

Caracteres

Un carácter es un símbolo que la computadora puede representar. Cuando se asigna una constante de carácter a una variable tipo char, esta debe estar entre apostrofes.

Ejemplo:

```
char letra='a';
```

Funciones mas importantes para el manejo de Caracteres

isdigit(c)	Devuelve 1 si s es un dígito y 0 en caso contrario
isalpha(c)	Devuelve 1 si s es una letra y 0 en caso contrario
islower(c)	Devuelve 1 si s es una letra minúscula y 0 en caso contrario
isupper(c)	Devuelve 1 si s es una letra mayúscula y 0 en caso contrario
tolower(c)	Convierte de mayúsculas a minúsculas. Si la letra es minúscula no modifica su valor.
toupper(c)	Convierte de minúsculas a mayúsculas. Si la letra es mayúscula no modifica su valor.

Ejemplo:

```
char x='a';
```

```
if( isdigit(x))
    ShowMessage("El carácter es un numero");
else
    if(isalpha(x))
        ShowMessage("El carácter es una letra");

    if(islower(x))
        ShowMessage("El character es una minuscule");
    else
        if(isupper(x))
            ShowMessage("El carácter es una mayuscula");

char w='R';
w=tolower(w);

char w='z';
w=toupper(w);
```

Cadenas de Caracteres

Las operaciones de Cadenas en C++ Builder son principalmente ejecutadas usando una clase llamada AnsiString. Muchos controles usan las Propiedades de AnsiString. Todos los controles que usan un titulo (form, panels, labels, etc), tienen su propiedad Caption colocado como un valor AnsiString. Muchos otros controles tales como el edit box usan la clase AnsiString en la propiedad text.

Para declarar una cadena, se usa la palabra AnsiString o solamente String seguido de un nombre valido en C++.

Ejemplo:

```
AnsiString pais; o tambien: String pais;
```

Para mayor comodidad usaremos la segunda.

1. Inicializacion de cadenas

Se declara la variable y después se le asigna el valor. Ejemplo:

```
String pais;
```

```
pais="Perú";
```

o tambien asignarle un valor al declarar la variable

```
String pais="Perú";
```

2. Verificar si una cadena esta vacía

Hay dos maneras que puede chequear si una cadena esta vacía

a) Usando el operador sobrecargado ==

Ejemplo:

```
String x="Hola";
```

```
if(x=="")
```

```
    ShowMessage("La cadena esta vacia");
```

```
else
```

```
    ShowMessage("La cadena no esta vacia");
```

b) Usando el metodo isEmpty() de la clase String. Este método devuelve verdadero si la cadena esta vacia o falso si no esta vacia.

Ejemplo:

```
String x="Hola";
```

```
if(x.isEmpty())
```

```
    ShowMessage("La cadena esta vacia");
```

```
else
```

```
    ShowMessage("La cadena no esta vacia");
```

3. Longitud de una cadena

Para devolver el numero de caracteres de una cadena se usa el metodo Length().

Ejemplo:

```
String cad="Marcelino";
```

```
int n;
```

```
n=cad.Length();
```

como la cadena tiene 9 caracteres n toma este valor.

4. Remover espacios en blanco de la parte izquierda de una cadena:

El Metodo **TrimLeft()** permite remover los espacios en blanco de la parte izquierda de una cadena.

Ejemplo:

```
String cadena=" C++ Builder ";
```

```
String resultado= cadena.TrimLeft();
```

En la variable resultado queda "C++ Builder ", sin los espacios de la izquierda.

5. Remover espacios en blanco de la parte derecha de una cadena:

El Metodo **TrimRight()** permite remover los espacios en blanco de la parte derecha de una cadena.

Ejemplo:

```
String cadena=" C++ Builder ";
```

```
String resultado= cadena.TrimRight();
```

En la variable resultado queda " C++ Builder" sin los espacios de la derecha.

6. Remover espacios vacios de la parte izquierda y derecha de una cadena

El metodo **trim()** devuelve una cadena sin espacios en blanco de la parte izquierda y derecha de una cadena

Ejemplo:

```
String cadena=" Esto es una prueba ";
```

```
String resultado= cadena.Trim();
```

En la variable queda como cadena="Esto es una prueba" sin los espacios de la izquierda ni la derecha.

7. Convertir a mayúsculas una cadena.

El método `UpperCase()` devuelve una cadena en mayúsculas.

Ejemplo:

```
String cadena="c++ builder 6.0";
```

```
String resultado=cadena.UpperCase();
```

En la variable resultado se asigna "C++ BUILDER 6.0".

8. Convertir a minúsculas una cadena

El método `LowerCase()` devuelve una cadena en minúsculas

Ejemplo:

```
String cadena="UNIVERSIDAD NACIONAL";
```

```
String resultado=cadena.LowerCase();
```

En la variable resultado se asigna "universidad nacional".

9. Concatenar cadenas

Para concatenar cadenas se puede hacer de dos maneras.

- Usando el operador `+`. El resultado es una cadena que combina la primera y la segunda cadena.

Ejemplo:

```
String cad1="Esto es";
```

```
String cad2="Una prueba";
```

```
String cad = cad1+cad2;
```

La cadena cad contiene "Esto esUna prueba", la concatenación de la cadena cad1 con la cadena cad2.

- Usando la función `AppendStr(cad1,cad2)` une cad2 al final de cad1. La cadena cad1 sale modificada.

Ejemplo:

```
String cad1="Esto es"
```

```
String cad2="Una prueba";
```

Al aplicar `Append(cad1,cad2)` obtenemos en cad1="Esto esUnaPrueba".

10. Comparación de Cadenas

Se puede comparar dos cadenas usando el operador `==`.

Además existen varias funciones para comparar cadenas:

- **SameText()**: Esta función requiere dos cadenas para compararlas. Esta comparación es ejecutada sin diferencias mayúsculas y minúsculas. Si las cadenas son iguales el resultado es true en caso contrario devuelve false.

Ejemplo:

```
String cad1="ALEMANIA";
```

```
String cad2="alemania";
```

```
if(SameText(cad1,cad2))
```

```
    ShowMessage("Las cadenas son iguales");
```

```
else
```

```
    ShowMessage("Las cadenas no son iguales");
```

En este ejemplo se mostrará el mensaje las cadenas son iguales.

- **AnsiCompareIC()**: Este método ejecuta una comparación de dos cadenas. También no diferencia las mayúsculas de las minúsculas.

```
Cad1.AnsiCompareIC(cad2)
```

Esta función devuelve un número entero:

```
>0 Si cad1 es mayor que cad2
```

```
=0 Si cad1 es igual que cad2
```

```
<0 Si cad1 es menor que cad2
```

Ejemplo:

```
String cad1="Alemania";
```

```
String cad2="Francia";
```

```
if(cad1.AnsiCompareIC(cad2)>0)
```

```
    ShowMessage("La cadena "+cad1+" es mayor");
```

```
else
```

```
    if(cad1.AnsiCompareIC(cad2)<0)
```

```
        ShowMessage("La cadena "+cad2+" es mayor");
```

```
    else
```

```
        ShowMessage("Las cadenas son iguales "
```

- **AnsiCompare():** Este método compara dos cadenas diferenciando las mayúsculas de las minúsculas.
cad1.AnsiCompare(cad2)

Esta función devuelve un número entero:

- >0 Si cad1 es mayor que cad2
- =0 Si cad1 es igual que cad2
- <0 Si cad1 es menor que cad2

11. Obtener el Último carácter de la cadena

Para hacer esto se usa el Método AnsiLastChar()

Ejemplo:

String cadena="Canción",

char car=cadena.AnsiLastChar();

El resultado es que en car se guardo el carácter 'n'.

12. Eliminar un grupo de caracteres de una cadena

cadena.Delete(posición, numCaracteres)

Este método elimina a partir de una posición una determinada cantidad de caracteres.

Ejemplo:

String cadena="CARRO";

cadena.Delete(2,3);

En este ejemplo le decimos que elimine a partir de la posición 2 la cantidad de 3 caracteres. El resultado de esto es que cadena se quedo como "CO"

13. Obtener una subcadena de una Cadena

cadena.SubString(posición, numCaracteres)

Este método devuelve una subcadena que esta compuesto por caracteres a partir de posición una determinada cantidad de caracteres

Ejemplo:

String cadena="CARRO";

String resultado=cadena.SubString(2,3);

La cadena resultado contiene "ARR"

14. Obtener la posición comienza una subcadena dentro de una cadena

cadena.Pos(subcadena)

Esta función devuelve la posición donde comienza subcadena dentro de cadena. Si la subcadena no se encuentra devuelve 0.

Ejemplo:

String cadena="Universidad Nacional de Trujillo"

String subcad="Nacional";

int p=cadena.Pos(subcad);

En este ejemplo p vale 13 pues la subcadena "Nacional" comienza en el indice 13.

Nota : En las cadenas tipo String el indice del primer carácter es 1.

15. Recorrer la cadena carácter por carácter

Las cadenas String en C++ las podemos considerar como un arreglo de caracteres cuyo índice inicial es uno. Para capturar cada carácter de la cadena podemos hacerlo de la siguiente manera:

Si declaramos la variable

String nombre="Juan"

nombre[0] es igual a 'j'

nombre[1] es igual a 'u'

nombre[2] es igual a 'a'

nombre[3] es igual a 'n'

Con el siguiente código podríamos obtener

n=nombre.Length();

for(i=1;i<=n;i++)

Hacer algo con nombre[i]

16. Convertir una cadena a un número Entero

El método ToInt() convierte una cadena conformada por caracteres numéricos a un número entero.

Ejemplo:

String S="123";

int x;

x=S.ToInt();

17. Convertir una cadena a un número Real

El método ToDouble() convierte una cadena conformada por caracteres numéricos a un número real.

Ejemplo:

```
String S="45.78";
```

```
float x;
```

```
x=S.ToDouble();
```

18. Insertar una cadena en otra

```
cadena1.Insert(cadena2,pos)
```

Este método inserta cadena2 en cadena1 a partir del índice pos.

Ejemplo:

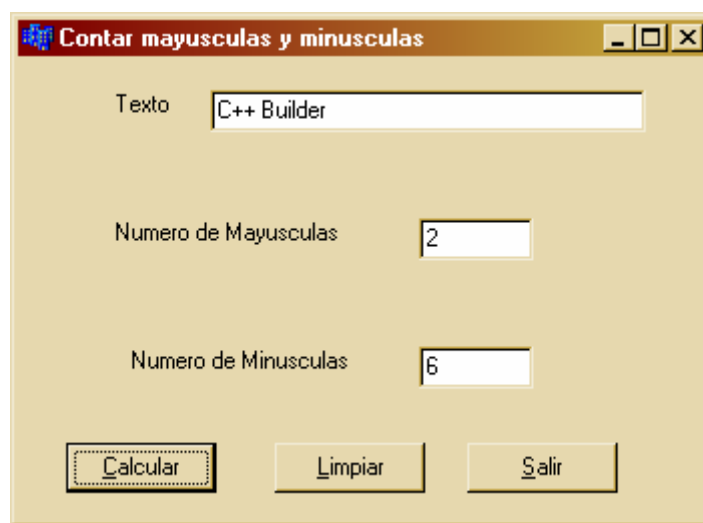
```
String cad1="Visual es facil";
```

```
String cad2="Basic ";
```

```
cad1.Insert(cad2,8);
```

Problemas Resueltos

1. Escriba un programa que ingrese una cadena de caracteres y determine el número de minúsculas y mayúsculas.



```
void __fastcall TForm1::btnCalcularClick(TObject *Sender)
{
    int i,n,cmay=0,cmin=0,cod;
    String texto;
    texto=edTexto->Text;
    n=texto.Length();
    for(i=1;i<=n;i++)
    {
        if(texto[i]>='A' && texto[i]<='Z')
            cmay=cmay+1;
        else
            if(texto[i]>='a' && texto[i]<='z')
                cmin=cmin+1;
    }
    edMay->Text=cmay;
    edMin->Text=cmin;
}
```

```
//-----
void __fastcall TForm1::btnLimpiarClick(TObject *Sender)
{
    edTexto->Clear();
    edMay->Clear();
    edMin->Clear();
    edTexto->SetFocus();
}
//-----
```

```

void __fastcall TForm1::btnSalirClick(TObject *Sender)
{
    Close();
}
//-----

```

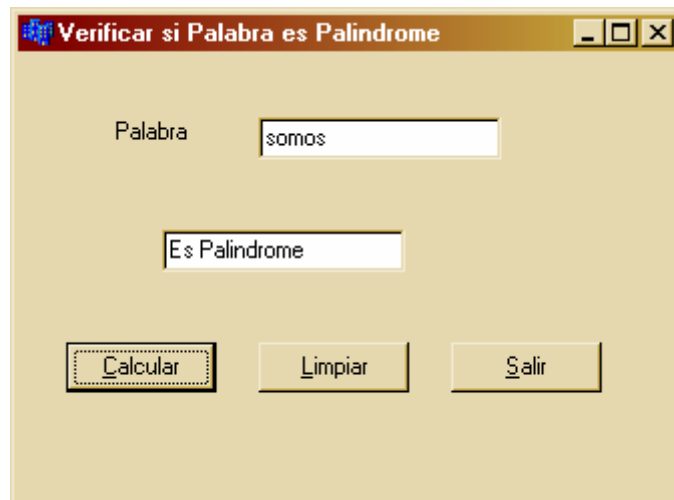
2) Escribir un programa que reciba como datos una cadena de caracteres y un carácter y reporta el numero de veces que se encuentra el carácter en la cadena.

```

void __fastcall TbtnLimpiar::btnCalcularClick(TObject *Sender)
{
    String texto;
    char car;
    int i,n,cv=0;
    texto=edTexto->Text;
    n=texto.Length();
    car=edCaracter->Text[1];
    for(i=1;i<=n;i++)
    {
        if(texto[i]==car) cv++;
    }
    edNV->Text=cv;
}
//-----
void __fastcall TbtnLimpiar::btnLimpiarClick(TObject *Sender)
{
    edTexto->Clear();
    edCaracter->Clear();
    edNV->Clear();
    edTexto->SetFocus();
}
//-----
void __fastcall TbtnLimpiar::btnSalirClick(TObject *Sender)
{
    Close();
}
//-----

```

3) Implemente una función que indique si una palabra es un palíndromo .
Una palabra es palidrome si se lee igual de izquierda a derecha que de derecha a izquierda..



```
void __fastcall TForm1::btnCalcularClick(TObject *Sender)
```

```
{
    String palabra;
    int i,j,n;
    bool exito=true;
    palabra=edPalabra->Text;
    n=palabra.Length();
    for(i=1,j=n;i<=n/2;i++,j--)
    {
        if(palabra[i]!=palabra[j])
            exito=false;
    }
    if(exito)
        edMensaje->Text="Es Palindrome";
    else
        edMensaje->Text="No es Palindrome";
```

```
}
//-----
```

```
void __fastcall TForm1::btnLimpiarClick(TObject *Sender)
```

```
{
    edPalabra->Clear();
    edMensaje->Clear();
    edPalabra->SetFocus();
```

```
}
//-----
```

```
void __fastcall TForm1::btnSalirClick(TObject *Sender)
```

```
{
    Close();
}
```

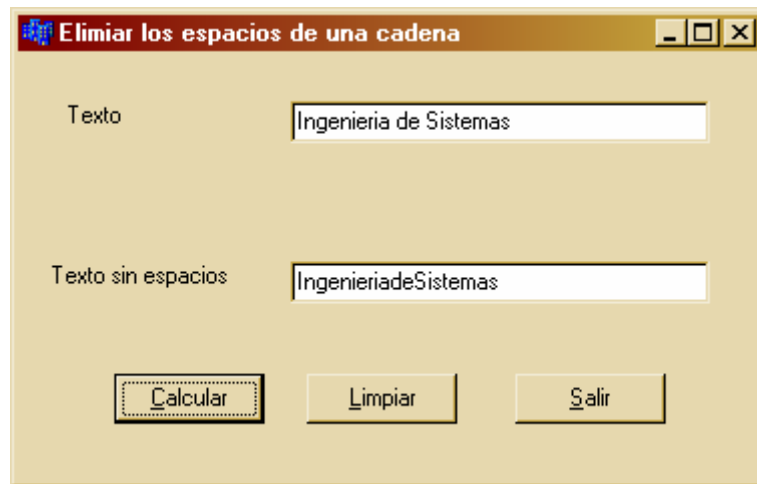
```
//-----
```

4) Implemente una función que reciba un string S y una letra X, y ponga en mayúsculas cada ocurrencia de X en S. (la función debe modificar la variable S).

The screenshot shows a simple graphical user interface for a string transformation program. The title bar reads 'Convertir a Mayusculas un caracter dado'. The interface includes three text boxes: the first labeled 'Texto' contains the word 'realidad'; the second labeled 'caracter' contains the letter 'a'; and the third labeled 'Texto Modificado' contains the result 'reAlidAd', where only the occurrences of the character 'a' have been converted to uppercase. Below the text boxes are three buttons: 'Calcular' (highlighted), 'Limpiar', and 'Salir'.

```
void __fastcall TForm1::Button1Click(TObject *Sender)
{
    String S;
    char X;
    int i,n;
    S=edTexto->Text;
    X=edCaracter->Text[1];
    n=S.Length();
    for(i=1;i<=n;i++)
    {
        if(S[i]==X) S[i]=toupper(S[i]);
    }
    edMod->Text=S;
}
//-----
void __fastcall TForm1::Button2Click(TObject *Sender)
{
    edTexto->Clear();
    edCaracter->Clear();
    edMod->Clear();
    edTexto->SetFocus();
}
//-----
void __fastcall TForm1::Button3Click(TObject *Sender)
{
    Close();
}
//-----
```

5) Hacer un programa que reciba un string S y devuelva el mismo string S al cual se le han eliminado los espacios.



```
//-----
#include <vcl.h>
#pragma hdrstop

#include "Unit1.h"
//-----
#pragma package(smart_init)
#pragma resource "*.dfm"
TForm1 *Form1;
//-----
__fastcall TForm1::TForm1(TComponent* Owner)
    : TForm(Owner)
{
}
//-----

void __fastcall TForm1::Button1Click(TObject *Sender)
{
    String S;
    int p;
    S=edTexto->Text;
    p=S.Pos(" ");
    while(p!=0)
    {
        S.Delete(p,1);
        p=S.Pos(" ");
    }
    edSinEspacios->Text=S;
}
//-----
void __fastcall TForm1::Button2Click(TObject *Sender)
{
    edTexto->Clear();
    edSinEspacios->Clear();
    edTexto->SetFocus();
}
//-----
void __fastcall TForm1::Button3Click(TObject *Sender)
{
    Close();
}
//-----
```

Problemas Propuestos

1. Contar el numero de vocales que se encuentran en una cadena
2. Contar el numero de consonantes que se encuentran en una cadena.
3. Dada una Cadena de caracteres invertirla.
4. Contar cuantas veces se encuentra una subcadena dentro de una cadena.
5. Ingresar una cadena de caracteres formadas por números y letras, en ese orden y obtenga otra cadena donde aparezca cada letra tantas veces como lo indique el numero que lo precede. Ejemplo: Si la cadena es 2w4r5f debemos obtener la siguiente cadena: wwrrrrffff
6. Ingresar una cadena y determinar cuantas palabras se encuentran en la cadena. Cada palabra se separa por medio de un espacio en blanco.
7. Leer una frase y encontrar la palabra de mayor longitud. El programa debe imprimir la palabra como el número de caracteres de la misma
8. Ingresar una cadena de caracteres y obtener otra con todos los caracteres de las posiciones impares de la cadena.
9. Hacer un programa que al recibir como dato una cadena de caracteres que represente una fecha en formato (dd/mm/aa), genere otra cadena con la misma fecha pero con formato (dd de nombre del mes de aaaa). Por ejemplo si se ingresa la cadena: 20/11/2006 la otra cadena será: 20 de Noviembre de 2006
10. Ingresar una frase y modificarla convirtiendo el primer carácter de cada palabra si esta fuera una letra, de minúsculas a mayúsculas.
11. Ingresar un numero telefónico en formato de cadena y luego lo convierta de la siguiente manera:
Numero Telefonico : 5256284000
Nueva Cadena : (52)-5-6284000
12. Hacer un programa que al recibir como datos dos cadenas de caracteres forme una tercera cadena intercalando los caracteres de las palabras de las cadenas recibidas.
13. Hacer un programa que permita la entrada de un nombre consistente en un nombre, un primer apellido y un segundo apellido, en ese orden, y que imprima a continuación el último apellido, seguido del primer apellido y el nombre.
14. Hacer un programa cuya entrada sea una cadena S y un factor de multiplicación N, cuya función sea generar la cadena N veces. Por ejemplo:
Hola,3 se convertirá en Hola Hola Hola
15. Hacer un programa que elimine todas las ocurrencias de una subcadena dentro de una cadena.
16. Hacer un programa que permita convertir un numero n en base 10 a un numero en base b (donde b esta entre 2 y 16).
17. Hacer un programa que permita convertir un numero en arábigos a romanos.
18. Ingrese un texto e indique que letras no han aparecido en el texto.
19. Ingrese un texto e indique el porcentaje de palabras que tiene menos de 5 caracteres y el porcentaje de palabras con 5 o mas caracteres.
20. Escriba un programa que lea un numero binario y muestre su valor decimal. Almacene los dígitos binarios en un String.
21. Construir una funcion para que busque una subcadena en una cadena a partir de una determinada posición y que devuelva la posición donde se encuentra la subcadena en caso contrario que devuelva cero.
22. Escriba un progrma que lea una frase ,sustituir todas las secuencias de dos o varios blancos por un solo blanco y visualizar la frase obtenida.
23. Escriba un programa que lea una frase y a continuación visualice cada palabra de la frase en columnas, seguida del número de letras que tiene cada palabra.
24. Escriba un programa que calcule la frecuencia de aparición de las vocales de un texto proporcionado por el usuario. Esta solucion se debe presentar en forma de histograma

a	15	*****
e	8	*****
...		

25. Ingresar una frase y reportar cada palabra en una linea diferente.