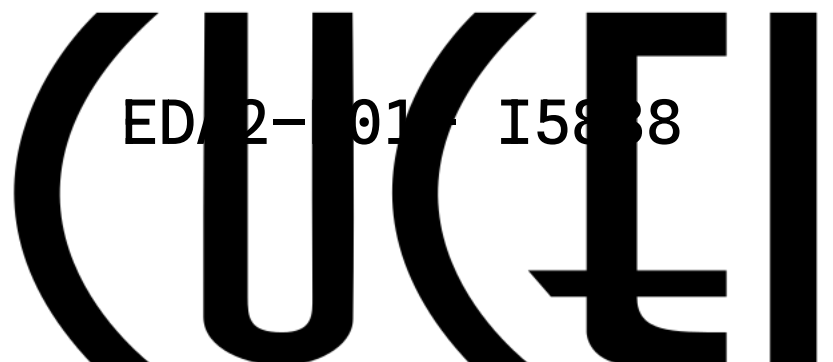


# Programa 04

## Indices y serialización

Profesor: Noé Ortega Sanchez

Martínez Orozco Victor Manuel



El programa es prácticamente la implementación de una tabla hash. Esta tabla con el fin de cumplir los requisitos solicitados almacena un tipo de dato denominado registro. Este dato registro contiene 3 atributos; su id que es un dato tipo entero para lograr calcular la ubicación de los datos en la tabla, un nombre que es un dato de tipo char y un número de teléfono que también es un char.

Cabe aclarar que los métodos implementados en este programa son los más simples e indispensables para poder operar con los datos. Claro está que los mismos son totalmente funcionales y aparentemente no presentan ningún error.

Ignorando que la clase registro tiene propiamente implementados sus métodos getter y setter junto con otro método para mostrar, la tabla implementa los siguientes métodos:

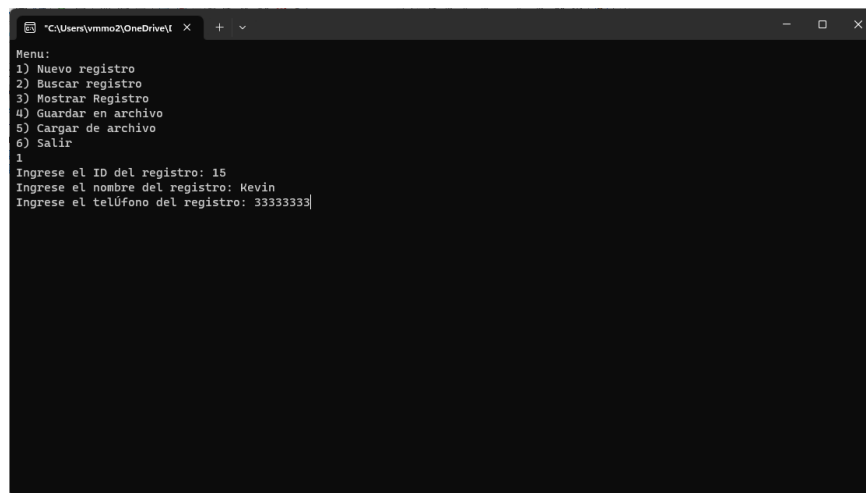
- Tabla de índices: esta sirve principalmente para crear como tal la “lista de listas” y principalmente darle un lugar y utilidad a la función hash. Este método consta únicamente de una iteración posprueba que inicializa los espacios correspondientes a la misma dependiendo el tamaño asignado a la constante antes de ejecutar el programa.
- Función hash: Esta función retorna un entero que servirá para asignar el espacio correspondiente al dato en la tabla y para ello, solamente recibe el id del dato a insertar y regresa el residuo entre el id y el tamaño de la tabla de índices.
- Buscar: Este método retorna un dato tipo puntero a nodo y recibe el id del dato a buscar. Para ello, se declarará un entero con el valor que retorne la función hash mandando como parámetro el id que recibió el método de búsqueda. Posteriormente con un puntero a nodo igualado al dato (en este caso tipo registro) que se encuentre en la tabla en el espacio que regresó la función hash, se verifica si este es diferente de nulo. En caso de ser nulo, el dato no existe. Caso contrario, solamente se imprime el dato junto con todos sus campos.
- Mostrar: Este método únicamente recorre la lista mediante un ciclo for que usa como criterio de paro el tamaño máximo de la tabla. Para esto, únicamente se declara un puntero a nodo inicializado al inicio de la tabla y mientras la dirección del puntero a nodo sea diferente de nula, se recorre de inicio a fin la tabla y se imprimen en pantalla los campos del nodo. Una vez siendo nulo el nodo, el método cancela la ejecución.
- Guardar en archivo: Este método utiliza un objeto de la clase ofstream para crear y abrir un archivo binario llamado "file01" en modo escritura. Posteriormente, recorre la tabla de índices y para cada registro almacenado en un nodo, utiliza la función write del objeto archivo para escribir la información en el archivo. La función “reinterpret\_cast” sirve para convertir el puntero al objeto registro en un puntero a un bloque de memoria de tamaño

sizeof(Registro) bytes, de modo que se pueden escribir todos los bytes que componen el objeto.

- Cargar desde archivo: Este método utiliza un principio muy similar al de guardado de datos en archivo, con la única diferencia de que mediante un bucle while lee los datos del archivo y los inserta en la tabla de índices, reusando el método propio de la tabla "insertar".  
El operador new crea un nuevo objeto de tipo registro y el operador reinterpret\_cast para convertir el puntero al objeto registro leído del archivo en un puntero a un objeto de la clase registro.

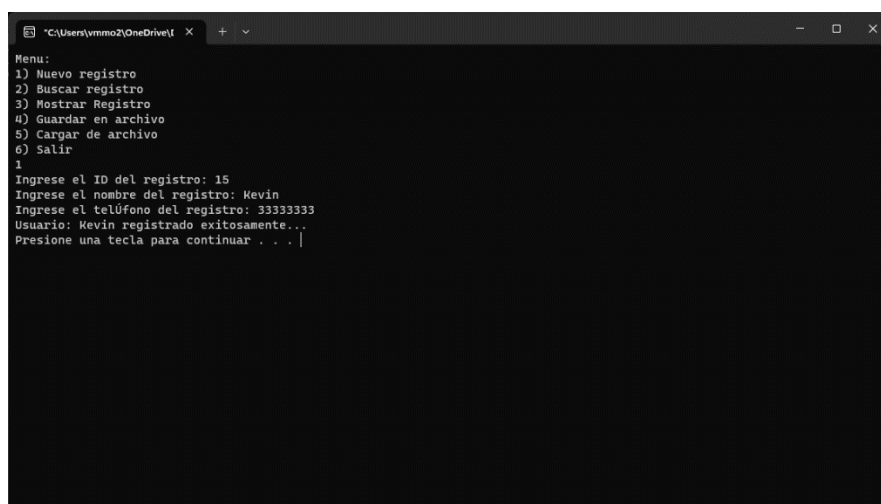
## Impresiones de pantalla

### Menú principal:



```
C:\Users\ymmo2\OneDrive\I >
Menu:
1) Nuevo registro
2) Buscar registro
3) Mostrar Registro
4) Guardar en archivo
5) Cargar de archivo
6) Salir
1
Ingrese el ID del registro: 15
Ingrese el nombre del registro: Kevin
Ingrese el teléfono del registro: 33333333
```

### Nuevo registro:



```
C:\Users\ymmo2\OneDrive\I >
Menu:
1) Nuevo registro
2) Buscar registro
3) Mostrar Registro
4) Guardar en archivo
5) Cargar de archivo
6) Salir
1
Ingrese el ID del registro: 15
Ingrese el nombre del registro: Kevin
Ingrese el teléfono del registro: 33333333
Usuario: Kevin registrado exitosamente...
Presione una tecla para continuar . . . |
```

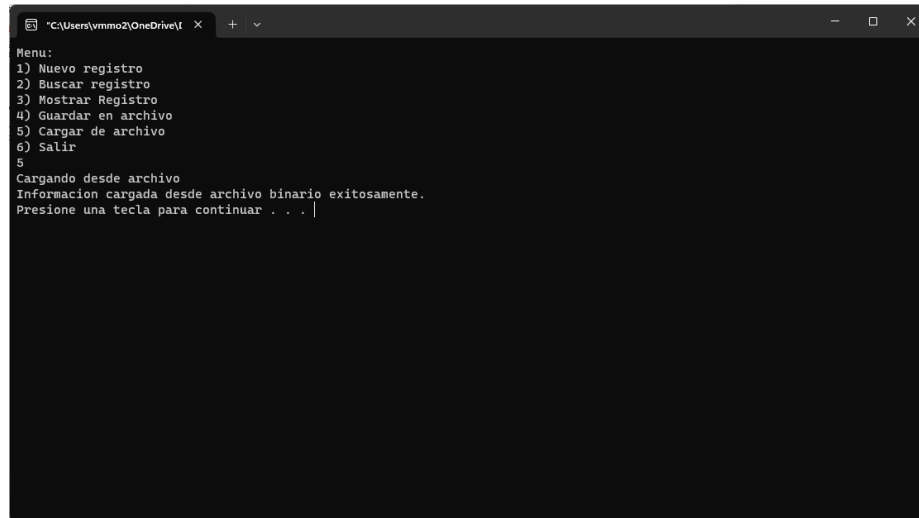
## Buscar registro:

```
"C:\Users\ymmo2\OneDrive\I  X + v
Menu:
1) Nuevo registro
2) Buscar registro
3) Mostrar Registro
4) Guardar en archivo
5) Cargar de archivo
6) Salir
3
ID: 12
Nombre: vianey
Telefono: 33242526
ID: 15
Nombre: Kevin
Telefono: 33333333
ID: 23
Nombre: victor
Telefono: 333222
Presione una tecla para continuar . . . |
```

## Guardar en archivo:

```
"C:\Users\ymmo2\OneDrive\I  X + v
Menu:
1) Nuevo registro
2) Buscar registro
3) Mostrar Registro
4) Guardar en archivo
5) Cargar de archivo
6) Salir
4
Guardando en archivo
Informacion guardada en archivo binario exitosamente.
Presione una tecla para continuar . . . |
```

## Cargar en archivo:



```
"C:\Users\ymmo2\OneDrive\... X + v
Menu:
1) Nuevo registro
2) Buscar registro
3) Mostrar Registro
4) Guardar en archivo
5) Cargar de archivo
6) Salir
5
Cargando desde archivo
Informacion cargada desde archivo binario exitosamente.
Presione una tecla para continuar . . . |
```

En conclusión, el tema de las tablas hash es una estructura de datos eficiente para almacenar y buscar información en una colección de datos y cuando se abarca el tema del manejo de datos en documentos de texto usando esta estructura, se crea un complemento muy bueno. Simplemente, la estrategia implementada para guardar y cargar la información desde la tabla hash y los documentos txt, es muchísimo más simple, más eficiente y a comparación de las otras estrategias implementadas anteriormente, esta es la que a mi parecer consideraría la mejor.