



Facultad de Ingeniería Ingeniería y de Sistemas

Algoritmia y Estructura de Datos: Archivos

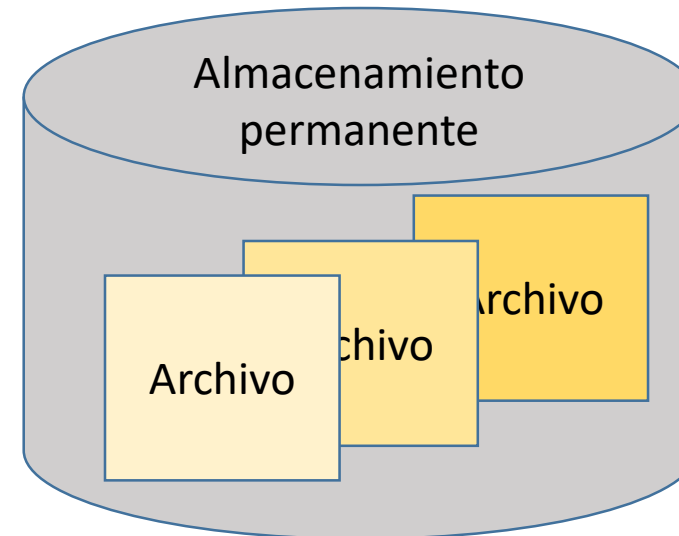
¿Qué es un archivo?

Definición

Es un elemento para almacenar de manera permanente un conjunto de datos.

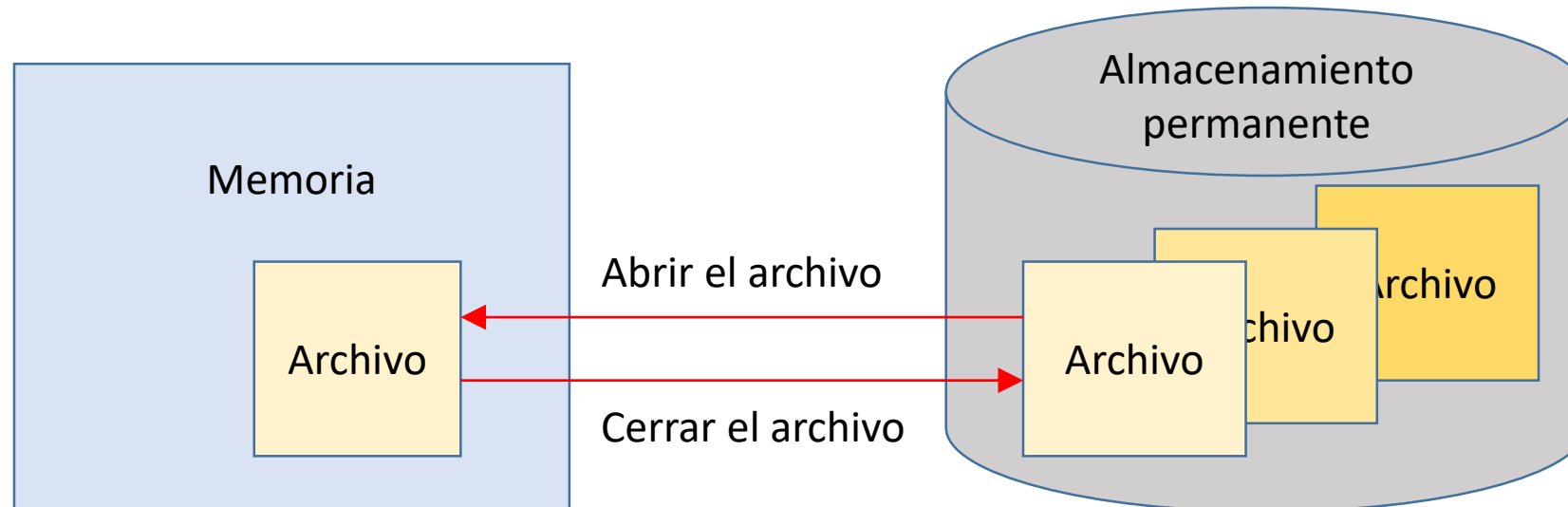
Un archivo requiere un dispositivo de almacenamiento como el disco duro.

Memoria
Solo almacena los datos
mientras esta el programa
en ejecución



Utilización de archivos en C++

Para utilizar un archivo en C++ es necesario que pase del medio de almacenamiento permanente a la memoria



Tipos de archivos

Archivos de texto

Juan Gonzales Av Los Robles 123 Surquillo 978678543
Luisa Rios Jr Las Lilas 342 Barranco 897654234
Mario Merino Av Tupac Amaru 234 Rimac 978654123

- Almacenan los datos como secuencias de caracteres.
- Son secuenciales se leen de inicio a fin

Archivos binarios

Registro

Registro

Registro

- Almacenan bytes de memoria.
- Muy útiles para almacenar registros (structs)
- Se puede acceder a cualquier registro directamente

Declarar una variable archivo de texto para lectura

```
#include <iostream>
```

```
#include <fstream>
```

```
using namespace std;
```

```
main() {
```

```
    ifstream fichero("Agenda.txt");
```

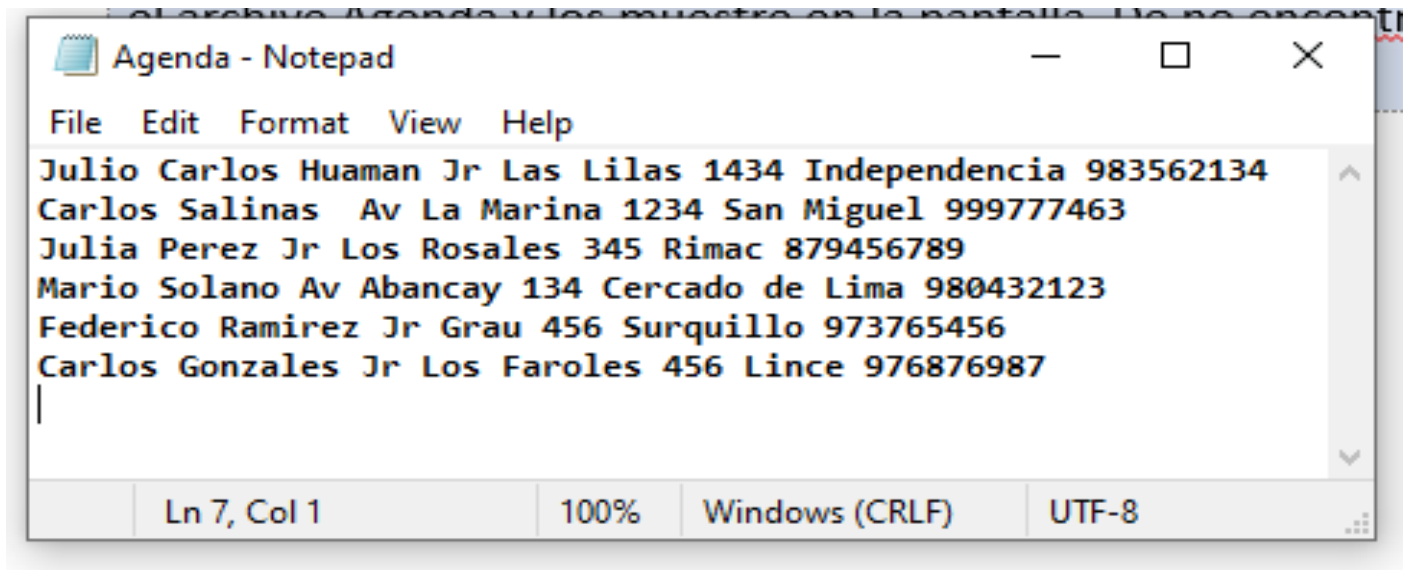
Librería para manejo de archivos

Declaración de variable "fichero" como
archivo para lectura

Lectura de archivo de texto: El problema de la “Agenda”

Se tiene un archivo de texto denominado “Agenda.txt” que contiene en cada línea de texto el nombre, la dirección y el telefono de una persona. Se pide desarrollar un programa que solicite un nombre y luego encuentre los datos de la persona buscada en el archivo Agenda y los muestre en la pantalla. De no encontrar ninguna coincidencia debe mostra el mensaje “No encontrado”.

Ejemplo del archivo
Agenda.txt



```
Agenda - Notepad
File Edit Format View Help
Julio Carlos Huaman Jr Las Lilas 1434 Independencia 983562134
Carlos Salinas Av La Marina 1234 San Miguel 999777463
Julia Perez Jr Los Rosales 345 Rimac 879456789
Mario Solano Av Abancay 134 Cercado de Lima 980432123
Federico Ramirez Jr Grau 456 Surquillo 973765456
Carlos Gonzales Jr Los Faroles 456 Lince 976876987
|
Ln 7, Col 1 100% Windows (CRLF) UTF-8
```

Lectura de archivo de texto: El problema de la "Agenda"

```
1 #include <iostream>
2 #include <fstream>
3 using namespace std;
4 main() {
5     ifstream fichero("Agenda.txt");
6     bool encontrado=false;
7     int lc,ln,i,j;
8     string nombre,cadena, subcadena;
9     cout<<"Ingresar nombre a buscar en la agenda: ";
10    getline(cin,nombre);
11    ln=nombre.length();
12    for(i=0;i<ln;i++)
13        nombre[i]=toupper(nombre[i]);
14    while(!fichero.eof()){
15        getline(fichero,cadena);
16        lc=cadena.length();
17        for(i=0;i<lc;i++)
18            cadena[i]=toupper(cadena[i]);
19        i=0;
20        j=0;
21        subcadena="";
```

Libreria para manejo de archivos

Declaracion y apertura del archive

Lectura de cadena desde archivo

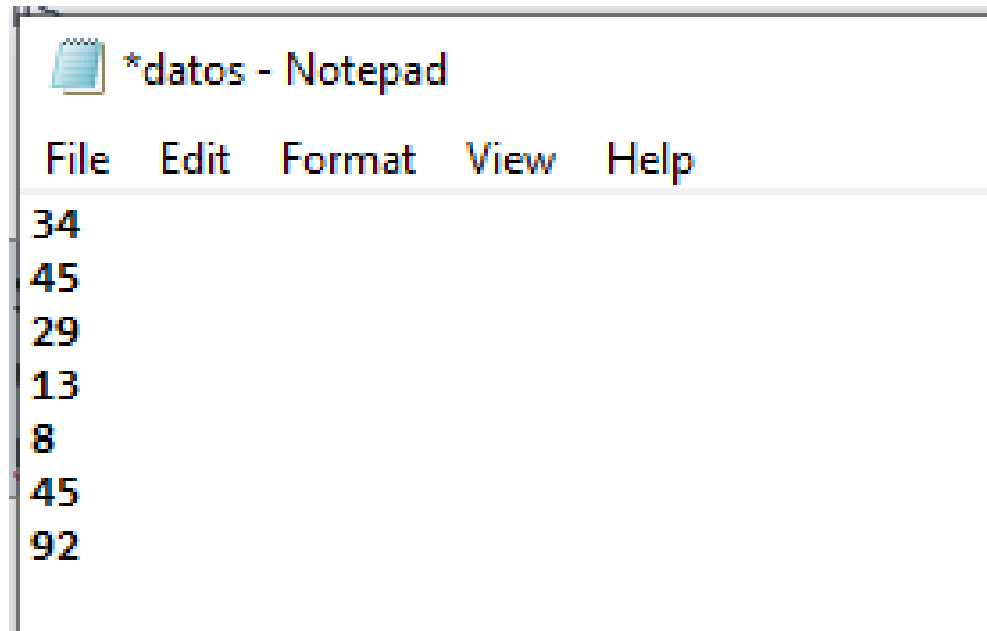
```
22    while (i<lc){
23        if (cadena[i]==nombre[j]){
24            subcadena=subcadena+cadena[i];
25            j++;
26        }
27        else
28            if (subcadena==nombre){
29                encontrado=true;
30                j=0;
31                i=lc;
32                subcadena="";
33                cout<<cadena<<endl;
34            } else {
35                j=0;
36                subcadena="";
37            }
38        i++;
39    }
40 }
41 fichero.close();
42 if (!encontrado)
43     cout<<"No encontrado"<<endl;
44 }
```

Cerrar archivo

Problema con lectura de números desde un archivo de texto

Se tiene un archivo de texto denominado “Datos.txt” que contiene en cada línea un número entero. Se pide leer los datos del archivo y determinar el promedio, el máximo y mínimo valor

Ejemplo del archivo
datos.txt



```
*datos - Notepad
File Edit Format View Help
34
45
29
13
8
45
92
```


Declarar una variable archivo de texto para escritura

```
#include <iostream>
```

```
#include <fstream>
```

```
using namespace std;
```

```
main() {
```

```
    ofstream fichero("Agenda.txt");
```

Librería para manejo de archivos

Declaración de variable “fichero” como
archivo para escritura

Ejemplo de escritura en archivo de texto

Programa que escribe una secuencia de números enteros en un archivo Datos2.txt

```
#include <iostream>
#include <fstream>
using namespace std;
main() {
    ofstream fichero("datos2.txt");
    int num,i,n;
    cout<<"Ingrese el numero de datos a guardar en el archivo: ";
    cin>>n;
    i=0;
    for(i=0;i<n;i++){
        cout<<"Ingrese numero: ";
        cin>>num;
        if (i<n-1)
            fichero<<num<<endl;
        else
            fichero<<num;
    }
    fichero.close();
}
```

Problema para escribir en un archivo de texto

Escribir un programa que genere un archivo "Agenda.txt". El programa debe solicitar un nombre, una dirección y un teléfono y luego guardarlo en el archivo. El ingreso de datos debe terminar cuando se ingresa enter en lugar del nombre.



Solución

```
1  #include<iostream>
2  #include<fstream>
3  using namespace std;
4  main(){
5      ofstream fichero("Agenda.txt");
6      string nombre,direccion, telefono;
7      bool sigue=true;
8      while(sigue){
9          cout<<"Ingrese nombre: "; getline(cin,nombre);
10         if(nombre!=""){
11             cout<<"Ingrese direccion: "; getline(cin,direccion);
12             cout<<"Ingrese telefono: "; getline(cin,telefono);
13             fichero<<nombre<<" "<<direccion<<" "<<telefono<<endl;
14         }
15         else
16             sigue=false;
17     }
18     fichero.close();
19 }
```

Escritura en un archivo binario

Programa para escribir una estructura en un archivo binario. Ejemplo crear una estructura para leer los datos de un registro de la agenda y luego guardar la estructura en un archivo.

```
1  #include <iostream>
2  #include <fstream>
3  using namespace std;
4  int main(){           // ejemplo de archivo binario - guardar registros de agenda
5      char resp;
6      struct reg_agenda{
7          char nombre[30], direccion[50], telefono[10];
8      } ragenda;
9      ofstream archi("Agenda.dat",ios::out|ios::binary);
10     do {
11         cout<<"Ingrese nombre: "; gets(ragenda.nombre);
12         cout<<"Ingrese direccion: "; gets(ragenda.direccion);
13         cout<<"Ingrese telefono: "; gets(ragenda.telefono);
14         archi.write(reinterpret_cast<char *>(&ragenda),sizeof(reg_agenda));
15         cout<<"Agrega otro registro (S/N)? ";
16         cin>>resp; resp = toupper(resp);
17         getchar();
18     }while (resp=='S');
19     archi.close();
20 }
```

Libreria para manejo de archivos

Abrir para escritura como binario

Escribir registro en el archivo

Programa para leer los registros de un archivo binario. Ejemplo leer de un archivo binario los registros de la agenda.

```
1 #include <iostream>
2 #include <fstream>
3 using namespace std;
4 int main() { // ejemplo de archivo binario - Lee registros de agenda.dat
5     struct reg_agenda{
6         char nombre[30], direccion[50], telefono[10],
7     } agenda;
8     ifstream archi("Agenda.dat", ios::in | ios::binary);
9     while (!archi.eof()) {
10         archi.read(reinterpret_cast<char *>(&agenda), sizeof(reg_agenda));
11         cout << "Nombre: " << agenda.nombre << endl;
12         cout << "Direccion: " << agenda.direccion << endl;
13         cout << "Telefono: " << agenda.telefono << endl;
14     }
15     archi.close();
16 }
```

Libreria para manejo de archivos

Abrir para lectura como binario

Leer registro del archivo

Abrir un archivo de texto para agregar datos

Programa para agregar datos a un archive, sea binario o de texto. Ejemplo agregar datos al archive Agenda.txt o Agenda.dat

```
#include <iostream>
#include <fstream>
using namespace std;
main() {
    ofstream archi("Agenda.txt", ios::app);
    ...
    ...
    ...
    archi<<nombre<<" "<<direccion<<" "<<telefono<<endl;
    ...
    ...
}
```

Abrir para agregar

Escribe al final del archivo

Programa para agregar datos a un archive, sea binario o de texto. Ejemplo agregar datos al archive Agenda.txt o Agenda.dat

```
#include <iostream>
#include <fstream>
using namespace std;
main() {
    struct reg_agenda{
        char nombre[30], direccion[50],
            telefono[10];
    }ragenda;
    ofstream archi("Agenda.dat", ios::app|ios::binary);
    ...
    ...
    archi.write(reinterpret_cast<char*>(&ragenda), sizeof(reg_agenda));
    ...
    ...
```

Abrir para agregar y como binario

Escribe el registro al final del archivo

Problema que requiere volver al inicio de un archivo de texto

Escribir un programa que permita ubicar los datos de una persona en un archivo de texto Agenda.txt. Debe pedir una cadena que representa a algún dato de la persona (Nombre, dirección o teléfono) y buscar en el archivo los reglones de datos que contengan la cadena ingresada y mostrarlos en pantalla. De no encontrar coincidencia debe mostrar "Datos no encontrados". Luego debe preguntar si desea ubicar a otra persona, de ser así debe volver a preguntar por la cadena a buscar y volver a explorar el archivo. De lo contrario terminar.

Problema que permite agregar o modificar los datos de una persona en un archivo binario de registros

Escribir un programa que sobre un archivo binario de registros (Agenda.dat), que almacena los registros de personas, nombre (cadena de 30), dirección (cadena de 50) y teléfono (cadena de 10), permita agregar o ubicar a una persona. En caso de ubicar a una persona debe ofrecer la opción de modificar cualquier dato de la persona.