

||Funciones para el manejo de Strings

La mayoría de las f(x) especialmente dedicadas al manejo de cadenas de caracteres se encuentran en la librería string.h

```
strlen( );                                     #include <string.h>
```

Conocer la longitud de una cadena, devuelve un valor entero que es la cantidad de caracteres.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <conio.h>

int main ( ) {
    char string [ ] = "Borland International";
    int smax;
    printf ("\n\t\t FUNCION: strlen ( ):\n");
    printf ("\n char string [ ] = \"Borland International\\\"\\n\\n\\n\t");

    printf (" strlen (string) = %d\\n\\n\\n\\n\t", strlen (string));

    system("pause");
    printf (" smax = strlen (string);\\n\\n\\n\t");
    printf (" printf(\"%d\\", smax); // Imprime: ");

    smax = strlen (string);

    printf ("%d", smax);
    system("pause");
    return 0;
}
```

```
strcmp( );   ///string compare                #include
<string.h>
```

Comparar dos cadenas para saber si son iguales o no, devuelve un valor entero (0, > 0 ó < 0), cero si son iguales y mayor o menor que cero dependiendo del orden de comparación de los strings.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <conio.h>
int main () {
    char buf1 [ ] = "aaa", buf2 [ ] = "ccc", buf3 [ ] = "ddd";
    int res;
    res = strcmp (buf2, buf1);
    gotoxy (9,2);
    printf ("string2 es: \"%s\" y string1 es: \"%s\"", buf2, buf1);
    system("pause");
    gotoxy (9,5);
    printf ("string2 = string1 -----> res: 0\n\t");
    printf ("string2 < string1 -----> res: < 0\n\t");
    printf ("string2 > string1 -----> res: > 0\n\n\t");
    system("pause");
    if (res == 0)
        printf ("string2 es igual a string1 -----> res = %d\n\n\n\n\n\t", res);
    else if (res > 0)
        printf ("string2 es mayor que string1 -----> res = %d\n\n\n\n\n\t", res);
    else printf ("string2 es menor que string1 -----> res = %d\n\n\n\n\n\t", res);
    system("pause");
    res = strcmp (buf2, buf3);
    printf ("string2 es: \"%s\" y string3 es: \"%s\"\n\n\n\t", buf2, buf3);
    system("pause");
    printf ("string2 = string3 -----> res: 0\n\t");
    printf ("string2 < string3 -----> res: < 0\n\t");
    printf ("string2 > string3 -----> res: > 0\n\n\t");
    system("pause");
    if (res == 0)
        printf ("string2 es igual a string3 -----> res = %d\n\n\t", res);
    else if (res > 0)
        printf ("string2 es mayor que string3 -----> res = %d\n\n\t", res);
    else
        printf ("string2 es menor que string3 -----> res = %d", res);
    system("pause");
    return 0;
}
```

strcmpi();

#include <string.h>

Compara dos strings para ver si son iguales, pero sin diferenciar mayúsculas y minúsculas.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <conio.h>
int main () {
    char buf1 [ ] = "bbb", buf2 [ ] = "BBB";
    int res; clrscr ( );
    gotoxy (9,3);
    printf ("string2 es: \"%s\" y string1 es: \"%s\"", buf2, buf1);
    system("pause");
    res = strcmpi(buf2, buf1);
    if (res > 0)
        printf ("\n\n\tstring2 es mayor que string1 -----> res: %d\n", res);
    else if (res < 0)
        printf ("\n\n\tstring2 es menor que string1 -----> res: %d\n", res);
    else
        printf ("\n\n\tstring2 es igual que string1 -----> res: %d\n", res);
    system("pause");
    return 0;
}
```

```
strcpy( );
```

```
#include <string.h>
```

Copiar una cadena a otra, la primera es la cadena de destino, es decir, a donde se copiarán los datos y la segunda la de origen, la segunda sobrescribirá a la primera.

```
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>
#include <string.h>
int main ( ) {
    char string [10] = "nada";
    char str1[ ] = "abcdefghi";
    gotoxy (9,3);
    printf ("string es: \"%s\" y str1 es: \"%s\"\n\n\n\n\t", string, str1);
    system("pause");
    strcpy (string, str1); // strcpy (destino, origen);
    // sobrescribe los datos existentes en string
    printf ("Luego de copiarlo, string es: \"%s\"\n", string);
    system("pause");
    return 0;
}
```

```
strcat( );                                     #include <string.h>
```

Unir o concatenar dos o más strings, copia el contenido de la segunda cadena al final de la primera.

```
#include <stdio.h> #include <stdlib.h>
#include <string.h> #include <conio.h>

int main () {
    char destination [25];
    char destino [25];
    char blank [ ] = " ", c [ ] = "C++", turbo [ ] = "Turbo";
    strcpy (destination, turbo);
    strcat (destination, blank);
    strcat (destination, c);
    printf ("\n\t%s\n", destination);
    system("pause");
    strcpy (destino, turbo);
    strcat (strcat (destino, blank), c);
    printf ("\n\t%s\n", destino);
    system("pause");
    return 0;
}
```

```
strstr( );                                     #include <string.h>
```

Permite saber si una cadena está contenida dentro de otra. Devuelve un puntero a la posición de memoria donde se aloja el comienzo de la cadena buscada, si es que la encuentra, de lo contrario devuelve "NULL".

```
#include <stdio.h> #include <stdlib.h>
#include <string.h>
#include <conio.h>
int main () {
    char str1 [ ] = "Borland Int", str2 [ ] = "at", str3 [ ] = " Et" , * substr;
    substr = strstr(str1, str2);
    //Compara y muestra a partir de la subcadena encontrada.
    printf ("\n\tEl primer substring es: \"%s\"\n", substr);
    system("pause");
    substr = strstr(str1, str3);
    //Compara y si no encuentra la subcadena devuelve NULL.
    printf ("\n\tEl segundo substring es: \"%s\"\n", substr);
    system("pause");
    return 0;
}
```

```
}
```

```
tolower( ); toupper( );
```

```
#include <ctype.h>
```

En la librería standard <ctype.h> se encuentran estas funciones que nos posibilitan transformar a minúsculas o mayúsculas cualquier string.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <conio.h>
#include <ctype.h>

int main () {
    int length, i;
    char string [ ] = "ESTO ERA UN STRING en MAYUSCULAS";
    length = strlen (string);
    for (i=0; i < length; i++) {
        string [i] = tolower (string [i]);
    }
    printf ("\n\n\t%s\n", string);
    system("pause");
    char str [ ] = "y ahora tenemos un string en minusculas" ;
    length = strlen (str) ;
    for (i=0; i < length; i++) {
        str [i] = toupper (str [i]);
    }
    printf ("\n\n\t%s\n", str);
    system("pause");
    return 0;
}
```

```
strcoll();
```

```
#include <string.h>
```

Comparar dos strings, devuelve un int, cero (0) si son iguales, menor que cero (0) si el primero es menor que el segundo y mayor que cero (0) si el primero es mayor que el segundo.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
```

```
#include <conio.h>

int main ( ) {
    char two [ ] = "International";
    char one [ ] = "Borland";
    int check;
    check = strcoll (one, two) ;
    if (check == 0)
        printf ("The strings are equal\n"); // las cadenas son iguales
    else if (check < 0)
        printf ("%s comes before %s\n", one, two); // la primer cadena es menor
    else
        printf ("%s comes before %s\n", two, one); // la primer cadena es mayor
    system("pause");
    return 0;
}
```

```
gets(); puts();
```

```
#include <string.h>
```

```
int main()
{
    ///Declaracion del string
    char palabra[] = "";

    ///Pedir datos al usuario
    printf("A continuación, ingrese una palabra: ");
    fflush(stdin);
    //Ingresa un string desde el teclado, permite ingresar espacios en blanco
    gets(palabra);

    ///Mostrar datos por pantalla
    printf("La palabra ingresada es: ");
    //Muestra el string por pantalla, reemplaza el printf
    puts(palabra);

    return 0;
}
```

```
Otras funciones de string
```

```
#include <string.h>
```

```
strcspn ( );
```

```
// compara y devuelve la primera posición coincidente entre dos strings.
strspn ( );
// compara y devuelve la primera posición diferente entre dos strings.
strdup ( );
// copia un string a una nueva posición, desde la original, a través de punteros.
strchr ( );
// busca en un string la primera ocurrencia de un carácter especificado, con punteros.
strrev ( );
// invierte un string, mediante el uso de punteros.
strupr ( );
// convierte un string a mayúsculas, usa punteros.
strlwr ( );
// convierte un string a minúsculas, usa punteros.
```