

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA



Ingeniería en Computación

Actividad 13

Programación Estructurada

ALUMNO: Arredondo Urbalejo Isai Alexis

MATRÍCULA: 368747

GRUPO: 932

PROFESOR: Pedro Nunez Yepiz



Introducción

Los archivos binarios son un tipo de archivo que almacena datos en forma binaria, es decir, en forma de bits. Esto los diferencia de los archivos de texto, que almacenan datos en forma de caracteres.

Los archivos binarios se utilizan en una gran variedad de aplicaciones, como el almacenamiento de datos de aplicaciones, el intercambio de datos entre aplicaciones y la creación de archivos ejecutables.

En C++, los archivos binarios se pueden manipular utilizando las funciones de la biblioteca estándar de entrada/salida (I/O). Estas funciones permiten leer, escribir y manipular datos en archivos binarios.

Competencia

Aprender a manipular tanto archivos binarios como archivos de texto, utilizando las diferentes librerías y funciones proporcionadas por el profesor.

Fundamentos

Los archivos binarios son un tipo de archivo que almacena datos en forma binaria, es decir, en forma de bits. Esto los diferencia de los archivos de texto, que almacenan datos en forma de caracteres.

Los archivos binarios se utilizan en una gran variedad de aplicaciones, como el almacenamiento de datos de aplicaciones, el intercambio de datos entre aplicaciones y la creación de archivos ejecutables.

En C++, los archivos binarios se pueden manipular utilizando las funciones de la biblioteca estándar de entrada/salida (I/O). Estas funciones permiten leer, escribir y manipular datos en archivos binarios.

Definición

Un archivo binario es un archivo que almacena datos en forma binaria. Esto significa que los datos se almacenan como una secuencia de bits, en lugar de como una secuencia de caracteres.



Los archivos binarios se utilizan para almacenar una amplia variedad de datos, como números, caracteres, estructuras de datos y objetos.

Funciones de la biblioteca estándar de I/O para archivos binarios

La biblioteca estándar de C++ proporciona una serie de funciones que se pueden utilizar para manipular archivos binarios. Estas funciones se encuentran en el encabezado `<fstream>`.

Las funciones más importantes para manipular archivos binarios son las siguientes:

`<fopen()>`: abre un archivo binario para lectura, escritura o ambos.

`<fclose()>`: cierra un archivo binario.

`<fread()>`: lee datos de un archivo binario.

`<fwrite()>`: escribe datos en un archivo binario.

Acceso secuencial a archivos binarios

El acceso secuencial a archivos binarios significa que los datos se leen o escriben en el archivo en orden secuencial.

Para acceder secuencialmente a un archivo binario, se utilizan las funciones `<fread()>` y `<fwrite()>`.

La función `<fread()>` lee un bloque de datos de un archivo binario. El bloque de datos puede contener cualquier tipo de datos, incluidos números, caracteres y estructuras de datos.

La función `<fwrite()>` escribe un bloque de datos en un archivo binario. El bloque de datos puede contener cualquier tipo de datos, incluidos números, caracteres y estructuras de datos.



Acceso aleatorio a archivos binarios

El acceso aleatorio a archivos binarios significa que los datos se pueden leer o escribir en cualquier orden en el archivo.

Para acceder aleatoriamente a un archivo binario, se utilizan las funciones `fseek()` y `fread()`.

La función `fseek()` mueve el puntero de archivo a una posición específica en el archivo.

La función `fread()` lee un bloque de datos de un archivo binario. El bloque de datos puede contener cualquier tipo de datos, incluidos números, caracteres y estructuras de datos.

Greyrat, R. (2022, 5 julio). *Diferencia entre archivo de texto C++ y archivo binario* – Barcelona Geeks.

<https://barcelonageeks.com/diferencia-entre-archivo-de-texto-c-y-archivo-binario/#:~:text=Los%20archivos%20de%20texto%20est%C3%A1n%20organizados%20en%20l%C3%A2n%2C,almacenan%20en%20la%20memoria%20principal%20para%20su%20procesamiento.>

C++ - 8.5. Ficheros binarios - AprendeAProgramar.com. (s. f.).

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

Archivos Binarios C++ [8x4e8y9xqg43]. (s. f.).

<https://idoc.pub/documents/archivos-binarios-c-8x4e8y9xqg43>



Procedimiento

CONTENIDO DE LA ACTIVIDAD

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

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**)



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.

Resultados y conclusiones

Los archivos binarios son un tipo de archivo esencial para muchas aplicaciones en C++. Las funciones de la biblioteca estándar de I/O proporcionan una forma eficiente y versátil de manipular archivos binarios.

Los archivos binarios se utilizan en una amplia variedad de aplicaciones, como el almacenamiento de datos de aplicaciones, el intercambio de datos entre aplicaciones y la creación de archivos ejecutables.