



UNIVERSIDAD DE GUADALAJARA

CUCEI

DIVISIÓN DE TECNOLOGÍAS PARA LA INTEGRACIÓN CIBER-HUMANA

DEPARTAMENTO DE CIENCIAS COMPUTACIONALES

PRÁCTICA No. 1

TEMA: Archivos de Acceso Secuencial (Consultas Filtradas)

EQUIPO No. 4

INTEGRANTES:

Munguía Guízar Marlon Uriel

Padilla Perez Jorge Daray

Pérez Palacios Perla Michelle

Pulido Tobias Rafael Agustin

**SEMINARIO DE SOLUCIÓN DE PROBLEMAS DE ESTRUCTURAS DE DATOS
II | SECCIÓN D19 | 2022B**

PROF. Mariscal Lugo Luis Felipe

MARCO TEÓRICO

¿Qué es un archivo físico?

Archivo que en realidad existe en el almacenamiento secundario. Es el archivo tal como lo conoce el sistema operativo y que aparece en su directorio de archivos.

¿Qué es un archivo lógico?

El archivo, visto por el programa. El uso de archivos lógicos permite a un programa describir las operaciones que van a efectuarse en un archivo sin saber cuál archivo físico se usará. El programa puede entonces usarse para procesar cualquiera de diversos archivos que compartan la misma estructura.

¿Qué clases se utilizan en el manejo de archivos en C++?

El archivo de cabecera `fstream.h` define las clases `ifstream`, `ofstream` y `fstream` para operaciones de lectura, escritura y lectura/escritura en archivos respectivamente. Para trabajar con archivos debemos crear objetos de estas clases de acuerdo a las operaciones que deseamos efectuar. Empezamos con las operaciones de escritura, para lo cual básicamente declaramos un objeto de la clase `ofstream`, después utilizamos la función miembro `open` para abrir el archivo, escribimos en el archivo los datos que sean necesarios utilizando el operador de inserción y por último cerramos el archivo por medio de la función miembro `close`.

¿Qué modos de apertura tiene C++?

- **`ios::app`**: Añade toda la salida al final del archivo.
- **`ios::ate`**: Abre un archivo en modo de salida y se desplaza hasta el final del archivo (por lo general se utiliza para añadir datos a un archivo). Los datos se pueden escribir en cualquier parte del archivo.
- **`ios::in`**: Abre un archivo en modo de entrada.
- **`ios::out`**: Abre un archivo en modo de salida.
- **`ios::trunc`**: Descarta el contenido, si es que existe (también es la acción predeterminada para `ios::out`).
- **`ios::binary`**: Abre un archivo en modo de entrada o salida binaria (es decir, que no es texto).

¿Para qué sirve la función `eof()`?

Se usa para verificar si la secuencia del archivo genera algún error, `eof` (significa End Of File o Fin de Archivo).

Básicamente nos indica si hay o no más información que se pueda obtener del archivo, después de que se intenta extraer datos después del final del flujo.

¿Para qué sirve la función `seekg()`?

reposiciona el apuntador de posición del archivo y lo coloca al principio del archivo (ubicación 0) adjunto a `archivoClientesEntrada`. El argumento para `seekg` es comúnmente un entero `long`.

Se puede especificar un segundo argumento para indicar la dirección de búsqueda, que puede ser `ios::beg` (la opción predeterminada) para un posicionamiento relativo al inicio de un flujo, `ios::cur` para un posicionamiento relativo a la posición actual en un flujo, o `ios::end` para un posicionamiento relativo al final de un flujo.

El apuntador de posición del archivo es un valor entero que especifica la ubicación en el archivo como un número de bytes desde la ubicación inicial del archivo (a ésta también se le conoce como el desplazamiento desde el inicio del archivo). Algunos ejemplos de cómo posicionar el apuntador “obtener” de posición del archivo son:

- `// se posiciona en el n-ésimo byte de objetoArchivo (asumiendo ios::beg)`
`objetoArchivo.seekg(n);`
- `// se posiciona n bytes hacia adelante en objetoArchivo`
`objetoArchivo.seekg(n, ios::cur);`
- `// se posiciona n bytes hacia atrás desde el final de objetoArchivo`
`objetoArchivo.seekg(n, ios::end);`
- `// se posiciona al final de objetoArchivo`
`objetoArchivo.seekg(0, ios::end);` Se pueden realizar las mismas operaciones mediante la función miembro `seekp` de `ostream`.

¿Para qué sirve la función `clear()`?

Esta función nos permite restaurar el estado de un flujo a “bueno (Goodbit)” de manera que la entrada/Salida pueda proceder en el flujo.

¿Para qué sirve la función `is_open()`?

Esta función revisa si un archivo está abierto, Devuelve si es que la secuencia está actualmente asociada a un archivo.

Los flujos se pueden asociar a archivos mediante una llamada exitosa a la apertura de miembros o directamente en la construcción, y se pueden desasociar llamando al cierre o la destrucción.

Bibliografía:

- Deitel, P. J., Deitel, P. J., & Elizondo, A. V. R. (2008). C++. Cómo programar (6.a ed.). Pearson Educación.
- C++ Reference. (2022). Cplusplus.com.
https://cplusplus.com/reference/fstream/fstream/is_open/#:~:text=std%3A%3Afstream%3A%3Ais_open&text=Returns%20whether%20the%20stream%20is,calling%20close%20or%20on%20destruction.


CÓDIGO FUENTE

[illegible]

[illegible]

```
108         getch();
109         break;
110
111         case 3:
112             system("cls");
113             cout<<"Deportistas adultos mayores:"<<endl;
114
115             cout<<left<<setw(10)<<"NOMBRE"<<setw(13)<<"EDAD"<<"DEPORTE"<<endl<<fixed<
116             <showpoint; //imprime la cabecera de la tabla
117
118             while (archivoDeportistasEntrada >> nombre >> edad >>
119             deporte)
120             {
121
122                 if (edad>59 && edad<86)
123                 {
124                     mostrarLinea(nombre, edad, deporte);
125                 }
126             }
127
128             getch();
129             getch();
130             break;
131         }
132     }while (seleccion != 0);
133
134     return 0;
135 } //main
136
137 // muestra un solo registro del archivo
138 void mostrarLinea(const char * const nombre, int edad, const char *
139 const deporte)
140 {
141     cout<<left<<setw(10)<<nombre<<setw(13)<<edad<<setw(7)<<deporte<<endl;
142 } //mostrar linea
143
```

IMAGEN(ES) DEL ARCHIVO(S) FÍSICO(S)

 deportistas.txt: Bloc de notas

Archivo Edición Formato Ver Ayuda

```
sayde 15 Natacion
Francisco 40 Futbol
Rafael 16 Voleibol
Alfredo 65 KickBoxing
Josue 37 Atletismo
Arturo 66 Beisbol
```


IMÁGENES DE LA CORRIDA DEL PROGRAMA

Menú del programa.

[illegible]

Opción 1.

Deportistas menores de edad:		
NOMBRE	EDAD	DEPORTE
Sayde	15	Natacion
Rafael	16	Voleibol

Opción 2.

Deportistas mayores de edad:		
NOMBRE	EDAD	DEPORTE
Francisco	40	Futbol
Josue	37	Atletismo

Opción 3.

Deportistas adultos mayores:		
NOMBRE	EDAD	DEPORTE
Alfredo	65	KickBoxing
Arturo	66	Beisbol

Opción salir.

>>>>>>>>>> Saliendo <<<<<<<<<<<

```
Process returned 0 (0x0)   execution time : 239.994 s
Press any key to continue.
```

Conclusiones:

- Munguía Guízar Marlon Uriel.

Esta primera práctica me pareció tanto interesante como a la vez educativa, ya que, al poder comenzar a escribir el código desde el aula con ayuda del libro de texto, no fue difícil comprender la forma de resolver la problemática.

- Padilla Perez Jorge Daray.

Mi conclusión es que la práctica es un tanto sencilla, aunque te obliga e incita a estudiar más y más para mejorar tu programación, además de que me parece super importante el correcto aprendizaje de las funciones anteriormente vistas, me gusto la práctica y la explicación de la clase como apoyo para su correcta resolución.

- Pérez Palacios Perla Michelle.

Lo anterior nos permite aprender más sobre el manejo de archivos, pudiendo entender de mejor manera la entrada y salida de datos. También el ver los diferentes modos de apertura de un archivo y reforzar los conocimientos de las primeras tareas.

- Pulido Tobias Rafael Agustin

Los archivos secuenciales tienen muchas aplicaciones desde el punto de guardar registros de información o bien extraer los datos, me quedó claro las limitantes que tiene un archivo de acceso secuencial. Por otro lado el aprender el funcionamiento de funciones como eof() ; o seekg() ; facilitan mucho el recorrer el archivo y manejar correctamente la información de este.