



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

### Ingeniero en computación

### Ingeniero en Software y tecnologías emergentes

**Materia:** Programación Estructurada / Clave 36276

**Alumno:** Kevin Alejandro Gonzalez Torres

**Matrícula:** 372354

**Maestro:** Pedro Núñez Yépiz

**Actividad Numero:** 14

**Tema - Unidad:** Archivos binarios

**Ensenada Baja California a 28 de Noviembre del 2022**



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

### 1. INTRODUCCIÓN

Este código en C utiliza archivos binarios para almacenar información persistente sobre trabajadores en una organización. Las funciones realizan operaciones como agregar, eliminar, buscar y ordenar registros directamente en el archivo binario. La eficiencia se mejora mediante el uso de índices. Estos archivos binarios proporcionan una forma eficaz y directa de manejar datos, permitiendo la persistencia de la información y facilitando operaciones sobre los registros de los trabajadores de manera organizada.

### 2. COMPETENCIA

- Programación en C
- Diseño de estructuras de datos
- Manipulación de datos complejos
- Uso de typedef
- Validación y manejo de excepciones
- Desarrollo de software que cumple con requerimientos específicos
- Integridad y precisión de datos
- Uso y manejo de archivos binarios

### 3. FUNDAMENTOS

Cadenas:

<https://www.programiz.com/c-programming/c-strings>

Funciones:

[https://www.w3schools.com/c/c\\_functions.php](https://www.w3schools.com/c/c_functions.php)

Typedef:

<https://www.geeksforgeeks.org/typedef-in-c/>

<https://learn.microsoft.com/es-es/cpp/c-language/typedef-declarations?view=msvc-170>

Struct:

[https://www.w3schools.com/c/c\\_structs.php](https://www.w3schools.com/c/c_structs.php)

<https://www.geeksforgeeks.org/structures-c/>

Archivos binarios:

<https://computer.howstuffworks.com/c39.htm>



#### 4. PROCEDIMIENTO

## ACTIVIDAD 14

### Archivos Binarios (archivos indexados)

#### MENÚ

- 1.- AGREGAR
- 2.- ELIMINAR
- 3.- BUSCAR
- 4.- ORDENAR
- 5.- IMPRIMIR REGISTROS ARCHIVO ORIGINAL
- 6.- IMPRIMIR REGISTROS ARCHIVO ORDENADO
- 7.- GENERAR ARCHIVO TEXTO
- 8.- EMPAQUETAR
- 0.- SALIR

**INSTRUCCIONES:** Programa que contenga el menú anterior, el programa utiliza un vector de índices de la siguiente estructura: [ llave, índice] donde *el campo llave es noemplado*. *registros.dat* es el archivo con los registros a cargar en el vector de índices *archivo binario sera proporcionado*,

**CARGAR ARCHIVO :** El programa deberá cargar al arrancar el programa, el archivo Binario generará el vector de índices (llave, indice) *sólo con registros válidos* (el tamaño del vector debera ser 25% mas grande que el la cantidad de registros que contenga el archivo binario ) utiliza un archivo externo para averiguar tamaño y retorne cantidad de registros.



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

### 5. RESULTADOS Y CONCLUSIONES

El código proporciona un sistema de gestión de registros de trabajadores eficiente y funcional. Las operaciones ofrecidas, como agregar, eliminar, buscar, ordenar y generar archivos de texto, facilitan la manipulación de datos de manera versátil. La persistencia de datos en archivos binarios garantiza la integridad y disponibilidad de la información incluso entre sesiones del programa. La generación aleatoria de datos para registros de prueba añade una característica práctica, permitiendo probar el sistema con conjuntos de datos variados.

La implementación de algoritmos de búsqueda y ordenamiento, como búsqueda lineal, búsqueda binaria, selección y quicksort, contribuye a la eficiencia del programa, especialmente al manejar grandes conjuntos de datos.

Eficiencia en Operaciones:

- La aplicación destaca por la eficiencia en búsquedas y operaciones, especialmente al aprovechar algoritmos de búsqueda y ordenamiento optimizados.

Persistencia de Datos:

- El uso de archivos binarios para almacenar y cargar registros ofrece una solución robusta y persistente, garantizando la integridad de los datos a lo largo del tiempo.

Versatilidad y Usabilidad:

- La variedad de operaciones proporcionadas en el menú y la generación aleatoria de datos demuestran una consideración hacia la versatilidad y usabilidad del sistema.

Organización y Claridad:

- La estructura modular del código, con funciones claramente definidas, contribuye a la legibilidad y mantenimiento del programa.

### 6. ANEXOS

GTKA\_RP13\_PE\_ANEXOS



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

### 7. REFERENCIAS

#### **Diseño de algoritmos y su codificación en lenguaje C**

Corona, M.A. y Ancona, M.A. (2011)..

España: McGraw-Hill.

ISBN: 9786071505712

#### **Programación estructurada a fondo: implementación de algoritmos en C**

:Pearson Educación. Sznajdleder, P. A. (2017)..

Buenos Aires, Argentina: Alfaomega

#### **Como programar en C/C++**

H.M. Deitel/ P.J. Deitel

Segunda edición

Editorial: Prentice Hall.

ISBN: 9688804711

#### **Programación en C. Metodología, estructura de datos y objetos**

Joyanes, L. y Zahonero, I. (2001)..

España: McGraw-Hill.

ISBN: 8448130138