



TEMA 5: ALGORITMOS DE ORDENAMIENTO Y BUSQUEDA

Fecha de entrega: 20/06/2024

- ❖ Arreglos de registros: Definición, declaración y uso.

Para comenzar es importante recordar lo siguiente, un arreglo es una estructura de datos lineal para almacenar datos en forma estática, por ejemplo:

0	1	2	3	4
4,3	2,8	4,9	1,8	3,5

Esta es la representación de un arreglo de números flotantes, almacenados en forma secuencial por ello su acceso es secuencial a través de los índices.

Ahora bien, así como almacenamos datos int, float, double u otros también podemos almacenar en estos arreglos datos tipo struct, por ejemplo:

```
typedef struct Persona{
```

```
    char nombre[25];
```

```
    char apellido[25];
```

```
    int edad;
```

```
}personas;
```

```
personas arreglo1[4];
```

//VARIABLE LLAMADA ARREGLO 1 DE DIMENSION 4, DE TIPO personas

0	1	2	3	4
Nombre	Nombre	Nombre	Nombre	Nombre
Apellido	Apellido	Apellido	Apellido	Apellido
Edad	Edad	Edad	Edad	Edad

Lo que se traduce en que cada posición del arreglo se almacena un dato tipo personas o struct Persona, esto lo conocemos como *Arreglo de Registros*

Para acceder a cada posición en el arreglo de registros, lo hacemos con un for en el índice i, y mediante el punto (.) se accede a cada uno de los campos

```
arreglo1[i].nombre = ´ Dubraska ´;
```

```
arreglo1[i].apellido = ´ Roca ´;
```

```
arreglo1[i].edad = 23;
```

Ya contextualizados, es recomendable practicar el manejo de este tipo de arreglos, si realizaste los ejercicios de struct en la guía, te familiarizas con su manejo y fácilmente puedes abordar el tema de Ordenamiento y Búsqueda en un arreglo.

Básicamente hay que ordenar el arreglo, puede ser en orden decreciente o creciente, te recomiendo revisar el material de apoyo que tienes en el aula virtual. Una vez revisado, comencemos:

Vamos a practicar, haz una función para cargar el arreglo tipo struct personas, muestra en pantalla su contenido y verifica si los datos mostrados son los correctos.

Una vez revisado el correcto funcionamiento de las funciones realizadas, procede a investigar lo siguiente:

- ❖ Defina y explique la importancia de un algoritmo de ordenamiento.
- ❖ Definición y uso de los algoritmos de ordenamiento
 - **Métodos principales:**
 1. por selección,
 2. de burbuja (bubble-sort),
 3. por inserción,
 4. por incrementos (Shell-sort)
 5. Por mezcla (merge-sort),
 6. por montículos (Heapsort) y

7. rápido (quicksort).

Es importante comprender cada método, que significa, como funciona, representarlo gráficamente y realiza una corrida en frío con el arreglo de Personas.

Como actividad práctica realiza lo siguiente:

- ❖ Procede a implementar cada método en lenguaje C, haz una función para cada uno
- ❖ Comprueba cada método, pásale a la función por parámetros un arreglo de personas y ordénalo por edad, en forma de menor a mayor y de mayor a menor.
- ❖ Verifica si los resultados son los correctos.
- ❖ Para cada prueba de cada método, toma un capture de pantalla de los datos ingresados y la corrida, y los pegas en un archivo Word, identifica el archivo con tu apellido_nombre_ordenamiento y envíalo al aula virtual.
- ❖ Finalmente explica con tus propias palabras que características observas en cada método y cuál de los métodos estudiados es el mejor.

Una vez revisado cada método, estás listo para realizar este ejercicio donde requieres aplicar todos los conocimientos adquiridos hasta ahora: De acuerdo al último nro., de tu cedula, selecciona el método que corresponde de los 7 algoritmos listados y aplícalo al siguiente ejercicio. Aquellos estudiantes cuyo nro., de cedula termine en **8,9 y 0 aplicaran el método Quicksort al ejercicio.**

El Banco Caroní, desea automatizar todas sus operaciones bancarias. Por tal razón desea contratar sus servicios, con la finalidad que Ud. diseñe una Estructura de Datos en Lenguaje C que permita manejar de una manera fácil y eficiente todas sus operaciones considerando los siguientes requerimientos:

- a. El banco maneja básicamente tres (3) tipos de cuentas: cuentas de ahorro, corrientes y de activos líquidos; además por cada una de ellas deberá llevarse la tasa de rendimiento correspondiente.

b. Para cada uno de los clientes se deberá conocer la siguiente información: Número de la Cuenta, Tipo de Cuenta, Nombre, Dirección, Teléfono, Saldo en la Cuenta, Cedula de Identidad, etc.

c. Deberá existir un registro de todas las transacciones realizadas por el Banco: Código de la Transacción, Fecha de la misma, Número de Cuenta Involucrada, Monto en Bolívares y el Número de la caja donde fue realizada.

Con el modelo diseñado, programe lo siguiente:

- a. Mostrar en pantalla, todos los clientes del banco ordenados en forma creciente por el nro., de cedula.
- b. Mostrar en pantalla, todos los clientes con Cuenta Corriente, ordenados alfabéticamente en orden creciente por el Apellido.
- c. Mostrar en pantalla, todos los clientes con Cuenta de Ahorros, ordenados alfabéticamente en orden decreciente por el Nombre.
- d. Mostrar todas las transacciones de la cuenta de ahorros nro. 010101088 desde el día 15-01-2024 al 31-05-2024.
- e. El cajero encargado de la caja 1, desea realizar su cuadre de caja para el día actual de trabajo. Para ello desea obtener la siguiente información:
 - Todos los cheques cobrados.
 - Todos los depósitos realizados.

Recuerda enviar al aula virtual el archivo .pdf de toda la actividad (incluir captures de la corrida) y el código .c del ejercicio.

Fecha de entrega: 20 de Junio 2024 a las 11:00 am