especificación de este enunciado.

5. Referencias

[1] Google: Apuntes de clase.
[2] Wikipedia: Algoritmo de ordenamiento:
https://es.wikipedia.org/wiki/Algoritmo_de_ordenamiento