



# Temas

- Archivos de Texto de manejo secuencial
- Leer datos de un archivo
- Archivos de texto secuencial

Sobrecarga de operadores y  
manejo de archivos de texto



# Sobrecarga de operadores y manejo de archivos de texto



TECNOLÓGICO  
DE MONTERREY



## Archivos de Texto de manejo secuencial

---

- Declarar una variable de tipo archivo

`ifstream NomArch; // para archivos de entrada`

`ofstream NomArch; // para archivos de salida`

Ejemplo:

`ifstream ArchEnt;`

`ofstream ArchSal;`

- Abrir el archivo

`NomArch.open ("nombre");`

# Sobrecarga de operadores y manejo de archivos de texto



TECNOLÓGICO  
DE MONTERREY



## Archivos de Texto de manejo secuencial

---

Ejemplo:

```
ArchEnt.open ("datosEnt.txt");
```

```
ArchSal.open ("datosSal.txt");
```

- Declarar y abrir el archivo en un solo paso

```
ifstream nomVar ("nombre");
```

```
ofstream nomVar ("nombre");
```

Ejemplo:

```
ifstream ArchEnt ("datosEnt.txt");
```

```
ofstream ArchSal ("datosSal.txt");
```

# Sobrecarga de operadores y manejo de archivos de texto



TECNOLÓGICO  
DE MONTERREY



## Leer datos de un archivo

- Para leer datos de un archivo se usa el operador >> con el nombre del archivo. Hay que recordar que el >> elimina los blancos y returns y toma solamente el dato que se espera.
- Si se lee un string toma sola una palabra y elimina los espacios antes y después.

Ejemplo:

```
ArchEnt>>variable;
```

# Sobrecarga de operadores y manejo de archivos de texto



## Archivos de texto secuencial

### Escribir en un archivo

- Para escribir datos en un archivo se utiliza el operador << con el nombre del archivo.

Ejemplo:

```
ArchSal<<dato;
```

### Cerrar el archivo

```
NomArch.close( );
```

Ejemplo:

```
ArchEnt.close( );
```

```
ArchSal.close( );
```

# Sobrecarga de operadores y manejo de archivos de texto



## Archivos de texto secuencial

### La función eof( )

- Se utiliza para verificar cuando se llega al final del archivo.
- La función miembro eof( ) regresa Falso (cero) si no es el final del archivo y regresa Verdadero (diferente de cero) cuando llega al final del archivo.

Ejemplo:

```
if (ArchEnt.eof())  
  
    cout<<"Se terminó el archivo";
```



# Sobrecarga de operadores y manejo de archivos de texto



TECNOLÓGICO  
DE MONTERREY



## Archivos de texto secuencial

### Leer un caracter de un archivo

- Para leer un carácter del archivo se utiliza la función miembro `get( )` que regresa como valor de retorno el carácter leído.

Ejemplo:

```
char c;  
c = ArchEnt.get( );
```

### Leer un string de un archivo

- Llamamos string a una variable definida de la siguiente forma:

```
char var_str[n];
```

donde  $n-1$  es la cantidad de espacios que se están reservando

# Sobrecarga de operadores y manejo de archivos de texto



TECNOLÓGICO  
DE MONTERREY



## Archivos de texto secuencial

---

- (C++ usa 1 espacio para indicar el final del string).
- Para leer un string de un archivo se puede utilizar el operador >> junto con el nombre del archivo, pero al hacerlo el operador solamente lee una palabra y no toma la marca de fin de línea ni los espacios en blanco.
- También se puede utilizar la función miembro getline, que lee desde la posición actual del archivo hasta donde encuentre un carácter \n, o bien, cuando complete longitud caracteres, la función getline tiene el siguiente formato:

```
NomArch.getline(var_str, longitud);
```



# Sobrecarga de operadores y manejo de archivos de texto



TECNOLÓGICO  
DE MONTERREY



## Archivos de texto secuencial

---

- donde `var_str` es una variable de tipo string y
- `longitud` es la cantidad de espacios que se reservaron para el string.

Ejemplo:

```
char Nombre[20];
```

```
ArchEnt.getline(Nombre, 20);
```