

Carátula

**Universidad Mariano Gálvez de Guatemala
Boca del Monte**

Ingeniería en Sistemas. Ciclo II, "c"
Jornada Sábado.
ALGORITMOS
MIGUEL PICHYA



Nombre: Luis Fernando Lima Ixcuná
Carné: 7690-20-17409

Índice

[Carátula](#)

[Índice](#)

[Introducción](#)

[Objetivos](#)

[Conclusión](#)

[Bibliografía](#)

[Librerías](#)

[Manual de usuario](#)

[Codigo fuente](#)

Introducción

C++ es un lenguaje de programación de propósito general y de alto nivel con facilidades para la programación a bajo nivel. Tiene un modelo de tipos de dato estático y es un lenguaje que soporta tres paradigmas de programación: programación procedimental, programación orientada a objetos y programación genérica. Cuenta con abstracción de datos e identificación de tipo de dato en tiempo de ejecución. Desde 1990, C++ ha sido uno de los lenguajes comerciales de programación más populares¹.

Desarrollado originalmente por Bjarne Stroustrup en 1983 en los Laboratorios Bell, es empleado desde entonces y hasta el día de hoy. A pesar de su antigüedad, el primer estándar del lenguaje fue ratificado en 1998 como ISO/IEC 14882:1998, cuya última versión se dio en 2003 bajo ISO/IEC 14882:2003. Actualmente se está desarrollando una nueva versión del lenguaje (informalmente conocida como C++0x).

Objetivos

Alto rendimiento: Es una de sus principales características, el alto rendimiento que ofrece. Esto es debido a que puede hacer llamadas directas al sistema operativo, es un lenguaje compilado para cada plataforma, posee gran variedad de parámetros de optimización y se integra de forma directa con el lenguaje ensamblador.

Lenguaje actualizado: A pesar de que ya tiene muchos años, el lenguaje se ha ido actualizando, permitiendo crear, relacionar y operar con datos complejos y ha implementado múltiples patrones de diseño.

Multiplataforma

Extendido: C y C++ están muy extendidos. Casi cualquier programa o sistema están escritos o tienen alguna parte escrita en estos lenguajes (desde un navegador web hasta el propio sistema operativo).

Las principales desventajas de C++ es que se trata de un lenguaje muy amplio (con muchos años y muchas líneas de código), tiene que tener una compilación por plataforma y su depuración se complica debido a los errores que surgen. Además el manejo de librerías es más complicado que otros lenguajes como Java o .Net y su curva de aprendizaje muy alta. Puede consultarse más información en esta [Página de referencia de C++](#).

Conclusión

Los lenguajes de alto nivel se desarrollaron con el objetivo de ser más accesibles y entendibles por la mayoría de programadores, de manera que los programadores pudieran concentrarse más en resolver la tarea o los problemas y no en el lenguaje que la máquina tenía que entender.

C++ surge de fusionar dos ideas: la eficiencia del lenguaje C para poder acceder al hardware al ejecutar tareas que realmente demandaban recursos de memoria; y las ideas de abstracción que representan los nuevos conceptos de clases y objetos.

El lenguaje C++ presenta grandes herramientas de desarrollo para los programadores como las funciones, bibliotecas, clases y los objetos. De manera que el programador se ocupa de utilizar dichas herramientas para resolver un problema específico.

El lenguaje C++ posee una serie de características que lo hacen distinto del lenguaje C. Aunque es posible verlo como una simple extensión del lenguaje C, en realidad implica un cambio en la forma de pensar por parte del programador.

Bibliografía

Bjarne Stroustrup es el creador de C++, cuya referencia oficial se publicó el 14 de octubre de 1985. Veinticinco años después de este hecho, Stroustrup hace algunas interesantes reflexiones.

El autor del lenguaje C++ empezó la búsqueda de algún lenguaje que reemplazara al lenguaje C, inventado en los laboratorios Bell, ahora AT&T Bell Labs, desde 1979, en donde él y sus colegas en el departamento de investigación, tenían permiso de experimentar con nuevas maneras de construir software.

C++ se convirtió en el lenguaje de programación más popular jamás creado. Se diseñó como un lenguaje de “propósito general”: podría correr en prácticamente cualquier plataforma y en casi todas partes, especialmente en videojuegos y sistemas embebidos.

Después de publicar su libro C++ Programming Language, Stroustrup continuó activo en la comunidad de C++, trabajando en el diseño del lenguaje, escribiendo compiladores y ayudando a hacer del lenguaje un estándar ISO en 1998. Actualmente es un profesor distinguido y poseedor de un puesto en el Colegio de Ingeniería en Ciencias de la Computación en la Universidad A&M de Texas.

Para conmemorar el aniversario de plata de la publicación de este texto definitivo sobre C++, la revista Wired le pidió a Stroustrup sus reflexiones en los últimos 25 años, incluyendo el legado de C++ y el impacto que ha tenido en las aplicaciones actuales del cómputo. Stroustrup nos dijo también qué música escucha mientras programa -una mezcla de clásico y Clapton. [Las respuestas las dio el afamado autor por correo electrónico].

Pregunta: ¿Para quién creó C++? ¿Para usted, o para una audiencia más amplia en mente?

Bjarne Stroustrup: Se escribió pensando en mí y en mis colegas. Teníamos una amplia variedad de problemas y C parecía ser una herramienta innecesariamente primitiva para hacer el tipo de diseño e implementación que necesitábamos y que podríamos hacer con computadoras más rápidas y más memoria. Por otra parte, las máquinas eran todavía ridículamente pequeñas para los estándares del 2010. Un MHz y un MegaByte eran un lujo. Así, el plan fue minimizar los cuellos de botella en lo que era lógicamente lo más importante.

En ese tiempo, AT&T Bell Labs simplemente era el lugar más emocionante del mundo para un ingeniero o para un científico de computadoras con una inclinación práctica. No había fin a la variedad de retos interesantes y mucha gente interesante colaboró en ellos.

Pregunta: ¿En qué momento se dio cuenta que su creación sería algo fenomenal?

Stroustrup: Esto llegó gradualmente. Estaba muy ocupado trabajando para reflejarlo filosóficamente, pero desde el día uno, lo que yo buscaba era la generalidad, y yo sabía que solamente con ideas articuladas podría sobrevivir. Yo sabía que la Programación Orientada a Objetos (POO) era el siguiente gran paradigma, pero estaba ya muy cerca del proceso de apreciar realmente lo que estaba haciendo.

Yo sólo quería hacer que las técnicas de programación se basaran en tipos estáticos seguros y las abstracciones disponibles, enseñables y aplicables a la programación de sistemas. Además de lo que se refiere al diseño del lenguaje y a la implementación. Ayudé en docena de proyectos para sentir las diferentes áreas de los problemas.

de 1979 a 1991, la comunidad de usuarios de C++ se duplicó cada 7.5 meses. Esto me dejó poco tiempo para una callada reflexión. Debe hacerse notar que fue un tiempo de fermentación de la comunidad de programación: Ada, Smalltalk, Eiffel, Objective C, CLOS (Common Lisp), y otros lenguajes compitieron por el gusto de los programadores. En ese contexto, C++ fue el único 'lenguaje avanzado' sin una organización de mercadeo.

Por favor, tome nota que mi camino a la fama no es por haber inventado la POO. No hice eso. Ese honor pertenece a los diseñadores de Simula: Ole-Johan Dahl y Kristen Nygaard, pero mi contribución fue mayormente en ponerlo a consideración de un amplio auditorio. También, mi idea de C++ fue siempre ir más allá de las definiciones apretadas, angostas, de la POO. De hecho incluso mi primer "paper" presentaba un ejemplo de programación genérica (el cual estaba todo mal). Muchas veces es más importante tener el problema correcto que la mejor solución.

Pregunta: En su visión, ¿cuál es el impacto más significativo que C++ ha tenido en la cultura en general, no los programadores, sino en los seres humanos de todos los días?

Stroustrup: La última medida del éxito de cualquier lenguaje de programación debe ser el impacto de sus aplicaciones.

La medida no es la belleza de una instrucción individual o qué tan elegante puede hacerse un ejemplo que pueda caber en el formato académico de dos columnas de un "paper". Así, para mí, la mayor satisfacción viene de las aplicaciones que son interesantes y además, son retos, las cuales no podrían haberse escrito sin C++, o posiblemente se hubiesen tardado años en aparecer por la falta de un lenguaje capaz de hacer aplicaciones que demanda el mundo real: El carrito marciano Mars Rover, el proyecto del genoma humano para encontrar cadenas de ADN, Google, Amazon, los sistemas de reservación de vuelos (Amadeus), análisis de código (Coverity), animación (Maya), autos, aviones, Photoshop, sistemas de telecomunicación, Videojuegos como Doom, Warcraft, Age of Empires, Halo. Túneles de viento, exploración de petróleo, la mayoría del software de Microsoft y mucho de las máquinas virtuales de Java en Apple, Thunderbird y Firefox, MySQL, mucho software financiero, Openoffice, etc.

Cuando está bien hecho, el software es invisible. Hay muchos pequeños procesadores en mi pequeña cámara fotográfica, pero no pienso en mi cámara como una computadora. Hay docenas de computadoras en un auto moderno, pero aún pienso en él como un auto.

Las personas ven las aplicaciones, pero primero yo debo tratar con los fundamentos y la infraestructura que soporta estas aplicaciones. Sospecho que si hubiese tenido algo de sentido comercial, cada computadora y casi cualquier gadget habría tenido una calcomanía que dijera: "C++ Inside".

En general, no pienso que la gente aprecia lo mucho que depende del software. Dada la complejidad de la sociedad moderna y de los sistemas de transportación modernos, es justo decir que la mayoría de la gente podría morir de hambre si no tuviese software. Así como sin electricidad, calor y teléfonos.

Pregunta: ¿Cuánto tiempo pasa actualmente haciendo programación?

Stroustrup: Muy poco, por supuesto. Hay días en que no escribo una sola línea de código y la mayoría de los días escribo pequeños fragmentos de código y programas experimentales. Paso más tiempo pensando sobre programación, tratando de enseñar sobre programación, tratando de escribir sobre programación y tratando de pensar en las maneras fundamentalmente mejores de desarrollar software.

No hallo esto sencillo. Escribir código es más simple y sencillo, pero por sí mismo, es típicamente tan estéril y poco importante como hacer crucigramas.

Sin embargo, hay un par de días en los cuales no hago nada de código y yo nunca doy una plática sin algo de código al menos.

Pregunta: ¿Qué hardware y software usa actualmente?

Stroustrup: Tengo una pequeña laptop (que pesa menos de 3 libras) con un montón de compiladores de C++. Es una máquina con Windows. Mi máquina de escritorio es un sistema Linux con más compiladores y herramientas para el desarrollo de software.

Pregunta: Con respecto a los nuevos sistemas, marcos de trabajo, protocolos y aplicaciones, ¿qué es lo que admira?

Stroustrup: Yo solamente noto que considero que un lenguaje de programación, una herramienta para programar, como la única y la mejor herramienta para todos y para todo problema, es una idea infantil. Si alguien clama que tiene el lenguaje perfecto o es un tonto, o un vendedor, o ambas cosas.

Pregunta: La mayoría de los programadores son muy especiales en lo que respecta a la música que escuchan mientras programan. ¿Qué música escucha usted?

Stroustrup: La quinta de Tchaikovsky, El Anillo sin Palabras, de Wagner, Las suites de Grieg, Sibelius, el Inextinguible de Nielsen, varios conciertos de Mozart, las Dixie Chicks, los Beatles, Abbey Road, el Mesías de Handel, Water Music, Eric Clapton, La quinta y séptima sinfonía de Beethoven. Tengo que ver en mi laptop que es lo último que he escuchado.

Pregunta: ¿De qué modos los dispositivos ultraportátiles con pantallas táctiles -el iPhone, los teléfonos Android, el iPad- han cambiado la manera en cómo las computadoras están en nuestras vidas?

Stroustrup: Sospecho que son tan invasivas en la medida que ayudan. Es bueno obtener indicaciones sobre calles y críticas sobre restaurantes en la palma de la mano, pero estar siempre conectado, siempre accesible o interrumpible, y con la música sonando todo el

tiempo, interfiere con el pensamiento. Necesito tiempo para una reflexión callada, por lo que probablemente yo no aprecie todos estos dispositivos como mucha otra gente.

Pregunta: ¿Qué le parecen las tiendas de apps que sirven a estas plataformas de hardware? ¿Qué impacto están teniendo en los usuarios, programadores, creatividad, el comercio del software?

Stroustrup: No lo sé. Creo que la clave es saber qué es importante al final del día: el uso de bibliotecas masivas de rutinas propietarias y de medios ambientes de ejecución o las oportunidades para aplicaciones innovadoras ofrecidas por la ubicua presencia de los dispositivos. Estos nuevos canales abren nuevas brechas a la imaginación.

Pregunta: ¿Lee libros electrónicos?

Stroustrup: Con muy poca frecuencia y la mayoría es literatura muy 'light'. Encuentro las pantallas muy chicas para los trabajos técnicos y las facilidades para hacer anotaciones y escribir sobre los documentos electrónicos son todavía muy débiles.

Pregunta: ¿Algún consejo para los programadores jóvenes?

Stroustrup: Sospecho que dar consejos es más fácil comparado con hacer caso a ellos. Conozca sus elementos fundamentales (algoritmos, estructura de datos, arquitectura de las computadoras y sistemas), y aprenda varios lenguajes de programación al punto que pueda usarlos como cualquier otro idioma.

Conozca su campo (que no tenga que ver con el cómputo), adecuadamente: matemáticas, biología, historia, óptica, lo que sea. Trate de comunicarse efectivamente verbal y por escrito. Gaste una irracional cantidad de tiempo en tópicos verdaderamente difíciles. Trate de hacer algo que haga la diferencia en nuestro mundo.

Librerías

- Bits STDC ++ H

bits / stdc ++. h no es un archivo de encabezado estándar de la biblioteca GNU C++. Entonces, si intenta compilar su código con algún compilador que no sea GCC, podría fallar; por ejemplo, MSVC no tiene este encabezado. Usarlo incluiría muchas cosas innecesarias y aumentaría el tiempo de compilación.

- Casting

Un objeto CString contiene datos de cadenas de caracteres. CString hereda el conjunto de métodos y operadores definidos en la plantilla de clase CStringT para trabajar con los datos de cadena. (CString es una typedef que especializa a CStringT para que trabaje con el tipo de datos de caracteres que admite CString).

CString no almacena datos de caracteres internamente como una cadena terminada en un valor nulo de estilo C. En su lugar, CString realiza un seguimiento de la longitud de los datos de caracteres para, así, poder vigilar los datos y el espacio que precisan de forma más segura.

Con todo, CString acepta cadenas de estilo C y ofrece formas de acceder a los datos de caracteres como una cadena de estilo C. Este tema está compuesto por las siguientes secciones, donde se explica cómo usar un objeto CString como si fuera una cadena terminada en un valor nulo de estilo C.

- fstream

Ifstream es una clase que permite crear la instancia fich para leer de un fichero. En este caso, se abre un fichero para lectura llamado "ejemplo. dat"

- iomanip

El método setw() de la biblioteca iomanip en C++ se utiliza para establecer el ancho del campo de la biblioteca ios en función del ancho especificado como parámetro de este método.

Manual de usuario

- Menú principal

```
llima@fedora: ~/Desktop/proyecto_final — /out
llima@fedora proyecto_final]$ ./out
1. Visualizar base de datos
2. Ingresar información de alumno
3. Actualizar información de alumno
4. Ordenar base de información
5. Eliminar alumno
6. Buscar alumno
0. Salir.
```

El sistema funciona en base a códigos de opción, es decir, si usted presiona la tecla 1 en su ordenador, ingresará a la opción “Visualizar base de datos” en el sistema. Como se muestra en pantalla.

```
0. Salir.
```

Alumnos						
Carnet de alumno	Primer Nombre	Primer Apellido	Teléfono	Sexo	Curso	Sección
7690-20-17409	Luis	Lima	42093868	H	Programacion	C
3412-19-13568	Fernanda	Monterroso	34657823	M	Psicologia	A
5467-20-34622	Keyla	León	45673412	M	Artes	B
3593-18-97436	Luisa	Maldonado	48298752	M	Sociologia	A
6534-20-13565	Fredy	Monterroso	94784743	H	Filosofia	C
4564-17-23434	Fernando	Martinez	34124567	H	Biologia	A
3455-20-13245	Josue	Ramirez	38945678	H	Desarrollo	C
8747-17-98726	Francisca	Ramirez	98734025	M	Logica	A
9873-19-34578	Roberto	Juarez	09456745	H	Antropologia	A
9847-20-34897	Juan	Hernandez	84924853	H	Metodologia	B
9873-20-97763	Lissa	León	83720762	M	Anatomia	C
0987-18-08724	Aracely	Jimenez	09385090	M	Estadistica	C
8384-20-28497	Selvin	Monteros	49382840	H	Semiologia	B
8937-19-13786	Michael	Yaguas	83974722	H	Psicometría	B
9479-20-23487	Javier	Ramirez	87346573	H	Teorias	A
9837-19-87278	Luis	Gonzales	03847227	H	Psicologia	C
2749-19-91873	Sara	Perez	90874513	M	Neurofisiologia	A
9347-20-92736	Maria	Ramirez	98453265	M	Forense	C
9389-17-82380	Ana	Gonzales	93082399	M	Informatica	B
3984-20-20841	Saraí	Vega	09348225	M	Administración	A
9398-18-10389	Andrea	Ramirez	98382344	M	Análisis	A
9237-20-29384	Cristina	Perez	84727497	M	Psicometría	C
9824-19-39482	Carolina	Vasquez	87342112	M	Semiologia	A
8298-18-73972	Sabrina	Linares	82847642	M	Forense	C
4298-19-03842	Irene	Perez	98457613	M	Neurofisiologia	C
9372-20-46257	Juana	Velasquez	83729472	M	Teorias	A
9238-20-83981	Katherine	Cruz	34747592	M	Informatica	A
9829-18-19327	Felipe	Juarez	98472478	H	Logica	B
7492-20-29477	Javier	Lopez	84729872	H	Psicometría	B

- Añadir un alumno

Para ingresar a esta opción, deberá presionar 2 en el menú principal. Al ingresar a esta opción el sistema comenzará a realizar las peticiones de información. Deberá agregar los campos que se le solicitan.

Si realizo todo correctamente, el sistema le mostrará un mensaje de respuesta.

```
Carnet de alumno: 7690-20-89674
Primer Nombre: Peter
Primer Apellido: Lopez
Número de teléfono: 44667890
Sexo: H
Curso: Telecomunicaciones
Sección: F
;Alumno registrado correctamente!
```

- Actualización de información

Opción 3 en el menú principal. El sistema le solicitará referencias del alumno que desee actualizar. Podrá indicar cualquier dato que se tenga del mismo, como por ejemplo:

- Numero de telefono
- Nombre
- Carnet
- Apellido
- etc.

En este caso, se usará el nombre del alumno Luis.

```

3
Escriba referencia de alumno: Luis
ID - 0 7690-20-17409 Luis Lima 42093868 H Programacion C
ID - 3 3593-18-97436 Luisa Maldonado 48298752 M Sociologia A
ID - 15 9837-19-87278 Luis Gonzales 03847227 H Psicologia C
Ingrese el ID del alumno correspondiente: █

```

El sistema mostrará los registros obtenidos con referencia de un ID, el cual deberá seleccionar el que se requiera actualizar.

```

ID - 15 9837-19-87278 Luis Gonzales 03847227 H Psicologia C
Ingrese el ID del alumno correspondiente: 15
9837-19-87278 Luis Gonzales 03847227 H Psicologia C9837-19-87278 Luis Gonzales 03847227 H Psicologia C

Actualizar detalles de alumno
1. Carnet de alumno
2. Primer Nombre
3. Primer Apellido
4. Teléfono
5. Sexo
6. Curso
7. Sección
8. Guardar y salir
9. Salir

```

Posteriormente, deberá indicar el campo a actualizar.

```

Actualizar detalles de alumno
1. Carnet de alumno
2. Primer Nombre
3. Primer Apellido
4. Teléfono
5. Sexo
6. Curso
7. Sección
8. Guardar y salir
9. Salir
6
Curso : Algebra

```

9479-20-23487	Javier	Ramirez	87346573	H	Teorias	A	
9837-19-87278	Luis	Gonzales	03847227	H	Algebra	C	
2749-19-91873	Sara	Perez	98874513	M	Neurofisiologia	A	
9347-20-92736	Maria	Ramirez	98453265	M	Forense	C	
9389-17-82380	Ana	Gonzales	93082399	M	Informatica	B	
3984-20-20841	Saraí	Vega	09348225	M	Administración	A	
9398-18-10389	Andrea	Ramirez	98382344	M	Analisis	A	
2137-20-29384	Cristina	Perez	84727497	M	Psicometria	C	
9824-19-39482	Carolina	Vasquez	87342112	M	Semiologia	A	
8298-18-73972	Sabrina	Linares	82847642	M	Forense	C	
4298-19-03842	Irene	Perez	98457613	M	Neurofisiologia	C	
9372-20-46257	Juana	Velasquez	83729472	M	Teorias	A	
9238-20-83981	Katherine	Cruz	34747592	H	Informatica	A	
9829-18-19327	Felipe	Juarez	98472478	H	Logica	B	
7492-20-29477	Javier	Lopez	84729872	H	Psicometria	B	

Efectuados todos los pasos, verá el resultado en la visualización de datos.

- Eliminar alumno

Este paso consiste inicialmente en los pasos de actualización.

```
5
Buscar alumno para eliminar: Peter
ID - 30 7690-20-89674 Peter Lopez 44667890 H Telecomunicaciones F
Ingrese el ID del alumno correspondiente:
```

El sistema preguntará solicitará su confirmación para realizar esta acción.

```
5
Buscar alumno para eliminar: Peter
ID - 30 7690-20-89674 Peter Lopez 44667890 H Telecomunicaciones F
Ingrese el ID del alumno correspondiente: 30
¿Está seguro? ¡Esto no se puede deshacer!
Ingrese 1. Para eliminar.
Ingrese 0. Para cancelar.
```

```
5
Buscar alumno para eliminar: Peter
ID - 30 7690-20-89674 Peter Lopez 44667890 H Telecomunicaciones F
Ingrese el ID del alumno correspondiente: 30
¿Está seguro? ¡Esto no se puede deshacer!
Ingrese 1. Para eliminar.
Ingrese 0. Para cancelar.
1
----->
|                ¡Alumno eliminado correctamente!                |
----->

1. Visualizar base de datos
2. Ingresar información de alumno
3. Actualizar información de alumno
4. Ordenar base de información
5. Eliminar alumno
6. Buscar alumno
0. Salir.
```

Código fuente

```
#include <bits/stdc++.h>
#include <stdio.h>
#include <cstring>
#include <fstream>
#include <iomanip>
#include <iostream>
using namespace std;

void createContact();
void displayAllContacts();
void searchContact();
void deleteContact();
void sortAllContact();
void actualSorting();
void updateContact();
void delete_line(const char*, int);
//uso de en caso de, para la selección de opciones multiples, realiza el
//llamado a la funcion correspondiente
int main() {
    int exit = 1;
    while (exit) {
        cout << "\n";
        cout << " 1. Visualizar base de datos \n 2. Ingresar información
de alumno \n 3. Actualizar información de alumno"
        "\n 4. Ordenar base de información \n 5. Eliminar alumno
\n 6. Buscar alumno"
        "\n 0. Salir.\n";
        cout << "\n";
        cin >> exit;
        switch (exit) {
            case 1:
                displayAllContacts();
                break;
            case 2:
                createContact();
                break;
            case 3:
                updateContact();

                break;
```

```

        case 4:
            sortAllContact();
            break;
        case 5:
            deleteContact();
            break;
        case 6:
            searchContact();
            break;
        case 0:
            cout << "¡Gracias por usar este sistema!\n";
            break;
        default:
            cout << "Enter Valid Input";
            break;
    }
}

return 0;
}

//Ordenamiento de información
void sortAllContact() {
    ofstream fout;
    string line;
    remove("temp.txt");
    fout.open("temp.txt", ios::app);
    fstream fin("data_base.txt");
    while (getline(fin, line)) {
        fout << line << endl;
    }
    fout.close();
    actualSorting();
}

void actualSorting() {
    ofstream fout;
    string myText;
    char v = 'a';
    // cout << int(v);
    // cout << endl << char(int(v));
    int letter = 97;
    remove("data_base.txt");
    fout.open("data_base.txt", ios::app);
SORTING:
    ifstream fin("temp.txt");

```

```

while (getline(fin, myText)) {
    char myChar = myText[0];
    if (isupper(myChar)) {
        myChar = tolower(myChar);
    }
    if (int(myChar) == letter) {
        fout << myText << endl;
    }
}
fin.seekg(0);
fin.close();
if (letter != 124) {
    letter++;
    goto SORTING;
}
cout << "\t\t;Registro de alumnos ordenado con éxito!\n";
fout.close();
}
//Actualización de alumnos
void updateContact() {
    int lineNumber = 0;
    string myText, search, smallString;
    ifstream MyReadFile("data_base.txt");
    cout << "Escriba referencia de alumno: ";
    cin >> search;
    while (getline(MyReadFile, myText)) {

        smallString = myText;
        transform(smallString.begin(), smallString.end(),
smallString.begin(),
                ::toupper);
        transform(search.begin(), search.end(), search.begin(),
::toupper);
        if (smallString.find(search) != string::npos) {
            cout << "ID - " << lineNumber << " " << myText << endl;
        }
        lineNumber++;
    }
    MyReadFile.close();

    int id, choice;
    cout << "Ingrese el ID del alumno correspondiente: ";
    cin >> id;

```



```

ifstream fin("data_base.txt");
ofstream fout;
remove("temp.txt");
fout.open("temp.txt", ios::app);
string line;
int lineCount = 0;
while (getline(fin, line)) {
    if (lineCount == id) {
        cout << line;
        string fname, mname, lname, number, sex, curs, sec;
        stringstream sss(line);
        string word;
        int index = 0;
        while (sss >> word) {
            switch (index) {
                case 0:
                    fname = word;
                    break;
                case 1:
                    mname = word;
                    break;
                case 2:
                    lname = word;
                    break;
                case 3:
                    number = word;
                    break;
                case 4:
                    sex = word;
                    break;
                case 5:
                    curs = word;
                    break;
                case 6:
                    sec = word;
                    break;
                default:
                    break;
            }
            index++;
        }
    }
}

```

```

        cout << fname << "\t" << mname << "\t" << lname << "\t" <<
number << "\t" << sex << "\t" << curs << "\t" << sec << "\t"
        << endl;
    cout << endl;
    int myUpdate;
    cout << "\t\tActualizar detalles de alumno\n 1. Carnet de
alumno\n 2. Primer Nombre"
        "\n 3. Primer Apellido "
        "\n 4. Teléfono"
        "\n 5. Sexo"
        "\n 6. Curso"
        "\n 7. Sección"
        "\n 8. Guardar y salir\n 9. Salir\n";
    cin >> myUpdate;
    switch (myUpdate) {
        case 1:
            cout << "Carnet de alumno : ";
            cin >> fname;
            break;
        case 2:
            cout << "Primer Nombre : ";
            cin >> mname;
            break;
        case 3:
            cout << "Primer Apellido : ";
            cin >> lname;
            break;
        case 4:
            cout << "Número de teléfono : ";
            cin >> number;
            break;
        case 5:
            cout << "Sexo : ";
            cin >> sex;
            break;
        case 6:
            cout << "Curso : ";
            cin >> curs;
            break;
        case 7:
            cout << "Sección : ";
            cin >> sec;
            break;
    }
}

```

```

        case 8:
            break;
        case 9:
            break;
        default:
            break;
    }
    fout << fname << " " << mname << " " << lname << " " <<
number << sex << " " << curs << " " << sec
    << "\n";
} else {
    fout << line << endl;
}
lineCount++;
}
fout.close();
remove("data_base.txt");
rename("temp.txt", "data_base.txt");
}
//creación de funcion, creación de contactos.
void createContact() {
    //declaración de variables para almacenamiento en txt
    char firstName[30], middleName[30], lastName[30], sex_student[30],
curso_student[30], section_student[30];
    char number[30];
fname:
    cout << "Carnet de alumno: ";
    cin >> firstName;
mname:
    cout << "Primer Nombre: ";
    cin >> middleName;
lname:
    cout << "Primer Apellido: ";
    cin >> lastName;
mo:
    cout << "Número de teléfono: ";
    cin >> number;
sex:
    cout << "Sexo: ";
    cin >> sex_student;
curs:
    cout << "Curso: ";
    cin >> curso_student;

```

```

sec:
    cout << "Sección: ";
    cin >> section_student;

    if (strlen(number) != 8) {
        cout << "xxx Ingrese un número de telefono válido de 8 dígitos
xxx \n ";
        goto mo;
    }
    ofstream fout;
    string line;
    fout.open("data_base.txt", ios::app);
    fout << firstName << " " << middleName << " " << lastName << " " <<
number << " " << sex_student << " " << curso_student << " " <<
section_student
        << "\n";
    cout << "\t\t;Alumno registrado correctamente!\n";
    fout.close();
}

//ornamentatia de impresión de datos
void displayAllContacts() {
    string myText;
    ifstream MyReadFile("data_base.txt");
    cout << "+" << setfill('-') << setw(140) << "+" << endl;
    cout << "|" << setfill(' ') << setw(75) << "Alumnos" << setw(65) <<
"|"
        << endl;
    cout << "+" << setfill('-') << setw(80) << "+" << setw(20) << "+"
        << setw(20) << "+" << setw(20) << "+" << endl;
    cout << "|" << setfill(' ')
        << setw(10) << "Carnet de alumno" << setw(4) << "|"
        << setw(15) << "Primer Nombre" << setw(5) << "|"
        << setw(17) << "Primer Apellido" << setw(3) << "|"
        << setw(12) << "Teléfono" << setw(9) << "|"
        << setw(10) << "Sexo" << setw(10) << "|"
        << setw(10) << "Curso" << setw(10) << "|"
        << setw(11) << "Sección" << setw(10) << "|" << endl;
    cout << "+" << setfill('-') << setw(80) << "+" << setw(20) << "+"
        << setw(20) << "+" << setw(20) << "+" << endl;
    //obtención de data
    while (getline(MyReadFile, myText)) {
        string fname, mname, lname, number, sex, curs, sec;
        stringstream ss(myText);

```

```
string word;
int index = 0;
while (ss >> word) {
    switch (index) {
        case 0:
            fname = word;
            break;
        case 1:
            mname = word;
            break;
        case 2:
            lname = word;
            break;
        case 3:
            number = word;
            break;
        case 4:
            sex = word;
            break;
        case 5:
            curs = word;
            break;
        case 6:
            sec = word;
            break;
        default:
            break;
    }
    index++;
}

int w1 = ((20 - fname.size()) / 2) + fname.size();
int w2 = 20 - w1;
int w3 = ((20 - mname.size()) / 2) + mname.size();
int w4 = 20 - w3;
int w5 = ((20 - lname.size()) / 2) + lname.size();
int w6 = 20 - w5;
int w7 = ((20 - number.size()) / 2) + number.size();
int w8 = 20 - w7;
int w9 = ((20 - sex.size()) / 2) + sex.size();
int w10 = 20 - w9;
int w11 = ((20 - curs.size()) / 2) + curs.size();
int w12 = 20 - w11;
int w13 = ((20 - sec.size()) / 2) + sec.size();
```

```

    int w14 = 20 - w13;
    cout << "|" << setfill(' ')
          << setw(w1) << fname << setw(w2) << "|"
          << setw(w3) << mname << setw(w4) << "|"
          << setw(w5) << lname << setw(w6) << "|"
          << setw(w7) << number << setw(w8) << "|"
          << setw(w9) << sex << setw(w10) << "|"
          << setw(w11) << curs << setw(w12) << "|"
          << setw(w13) << sec << setw(w14) << "|"
          << endl;

}

cout << "+" << setfill('-') << setw(80) << "+" << setw(20) << "+"
      << setw(20) << "+" << setw(20) << "+" << endl;
MyReadFile.close();
}

//busqueda de un alumno
void searchContact() {
    string myText, search, smallString;
    ifstream MyReadFile("data_base.txt");
    cout << "Ingrese carnet de alumno para buscar: ";
    cin >> search;
    cout << "\t\tRegistro(s) encontrado(s) :\n";
    while (getline(MyReadFile, myText)) {
        smallString = myText;
        transform(smallString.begin(), smallString.end(),
smallString.begin(),
::toupper);
        transform(search.begin(), search.end(), search.begin(),
::toupper);
        if (smallString.find(search) != string::npos) {
            int w1 = ((60 - myText.size()) / 2) + myText.size();
            int w2 = 60 - w1;
            cout << "+" << setfill('-') << setw(60) << "+" << endl;
            cout << "|" << setfill(' ') << setw(w1) << myText <<
setw(w2) << "|"
                << endl;
            cout << "+" << setfill('-') << setw(60) << "+" << endl;
        }
    }
    MyReadFile.close();
}

void deleteContact() {

```

```

int lineNumber = 0;

string myText, search, smallString;
ifstream MyReadFile("data_base.txt");
cout << "Buscar alumno para eliminar: ";
cin >> search;

while (getline(MyReadFile, myText)) {

    smallString = myText;
    transform(smallString.begin(), smallString.end(),
smallString.begin(),
                ::toupper);
    transform(search.begin(), search.end(), search.begin(),
::toupper);
    if (smallString.find(search) != string::npos) {
        cout << "ID - " << lineNumber << " " << myText << endl;
    }
    lineNumber++;
}

MyReadFile.close();
int id, choice;
cout << "Ingrese el ID del alumno correspondiente: ";
cin >> id;
cout << "¿Está seguro? ¡Esto no se puede deshacer! \nIngrese 1. Para
eliminar. \nIngrese 0. Para cancelar. \n";
cin >> choice;
if (!choice == 0) {
    delete_line("data_base.txt", id);
    cout << "+" << setfill('-') << setw(60) << "+" << endl;
    cout << "|" << setfill(' ') << setw(46)
        << "¡Alumno eliminado correctamente!" << setw(14) << "|" <<
endl;
    cout << "+" << setfill('-') << setw(60) << "+" << endl;
}
}

void delete_line(const char* file_name, int n) {
    ifstream is(file_name);
    ofstream ofs;
    ofs.open("temp.txt", ofstream::out);
    char c;

```

```
int line_no = 0;
while (is.get(c)) {
    if (c == '\\n') line_no++;

    if (line_no != n) ofs << c;
}

ofs.close();
is.close();
remove(file_name);
rename("temp.txt", file_name);
}
```