

**INSTITUTO POLITÉCNICO NACIONAL  
UNIDAD PROFESIONAL INTERDISCIPLINARIA EN INGENIERÍA Y TECNOLOGÍAS  
AVANZADAS**

## **Práctica No. 5. Manejo Cadenas.**

**Unidad Temática:** III Manejo de Funciones y cadenas

**Lugar de realización:** Laboratorio de Cómputo

**Duración:** 9.5 hrs.

### **Objetivo**

Desarrollar programas utilizando cadenas para el desplegado de información.

### **Resultados Esperados**

- Trabajo de investigación logrado en el pre-reporte
- Solución de casos prácticos en Lenguaje C haciendo uso de funciones que se encuentran en la librería `string.h` para el manejo de cadenas.



### **Pre-reporte.**

Para el buen desarrollo de la práctica el alumno entregará un trabajo previo que incluya:

1. Mostrar dos ejemplos dónde se haga uso del manejo de cadenas a través de las funciones que nos proporciona la librería `string.h`.

*“Convierte los puntos en letras, las letras en palabras, las palabras en un mundo. “  
(Anthony Doerr)*

### **Material y Equipo.**

- IDE Dev-Cpp o compatible.
- Computadora.

## Introducción.

### String

Algunas de las funciones contenidas en la librería `string.h` que permiten el manejo de cadenas (líneas de texto o conjunto de caracteres) en un programa en C/C++ son:

- `strcat`
- `strcmp`
- `strcpy`
- `strlen`
- `strtok`

### Descripción de funciones en string

La función `strcat`, permite concatenar dos cadenas de caracteres.

**Ejemplo:**

```
char primer_nombre[] = "Adriana";
strcat(primer_nombre, " Ramirez");
printf("Resultado: %s", primer_nombre);
```



Figura 5.1. Salida de función `strcat`

La función `strcmp`, permite comparar si dos cadenas de caracteres son iguales, la función regresa un 0 en caso de que lo sean.

**Ejemplo:**

```
char completo[] = "Hola";
if(strcmp(completo, "Hola")==0)
    printf("Son iguales las cadenas");
else
    printf("No son iguales las cadenas");
```



Figura 5.2. Salida de función `strcmp`

La función `strcpy`, permite copiar una cadena a la cadena principal.

**Ejemplo:**

```
char primer_nombre[] = "Bella";
strcpy(primer_nombre, "Citlalli");
printf("Resultado: %s", primer_nombre);
```

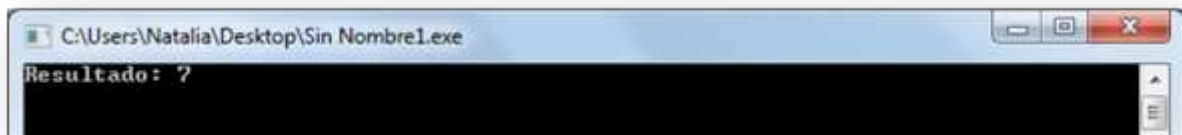


**Figura 5.3. Salida de función strcpy**

La función `strlen`, devuelve el tamaño de la cadena de caracteres.

**Ejemplo:**

```
char primer_nombre[] = "Adriana";
int longitud = strlen(primer_nombre);
printf("Resultado: %d", longitud);
```



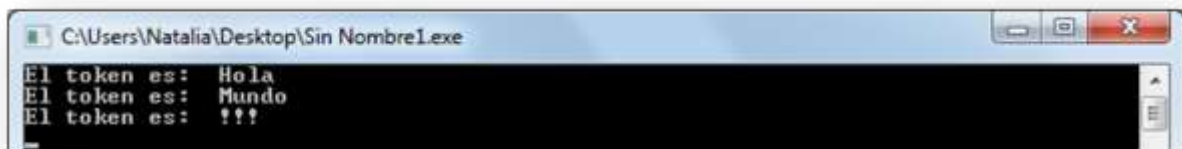
**Figura 5.4. Salida de función strlen**

La función `strtok`, permite partir una cadena en subcadenas usando como separador los caracteres que sean necesarios.

**Ejemplo:**

```
char string[] = "Hola Mundo !!!";
char *token = strtok(string, " "); //Separamos por espacios
                                   //la cadena

while (token != NULL){
    printf("El token es: %s\n", token);
    token = strtok(NULL, " ");
}
```



**Figura 5.5. Salida de función strtok**

## Desarrollo.

### Ejemplo:

Escribir un programa que simule un login (inicio de sesión), en el cual si el usuario y la contraseña introducidos son correctos mostrar el mensaje de Bienvenido y el nombre del usuario, pero si no son correctos mostrar un mensaje de Usuario y/o contraseña incorrectos.

### Solución:

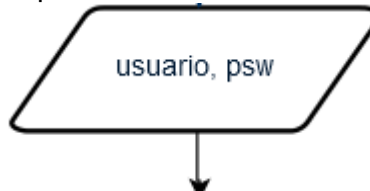
#### Diagrama de Flujo

1. Inicio del programa principal

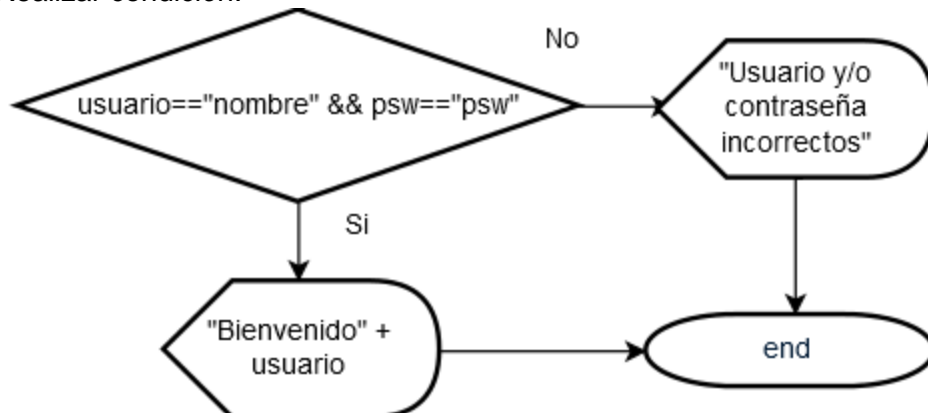


2. Identificación de entradas:

Las entradas ingresadas por el usuario son: usuario y psw



3. Realizar condición:



4. Identificación de salida

La salida será el mensaje de "Bienvenida" en caso de que el usuario y contraseña sean correctos y caso contrario un mensaje de "Usuario y/o contraseña incorrectos".

5. Fin

#### Programa en C *Ver código 5contrasenia.c anexo en disco*

1. Inicio del programa

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

2. Identificación de entradas:

Declarar las variables a utilizar en el programa

```
int main() {
```

```

char usuario[20], psw[20];
Ingresar el usuario y la contraseña:
scanf("%s", usuario);
scanf("%s", psw);
3. Realizar condición:
if(strcmp(usuario, "nombre")==0 && strcmp(psw,
"psw")==0)
    printf("Bienvenido: %s", usuario);
else
    printf("Usuario y/o contraseña incorrectos");
4. Fin de programa
}

```

## Programas

Realice los siguientes ejercicios:

1. Programa que introduzca una frase y muestre por separado cada una de las palabras contenidas en esa frase.

Entrada:

Valor de la variable: frase.

Procedimiento:

- 1) Declarar la variable frase como un arreglo de caracteres.
- 2) Introducir el valor de la variable frase.
- 3) Utilizar la función `strtok` definida en la librería `string.h` para separar las palabras.
- 4) Mostrar el resultado.

Salida:

Suponga el valor de entrada:

frase = Programacion en C

El resultado sería:

Resultado = Programacion  
en  
C

2. Programa que reciba como dato una línea de texto y escriba esa línea en forma inversa.

Entrada:

Valor de la variable: linea.

Procedimiento:

- 1) Declarar la variable linea como un arreglo de caracteres.
- 2) Introducir el valor de la variable linea.
- 3) Obtener el tamaño de la cadena usando la función `strlen` definida en la librería `string.h`
- 4) Recorrer la cadena al revés haciendo uso de un ciclo y guardarla en otra cadena.
- 5) Mostrar el resultado.

Salida:

Suponga el valor de entrada:

linea = Hola Mexico

El resultado sería:

Resultado = ocixeM aloH

3. Escribir un programa que cuente el número total de vocales en una línea de texto e imprima su frecuencia.

Entrada:

Valor de la variable: `linea`.

Procedimiento:

- 1) Declarar la variable `linea` como un arreglo de caracteres.
- 2) Introducir el valor de la variable `linea`.
- 3) Recorrer la cadena usando un ciclo y contar el número de vocales.
- 4) Mostrar el resultado.

Salida:

Suponga el valor de entrada:  
`linea = Murcielago`

El resultado sería:  
Resultado = a:1 e:1 i:1 o:1 u:1

4. Escribir un programa que, al recibir como dato una fecha en formato (dd/mm/aa), genere otra cadena con la misma fecha pero con formato (dd de nombre del mes de aaaa).

Entrada:

Valor de la variable: `linea`.

Procedimiento:

- 1) Declarar la variable `linea` como un arreglo de caracteres.
- 2) Introducir el valor de la variable `linea`.
- 3) Utilizar la función `strtok` definida en la librería `string.h` para obtener por separado día, mes y año.
- 4) Generar la nueva cadena en el nuevo formato.
- 5) Mostrar el resultado.

Salida:

Suponga el valor de entrada:  
`linea = 06/08/05`

El resultado sería:  
Resultado = 06 de Agosto de 2005

### Proyectos (opcionales).

1. Codificar un programa que simule un examen, el programa deberá hacerle preguntas al usuario y comparar si la respuesta que introdujo es la correcta, al final deberá mostrarle su calificación.
2. Codificar un programa que imprima si una frase es palíndroma o no haciendo uso de funciones de manejo de cadenas.

### Ponderación de la Práctica.

Sección	Elemento a Evaluar
Pre-reporte	Ejemplos de manejo de cadenas
Ejercicio 1	Manejo de cadenas aplicando la función <code>strtok</code>

<b>Ejercicio 2</b>	Manejo y recorrido de cadenas, uso de la función <code>strlen</code>
<b>Ejercicio 3</b>	Manejo y recorrido de cadenas, uso de la función <code>strlen</code>
<b>Ejercicio 4</b>	Manejo de cadenas aplicando la función <code>strtok</code>

## Bibliografía.

Deitel P.J.y Deitel H. M., Como Programar C++, Ed. Prentice Hall, 6<sup>a</sup> Impresión, México, 2009, ISBN: 970-26-1273-X, Págs: 1-1050.

Joyanes Aguilar Luis , Fundamentos de Programación, Ed Mc Graw Hill, 4<sup>a</sup> Impresión, España, 2008, ISBN: 8448161114, Págs: 47-73, 74-75, 76-101, 113-141 y 151 -534.

Guardati, Silvia, Estructura de Datos Orientado a Objetos con C ++, Ed.Prentice, México 2007, ISBN: 9702607922, Págs: 1–183.

Cairó Battistuti, Fundamentos de Programación. Piensa en C, Ed. Prentice Hall, México 2006, ISBN: 970-26-0810-4

**NOTA:** Presentar el reporte en un documento (PDF o DOC), el código fuente y la impresión de la pantalla de ejecución.

**"Puedes apoyarte de tus compañeros y profesor para aclarar tus dudas"**