

Machete de archivos

Funciones útiles

Archivo.close();Cerrar un archivo: Se encarga de vaciar los buffers que usa el archivo y cerrar el archivo.

Archivo .is_open(); retorna true si se abrió el archivo

Solo para archivos de texto

Función eof(): para conocer si se ha llegado al final del archivo al leerlo.

Archivos de texto

Acceso secuencial, mas pesados que los binarios, se pueden leer y modificar fácilmente mejor portabilidad que los binarios

Leer archivo de texto

El archivo a leer tiene este formato

Institución

Cod monto fecha

Cod monto fecha

.

.

Cod monto fecha

eof

```
#Include fstream
```

```
ifstream archi("dirección de archivo.txt");
```

Creo variables para guardar los datos a leer

```
int cod;
```

```
float monto;
```

```
char fecha[10];
```

```
char insti[10];
```

//comprobamos si se abrio

```
if( !Archi.is_open()){
```

```
cout<<"no se pudo abrir"<<endl;
```

```
}
```

leemos una vez;

//los q son char o string se deben leer como se leeria una variable de tipo char o string ingresada por consola

```
archi.getline(insti,11);
```

```
archi>>cod; //los demas tipos con >>;
```

```
archi>>monto;
```

```
archi.getline(fecha,11);
```

```
while(!archi.eof()){
```

//itero mientras sea distinto del final de archivo

//Hago algo con los datos q lei

//Y al final vuelvo a leer

```
archi>>cod;
```

```
archi>>monto;
```

```
archi.getline(fecha,11);
```

```
}
```

```
Archi.close() ;//cierro archivo
```

Escribir en un archivo de texto

//ios out crea el archivo si el archivo no existe

//ios::app agrega al final del archivo sin esto cada vez q escriba en el archivo lo estaría borrando y volviendo a escribir la primer linea

```
Ofstream archi("dirección de archivo.txt",ios::app|ios::out);
```

```
Char institución[10];
```

```
Char fecha[10];
```

```
Int cod;
```

```
Float monto;
```

Supongamos q ya estan inicializadas las variables con algun valor

```
Char c[3]="\n";
```

```
Strcat(institución,c); //le concateno un enter para que quede igual al de arriba
```

```
Strcat(fecha,c); //le concateno un enter para que quede igual al de arriba
```

```
Archi<<institución
```

```
Archi<<cod<<monto<<fecha;//esta línea si fueran mas datos estaqia enun while o for
```

Después q haya escrito todos los datos q quiera cierro archivo

```
Archi .close();
```

Archivos binarios

1. Se puede acceder directamente
2. es mas seguro q el de texto ya q no es legible al abrirlo normalmente
3. es mas liviano q el de texto
4. es menos portable q el de texto ya q algunos microprocesadores leen los archivos binarios de forma distinta (binario en mac != binario en windows)

LEER y ESCRIBIR UN ARCHIVO BINARIO

Necesito tener una estructura ya sea un struct o una clase no puede contener punteros en los datos que queramos guardar en el archivo;

Persona.h

Class persona{

Private:

char nom[10];

char fecha_nac[10];

float sueldo;

public

persona(char *n_nom,char *fecha,float s){

strcpy(nom,n_nom);

strcpy(fecha_nac,fecha);

sueldo=s;

}

Void mostrar(){

Cout<<"nombre: "<<nom<<endl

Cout<<"fecha de cumpleaños: "<<fecha_nac<<endl

Cout<<"sueldo": "<<sueldo<<endl

}

}

Leer archivos binarios

```
ifstream archi ("ruta.bin",ios::binary);
If(!archi.is_open()){
Cout<<"no se pudo abrir el archivo"<<endl;
}else{
// para leer un archive necesito saber dos cosas el tamaño del archive
// y la cantidad de elementos q tiene para poder iterarlo esa cantidad de
veces

Archi.seekg(0,ios::end);//voy al final del archivo
Int tamano=archi.tellg(); //devuelvo la pos(el tamaño);
Int cant_personas=tamano/sizeof(persona);
//creo una persona vacia donde copiamos lo q leemos
Persona p;

//importante antes de leer volver al inicio del archivo
Archi.seekg(0,ios::beg);

//ahora si leo
For(int i=0;i<cant_personas;i++){
    Archi.read((char *)&p,sizeof(persona));
    Cout<<p.mostrar();

    //aca solo lo muestro pero se podria agregar p a un vector u otro
    contenedor
}
Archi.close();
```

Escribir archivo binario

```
ofstream archi ("ruta.bin",ios::app|ios::out|ios::binary));
```

```
if(!archi.is_open()){
```

```
Cout<<"no se pudo abrir el archivo"<<endl;
```

```
}else{
```

Necesito una estructura del tipo q almacena el archivo en este caso una persona

//creo una persona con los datos que quiero guardar

```
Persona p("matias","18/06/1985");
```

```
Archi.write((char *)&p, sizeof(persona));
```

```
o
```

//si ya los tenia guardados en algún lado ej si tengo ya cargado un vector de personas llamémoslo v se lo pasamos a la función write con la posición del vector q queramos agregar

```
Archi.write((char *)&v[0], sizeof(persona));//agrego solo la primera
```

```
O
```

si queremos copiar todas las personas del vector al archivo

```
for(int i=0;i<v.size();i++){
```

```
Archi.write((char *)&v[i], sizeof(persona));
```

```
}
```

```
Archi.close();
```