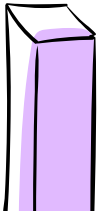
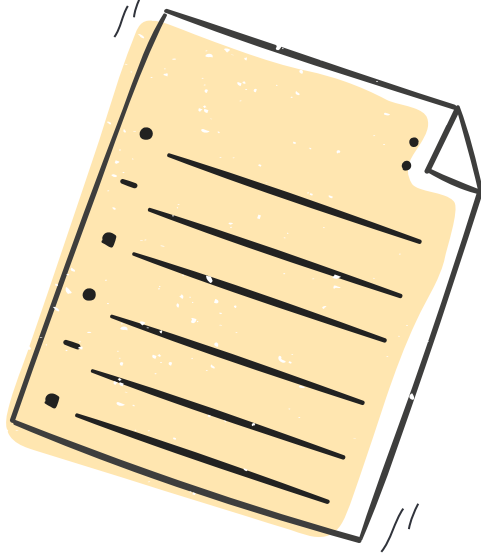


Cadenas

Uso string

Fundamentos de Programación





Temario

- string
- Operaciones con string
- Uso de getline()
- Problemas con getline
- Soluciones para getline

)

/





Cadena



Una **cadena en C++**, valores **tipo char**, **cadenas de C o matrices de caracteres nulos** es una serie de caracteres los cuales pueden ser tratados como una unidad y terminan en un carácter nulo.

Ejemplos de cadenas:

- Nombre
- Lugar de nacimiento
- Dirección
- Correo,, entre otros.

```
char universidad [3] = {'U','C','A'};  
char cadena [] = "Ejemplo";  
string materia = "Fundamentos Programación";
```

/





string



Strings, es una librería de c++, permite **la manipulación de cadenas** a diferencia de **char**, este **no tiene un tamaño específico** debido a que su tamaño se determinará durante el tiempo de ejecución.

string es un **tipo de dato** y el **valor asignado** a este tipo de dato debe estar **encerrado entre comillas dobles**.

La biblioteca <string.h> que permite **manipular las cadenas a través de objetos de cadenas**.

```
//Declaración de la variable
string nombre;
//inicialización
nombre = "Marlene";
```



Operaciones con string



Concatenación: Permite **unir dos o más string** a través del operador **+** o con la **función append()**.

Tamaño de la cadena: Permite **conocer la cantidad de caracteres en una cadena** a través de la **función length()** or **size()**.

Acceso a un carácter: Permite acceder a una posición en específico del string.



```
// Operaciones con string
string nombre = "Marlene";
string apellido = "Aguilar";
string apellidoSegundo = "Navarro";
//Concatenando con operador + y método append()
string nombreCompleto = nombre + apellido.append(
apellidoSegundo);
cout << nombreCompleto<<endl;
//Obteniendo el tamaño
cout<<nombreCompleto.size()<<endl;
//Accediendo a una posición
cout<<nombreCompleto[5]<<endl;
```

Material de Apoyo: https://www.w3schools.com/cpp/cpp_strings.asp





Uso de getline



getline, es un método que permite leer una cadena de caracteres o un flujo de entrada. funciona cuando se leen datos tipo de cadena de modo consecutivo.

getline(cin, <nombreVariable>);



```
string apellido;  
cout << "Ingresa tu apellido";  
getline(cin, apellido);  
cout << apellido;
```



Problema getline

La razón de los valores anteriores se debe a que al introducir 45, se debe pulsar la tecla **(ENTER)** a la terminación. Esta acción inserta un carácter CRLF (retorno de carro/avance de línea) y permanece en el buffer (memoria intermedia).

Cuando la sentencia `getline(cin,nombre)` se ejecuta, se lee la memoria intermedia del teclado y se encuentra el carácter CRLF, dado que éste es, por defecto, el carácter de separación, se detiene la lectura y se inserta el carácter de terminación nulo.

```
#include <iostream>

using namespace std;

int main(){

    int edad;
    string nombre;

    cout<<"Ingresa tu edad: "<<endl;
    cin>>edad;
    cout<<"Ingresa tu nombre: "<<endl;
    //cin>>nombre;45
    getline(cin,nombre);

    cout<<"Hola "<<nombre <<" tu edad "<<edad;

    return 0;
}

//////// Resultados de consola
Ingresa tu edad:
45
Ingresa tu nombre:
Hola  tu edad 45
```



Soluciones getline



Uso de un variable auxiliar, para almacenar los datos de la memoria intermedia.

```
#include <iostream>

using namespace std;

int main(){

    int edad;
    string nombre;
    string aux;

    cout<<"Ingresa tu edad: ";
    cin>>edad;
    //Limpiar el buffer de teclado
    getline(cin,aux);
    cout<<"Ingresa tu nombre: ";
    getline(cin,nombre);

    cout<<"Hola "<<nombre <<" tu edad "<<edad;

    return 0;
}

///////// Resultados de consola
Ingresa tu edad: 45
Ingresa tu nombre: Marlene E Aguilar
Hola Marlene E Aguilar tu edad 45
```




Soluciones getline



Uso de ignore, borra el carácter de salto de línea.

El método `getline` no puede descartar el de línea porque entonces la función tendría problemas para leer líneas que sí pueden estar vacías.

Antes de usar **getline** has usado el operador de **extracción >>**, importante usar `ignore` para descartar el salto de línea que se ha quedado en el buffer de entrada - por StackOverFlow.

```
#include <iostream>

using namespace std;

int main()
{
    int edad;
    string nombre;

    cout<<"Ingrese su edad";
    cin>>edad;
    //Uso de ignore()
    cin.ignore();
    cout<<"Ingrese su nombre";
    getline(cin, nombre);
    cout<<"Hola " << nombre << ", edad " << edad;

    return 0;
}
```



Consultas

meaguilar@uca.edu.sv

