|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nombre de la práctica** | **Cadenas** | | | **No.** | **1** |
| **Asignatura:** | **Métodos Numéricos** | **Carrera:** | **INGENIERÍA EN SISTEMAS COMPUTACIONALES** | **Duración de la práctica (Hrs)** | **2** |

**I. Competencia(s) específica(s):**

**II. Lugar de realización de la práctica (laboratorio, taller, aula u otro):**

**III. Material empleado:**

**IV. Desarrollo de la práctica:**

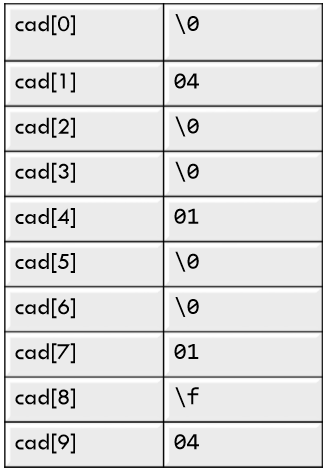
**CADENAS**

Antes de empezar debemos tener claro

¿Qué es una cadena?

Una cadena es un arreglo de caracteres. En donde, por lo general el último elemento deberá ser el carácter ‘\0’.

¿Cómo declarar una cadena?

 Crearla como un arreglo

Char c [10];

Insertamos un tipo de dato que almacena datos de carácter en un campo de longitud fija ya sea que los datos tengan una serie de letras, números y otros caracteres.

Int main () {

Char cad [10];

int i;

Printf ("Introduce una palabra \n");

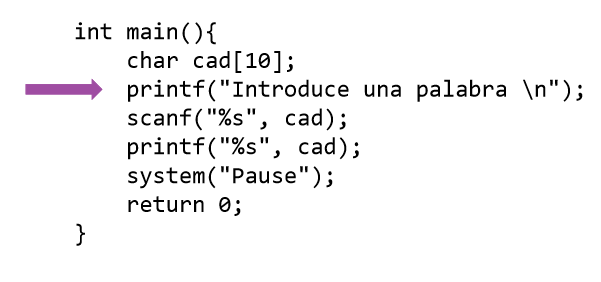
Scanf ("%s", cad);

Printf ("%s", cad);

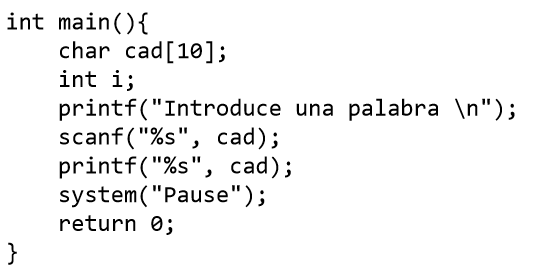
System ("Pause");

return 0;

}

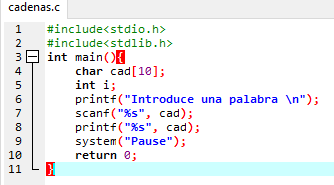


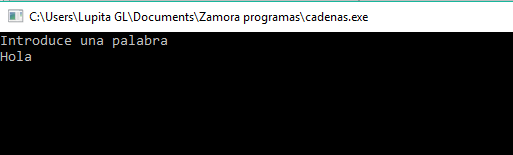
Printf muestra en pantalla un mensaje donde el usuario debe de ingresar lo que esté pidiendo el programa.

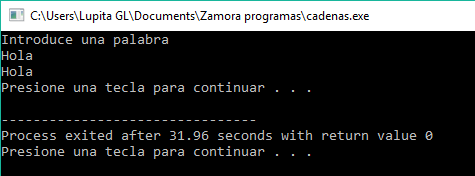


Muestra los datos ingresados por teclado.

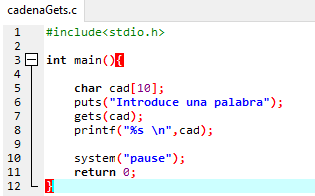
Utilizamos un Scanf para que el texto se pueda ingresar por teclado.







**Cadenas gets**



Insertamos un tipo de dato que almacena datos de carácter en un campo de longitud fija ya sea que los datos tengan una serie de letras, números y otros caracteres.

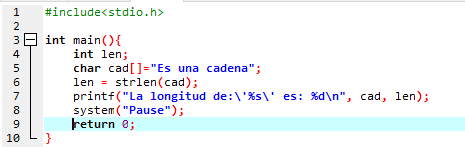
Recibe un argumento la variable en la que debe de almacenarse una entrada de usuario.

Recibe un argumento; donde la cadena de caracteres se muestra en pantalla.

**Funciones de cadena: strlen**

Devuelve la longitud de la cadena sin tomar en cuenta el carácter de final de cadena.

Strlen (<cadena>)

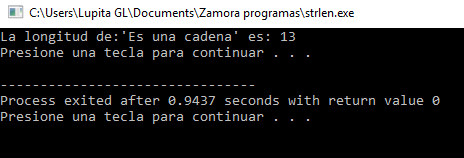


Devuelve la longitud de la cadena sin tomar en cuenta el carácter de final de cadena.

Es declarada un tipo de dato que permite al usuario introducir datos por teclado.

Bibliotecas

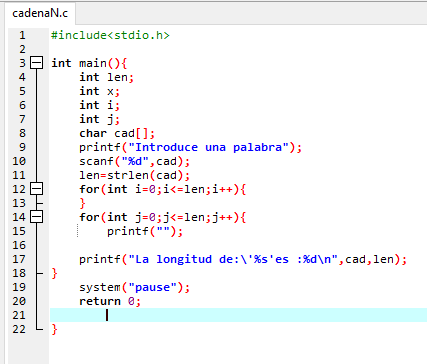
Imprime la longitud en pantalla.



**Ejercicio:**

**Escribe un programa que reciba una palabra por teclado.**

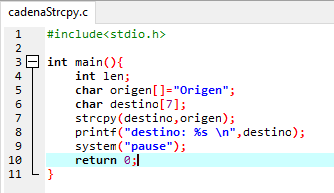
**De acuerdo a la longitud de la palabra (N) que se ingresó por teclado imprime un cuadrado de asteriscos de (N x N).**



**Funciones de cadena: strcpy**

**Copia el contenido de: <cadena\_origen> en <cadena\_destino>.**

**Strcpy (<cadena\_destino>, <cadena\_origen>)**



Cadena origen donde se ingresa un carácter.

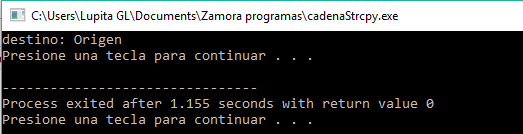
Cadena origen donde se ingresa un carácter

Copia el contenido de cadena origen a cadena destino.

Variable de tipo entero

Bibliotecas

Imprime el contenido de la cadena destino

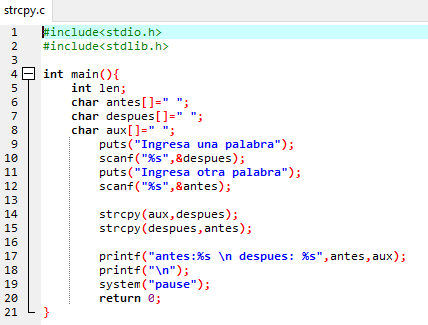


**Ejercicio**

**Escribe un programa que reciba por teclado dos palabras y cada una de ellas las almacene en un arreglo.**

**Después intercambia sus contenidos. Imprime el antes y el después.**

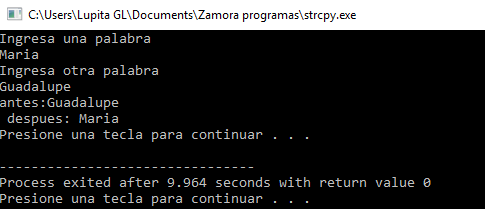
Bibliotecas



Solo captura una cadena de texto hasta que aparezca el primer espacio o fin de línea.

La función Puts () recibe un único argumento: la cadena de caracteres a mostrar en pantalla.

Variable de tipo entera

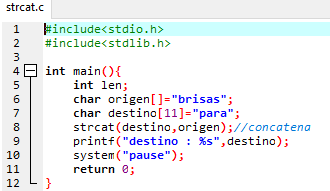


**Funciones de cadena: strcat**

**Concatena el contenido de <cadena\_origen> al final de <cadena\_destino>**

**Strcat (<cadena\_destino>, <cadena\_origen>)**

Bibliotecas

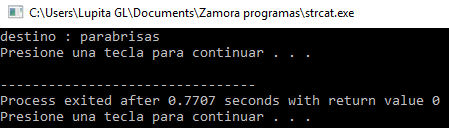


Cadena destino

Cadena origen

Concatenación de cadena

Variable de tipo entera

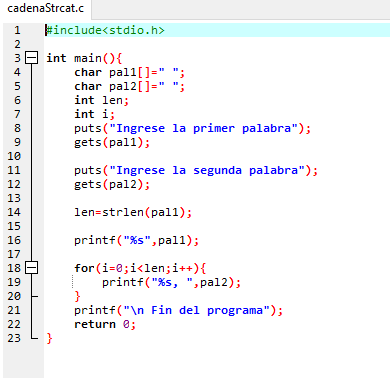


**Ejercicio**

**Escribe un programa que reciba por teclado dos palabras.**

**Y concatene N veces la segunda palabra a la primer palabra.**

**Donde N es la longitud de la primera palabra.**



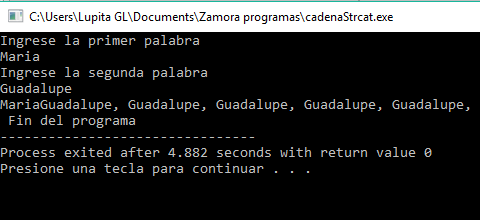
Recibe un argumento la variable en la que debe de almacenarse una entrada de usuario.

Recibe un argumento; donde la cadena de caracteres se muestra en pantalla.

Insertamos un tipo de dato que almacena datos de carácter en un campo de longitud fija ya sea que los datos tengan una serie de letras, números y otros caracteres.

Variables de tipo entero

Bibliotecas



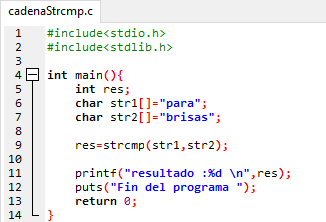
**Funciones de cadena: strcmp**

**Compara las dos cadenas y devuelve un 0 si las dos cadenas son iguales.**

**Un número negativo si <cadena1> es menor que <cadena2>**

**Un número positivo (mayor que cero) si <cadena1> es mayor que <cadena2>.**

**Strcmp (<cadena1>, <cadena2>)**



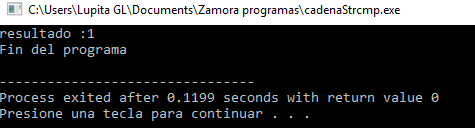
Variable de tipo char donde que almacena datos de carácter en un campo de longitud fija ya sea que los datos tengan una serie de letras, números y otros caracteres.

Variable de tipo entero

Bibliotecas

Imprime el resultado en pantalla

Strcmp Sirve para comparar la cadena1 con la cadena2.



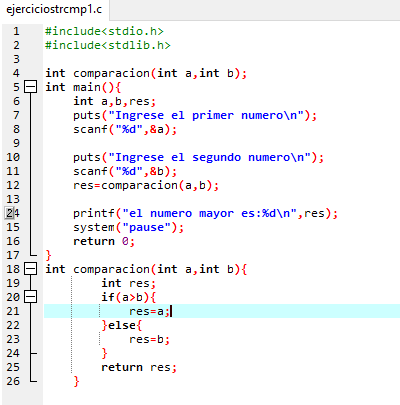
**Escribe un programa que reciba por teclado dos palabras y te indique cuál de ellas es mayor y cuál es la menor.**

**En caso de ser iguales, que imprima la leyenda 'ambas palabras son iguales'.**

**Compara las dos cadenas y devuelve un 0 si las dos cadenas son iguales.**

**Un número negativo si <cadena1> es menor que <cadena2>**

**Un número positivo (mayor que cero) si <cadena1> es mayor que <cadena2>.**



Se crea un if que comparan los números y si es mayor que b se guarda en el valor de a en res y si no se cumple res se almacena en b.

Se declara una variable

Se manda a llamar la función creada.

Se imprime res

Se pasan los datos por parámetros y se guardan los datos en res

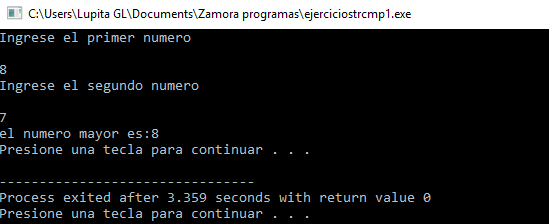
Se guarda el valor en a.

Se imprime un mensaje solicitando un número

Se crean variables

Se declara la función

Se crea el método principal



**Crea un programa que detecte una palabra palíndroma.**

**Los palíndromos son palabras que se leen igual de izquierda a derecha que de derecha a izquierda.**

**Crea un programa que cuente cuantas ocurrencias de cada letra contiene una palabra.**

**Ejemplo:**

**Palabra**

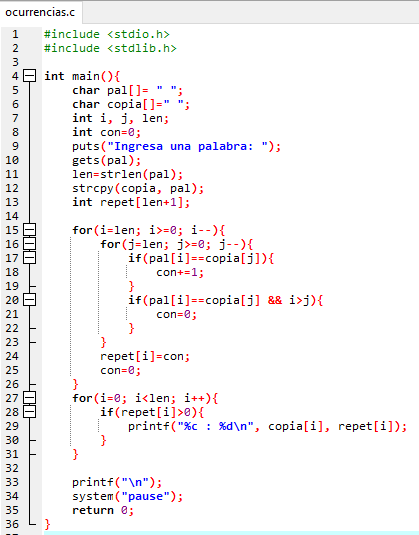
**P→1**

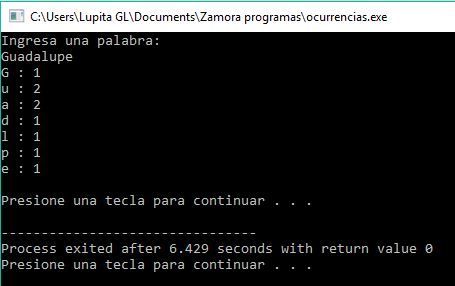
**a →3**

**l →1**

**b →1**

**r →1**





**V. Conclusiones:**

El tema es un arreglo de caracteres. En donde, por lo general el último elemento deberá ser el carácter o 0 las cadenas son importantes a diferencia de otros lenguajes de programación que emplea un tipo denominado cadena String para manipular un conjunto de símbolos, se debe de simular mediante un arreglo de caracteres, en donde la terminación de la cadena se debe de indicar con el nulo.

Se me hizo complicado entender la función de cada cadena pero es muy importante tratar de comprenderla para que se nos haga más fácil debemos de practicar y practicar para que tengamos una buena lógica.