



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: Villalobos Ensaldo Luis Daniel

Matrícula: 368617

Maestro: Pedro Núñez Yépiz

Actividad No. : 13

Tema - Unidad : Archivos

Ensenada Baja California a 20 de 05 del 2024



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

NOTA: guardar el archivo de la sig forma
INICIALES (numero práctica)_PE
ejemplo: Francisco Antonio Perez Lopez
PLFA_RP08_PE.pdf

1. INTRODUCCIÓN

Los archivos binarios son tipos de archivos que almacenan datos en formato binario, es decir, en una forma que está directamente interpretable por la máquina, a diferencia de los archivos de texto, que almacenan caracteres de texto codificados

2. COMPETENCIA

Que el alumno comprenda la lectura y escritura de archivos, para así hacer funciones que puedan extraer datos de los mismos de manera eficiente y lógica

3. FUNDAMENTOS

Un campo llave o clave primaria es un campo en una estructura o base de datos que se utiliza para identificar de manera única cada registro.

Abrir un Archivo Binario

Para trabajar con archivos binarios en C++, se utilizan las funciones fopen, fread, fwrite, fseek, y fclose de la biblioteca estándar de C

Eficiencia: Los archivos binarios son más compactos y rápidos de procesar que los archivos de texto porque no requieren conversión de datos.

Formato de Almacenamiento: Almacenan datos en el mismo formato que se utilizan en la memoria, sin conversión a texto.

Uso Típico: Se utilizan comúnmente para almacenar estructuras de datos complejas, imágenes, audio, video, y otros tipos de datos que no son fácilmente representables como texto.



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

4. PROCEDIMIENTO



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

ACTIVIDAD 13

REALICE EL SIGUIENTE PROGRAMA QUE CONTENGA UN MENÚ.

MENÚ

- 1.- AGREGAR (AUTOM 100 REGISTROS)
- 2.- EDITAR REGISTRO
- 3.- ELIMINAR REGISTRO (lógico)
- 4.- BUSCAR
- 5.- ORDENAR
- 6- IMPRIMIR
- 7.- GENERAR ARCHIVO TEXTO
- 8.- VER ARCHIVO TEXTO
- 9.- CREAR ARCH BINARIO
- 10.- CARGAR ARCH BINARIO
- 11.- MOSTRAR ELIMINADOS
- 0.- SALIR

UTILIZAR UN ARREGLO DE 5000 REGISTROS

SE DEBERÁ UTILIZAR ESTRUCTURAS CON LOS DATOS BÁSICOS DE UN EMPLEADO

preguntar nombre de archivo binario o de archivo texto

Busqueda y Ordenacion por CAMPO LLAVE

nota: usar librería propia con funciones

nota2: 100 % validado, Cuidar desbordamiento de vector

nota3: Campo llave matricula no repetido, archivos solo cargar 1 sola vez.

nota4: Usar el tipo Tkey para hacer mas practico el programa

AYUDA:

<https://www.fing.edu.uy/tecnoinf/mvd/cursos/prinprog/material/teo/prinprog-teorico08.pdf>

<https://kesquivel.files.wordpress.com/2013/05/estructuras2013final2.pdf>

http://platea.pntic.mec.es/vgonzale/cyr_0204/cyr_01/control/lengua_C/estructuras.htm

INSTRUCCIONES DEL MENU

- 1.- **Agregar** : El programa deberá ser capaz de agregar 100 registros al vector de registros (**Generar automáticamente los datos**).
- 2.- **Editar Registro** : El programa deberá buscar una matrícula en el vector por medio del método de búsqueda más óptimo. **Mostrar los datos en forma de registro** Preguntar que campo quiere Editar, actualizar los datos en el vector (**solo a registros activos**)
- 3.- **Eliminar Registro** : El programa deberá buscar una matrícula en el vector por medio del método de búsqueda más óptimo. Utilizar banderas para escoger el método más adecuado., imprimir el registro y preguntar si se quiere eliminar el registro.
- 4.- **Buscar** : El programa deberá buscar una matrícula en el vector por medio del método de búsqueda más óptimo. Utilizar banderas para escoger el método más adecuado. **Mostrar los datos en forma de registro**
- 5.- **Ordenar** : El programa deberá ordenar el vector por medio del método de ordenación más óptimo. Utilizar banderas para escoger el método más adecuado se ordenará por el **campo llave (matrícula)**
- 6.- **Imprimir:** El programa deberá mostrar todos los registros del vector y como están en ese momento ordenado o desordenado. (**mostrar en forma de tabla**)



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

- 7.- Generar Archivo Texto :** El programa deberá preguntar al usuario el nombre del archivo, **solo nombre sin extensión**, el programa generará un archivo con el nombre proporcionado por el usuario con **extensión .txt** los datos que pondrá en el archivo de texto serán idénticos a los contenidos en el Vector de registros. (ordenado o desordenado). El programa podrá generar múltiples archivos para comprobar las salidas.
- 8.- Mostrar Archivo Texto:** El programa deberá preguntar al usuario el nombre del archivo, **solo nombre sin extensión**, el programa generará un archivo con el nombre proporcionado por el usuario con **extensión .txt** mostrar el archivo de texto tal y como se encuentra.
- 9.- Crear archivo binario :** El programa deberá crear un archivo binario con los datos del vector actualizados, sustituir el archivo base, realizar respaldo del archivo anterior y guardarlo con el mismo nombre pero extensión .tmp (validar msges si el archivo no se puede crear por falta de registros en el vector)
- 10 .- Cargar Archivo Binario :** El programa deberá cargar al vector los registros del archivo binario **(solo podrá cargarse una sola vez el archivo, el archivo binario se deba llamar datos.dll y si no existe deba indicar)**
- 11.- Mostrar Borrados:** El programa deberá mostrar del archivo binario solo los registros que se eliminaron (marcados con status 0) y que fueron marcados en su momento como registros eliminados.

5. RESULTADOS Y CONCLUSIONES

Creo que me pareció un poco más fácil los archivos binarios que los de texto, por lo menos al leer los archivos y escribir información me resultó menos confuso, de igual manera no fue tan diferente de las actividades anteriores solo fue agregar la función de editar que fue algo larga pero simple, y las diferentes funciones de archivos

6. ANEXOS

CAPTURA DE PANTALLAS DEL PROGRAMA EJECUTÁNDOSE, DONDE SE ABORDA CADA UNO DE LOS PUNTOS CUBIERTOS DE LA PRÁCTICA

PONER COMO ANEXO, PDF CON CÓDIGO Y CAPTURA DE PANTALLAS...



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño



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

Computologos. (2012, March 30). *CAMPO LLAVE, LONGITUD FIJA y VARIABLE*. Archivostruc2012.

<https://archivostruc2012.wordpress.com/2012/03/06/campo-llave-longitud-fija-y-variable/>

C++ - 8.5. *Ficheros binarios* - AprendeAProgramar.com. (n.d.).

<https://www.aprendeaprogramar.com/cursos/verApartado.php?id=16801>

Moisset, D. (n.d.). *Archivos binarios: agregar, consultar y modificar registros (struct) en un archivo*.

<https://www.tutorialesprogramacionya.com/cya/detalleconcepto.php?punto=79&codigo=7>

[9&inicio=75](https://www.tutorialesprogramacionya.com/cya/detalleconcepto.php?punto=79&codigo=7)