



**Universidad Autónoma de Baja California**  
**Facultad de Ingeniería, Arquitectura y Diseño**



# Ingeniero en Software y Tecnologías Emergentes

Programación Estructurara/36276

Santos Tirado Martin/00369705

Pedro Nuñez Yepiz

Actividad N. 7

Practica con ejercicios de cadenas y funciones

Ensenada Baja California, 25 de septiembre del 2023

**Introducción:**

Veremos ejercicios que requiere de cadenas y funciones que cambien su estructura

**Competencia:**

El alumno llevará a cabo los ejercicios planteados con ayuda del profesor que dejo en un documento sobre el uso de las cadenas y sus diferentes cambios en ellas tales como, mayúsculas, minúsculas, capital, de arriba hacia abajo, entre otros.

**Fundamentos:**

El alumno aprenderá sobre el funcionamiento del uso de las cadenas con sus distintas maneras de modificar su estructura, juagara con la palabra intercambiándolas de distintas maneras, además que, utilizará como apoyo el código ASCII. <https://elcodigoascii.com.ar/>

## Procedimiento:

### PARTE 1:

Realizar un programa que contenga funciones que realice lo siguiente.

- 1.- Leer una cadena y desplegarla de la siguiente manera:  
(Realizar una función para cada salida)

cadena: Ensenada

<b>SALIDA 1</b> ENSENADA	<b>SALIDA 2</b> ADANESNE	<b>SALIDA 3</b> E N S E N A D A
<b>SALIDA 4</b> A D A N E S N E	<b>SALIDA 5</b> ENSENADA ENSENAD ENSENA ENSEN ENSE ENS EN E	<b>SALIDA 6</b> ADANESNE ADANESNE ADANES ADANE ADAN ADA AD A
<b>SALIDA 7</b>	<b>SALIDA 8</b>	<b>SALIDA 9</b>

ENSENADA NSENADA SENADA ENADA NADA ADA DA A	ADANESNE DANESNE ANESNE NESNE ESNE SNE NE E	NSND  SALIDA 10  EEAA

#### LIGAS DE INTERÉS Y AYUDA

<http://www.uco.es/grupos/eatco/informatica/metodologia/cadenasyarrays.pdf>

[http://platea.pntic.mec.es/vgonzalez/cyr\\_0204/cyr\\_01/control/lenguajeC/cadenas.htm](http://platea.pntic.mec.es/vgonzalez/cyr_0204/cyr_01/control/lenguajeC/cadenas.htm)

<http://c.conclase.net/cursos/?cap=008>

[https://www.aprenderaprogramar.com/index.php?option=com\\_content&view=article&id=934:funciones-para-cadenas-en-c-longitud-sizeof-string-y-strcpy-strlen-strcat-strcmp-ejemplos-cu00535f&catid=82&Itemid=210](https://www.aprenderaprogramar.com/index.php?option=com_content&view=article&id=934:funciones-para-cadenas-en-c-longitud-sizeof-string-y-strcpy-strlen-strcat-strcmp-ejemplos-cu00535f&catid=82&Itemid=210)

<https://es.slideshare.net/angenio2/programacin-1-cadenas-en-c>

<https://pablohaya.com/2013/10/12/diferencia-entre-scanf-gets-y-fgets/>

## PARTE 2:

### REALIZA LAS SIGUIENTES FUNCIONES:

- 1.- Función que reciba como parámetro una cadena y la convierta a MAYUSCULAS
- 2.- Función Que reciba como parámetro una cadena y la convierta a MINUSCULAS
- 3.- Función que reciba como parámetro una cadena y la convierta a CAPITAL
- 4.-Función que reciba como parámetro una cadena y retorne la cantidad de caracteres que tiene la cadena.
- 5.-Función que reciba como parámetro una cadena y retorne una cadena con sus caracteres acomodados de forma inversa (al revés)
- 6.-Función que reciba como parámetro una cadena y genere una nueva cadena basada en la original pero sin espacios.
- 7.-Función que sirva para leer una cadena y solo permita caracteres alfabéticos (A...Z) y el espacio, donde una cadena no puede comenzar o terminar con espacio, no debe tener dos espacios seguidos. retornar la cadena ya sea como parámetro o variable.
- 8.-Función que reciba como parámetro una cadena, y utilizando las funciones anteriores, imprima en MAYUSCULAS, MINUSCULAS , CAPITAL, SIN ESPACIOS, ALREVES la cadena original.
- 9.-Función que reciba como parámetro una cadena, y desplegar la leyenda si la cadena es un palíndromo SI o NO  
**(VALIDADA AL 100% NO NUMEROS, NO DOBLES ESPACIOS Y SOLO MAYUSCULAS EN LA CADENA)**

```

#include "libreria.h"

void funcion1(char cad[]);
void funcion2(char cad[],int tam,int i);
void funcion3(char cad[],int tam,int i);
void funcion4(char cad[],int tam,int i);
void funcion5(char cad[],int tam,int i,int j);
void funcion6(char cad[],int tam,int i,int j);
void funcion7(char cad[],int tam,int i,int j);
void funcion8(char cad[],int tam,int i,int j);
void funcion9(char cad[],int tam,int i);
void funcion10(char cad[],int tam,int i);

////////////////////////////////////

int main()
{
    system("CLS");
    char cad[30];
    int tam, i, j;
    printf("Dame una palabra: ");
    fflush(stdin);
    gets(cad);
    tam = tam_cad(cad);

    mayusculas(cad);

    funcion1(cad);
    funcion2(cad, tam, i);
    funcion3(cad, tam, i);
    funcion4(cad, tam, i);
    funcion5(cad, tam, i, j);
    funcion6(cad, tam, i, j);
    funcion7(cad, tam, i, j);
    funcion8(cad, tam, i, j);
    funcion9(cad, tam, i);
    funcion10(cad, tam, i);

    system("PAUSE");
    return 0;
}

////////////////////////////////////

```

```
void funcion1(char cad[])
{
    printf("SALIDA 1:\n");
    puts(cad);
    printf("\n");
}

void funcion2(char cad[], int tam, int i)
{
    printf("SALIDA 2:\n");
    for (i = tam - 1; i >= 0; i--)
    {
        printf("%c", cad[i]);
    }
    printf("\n\n");
}

void funcion3(char cad[], int tam, int i)
{
    printf("SALIDA 3:\n");
    for (i = 0; i < tam; i++)
    {
        printf("%c\n", cad[i]);
    }
    printf("\n");
}

void funcion4(char cad[], int tam, int i)
{
    printf("SALIDA 4:\n");
    for (i = tam - 1; i >= 0; i--)
    {
        printf("%c\n", cad[i]);
    }
    printf("\n");
}

void funcion5(char cad[], int tam, int i, int j)
{
    printf("SALIDA 5:\n");
    for (j = tam - 1; j >= 0; j--)
    {
        for (i = 0; i <= j; i++)
        {
            printf("%c", cad[i]);
        }
    }
}
```

```

    }
    printf("\n");
}
printf("\n");
}

void funcion6(char cad[], int tam, int i, int j)
{
    printf("SALIDA 6:\n");
    for (j = 0; j <= tam - 1; j++)
    {
        for (i = tam - 1; i >= j; i--)
        {
            printf("%c", cad[i]);
        }
        printf("\n");
    }
    printf("\n");
}

void funcion7(char cad[], int tam, int i, int j)
{
    printf("SALIDA 7:\n");
    for (j = 0; j <= tam - 1; j++)
    {
        for (i = j; i <= tam - 1; i++)
        {
            printf("%c", cad[i]);
        }
        printf("\n");
    }
    printf("\n");
}

void funcion8(char cad[], int tam, int i, int j)
{
    printf("SALIDA 8:\n");
    for (j = tam - 1; j >= 0; j--)
    {
        for (i = j; i >= 0; i--)
        {
            printf("%c", cad[i]);
        }
        printf("\n");
    }
}

```





Dame una palabra: Hola mundo cruel

SALIDA 1:

HOLA MUNDO CRUEL

SALIDA 2:

LEURC ODNUM ALOH

SALIDA 3:

H

O

L

A

M

U

N

D

O

C

R

U

E

L

SALIDA 4:

L

E

U

R

C

O

D

N

U

M

A

L

O

H

SALIDA 5:  
HOLA MUNDO CRUEL  
HOLA MUNDO CRUE  
HOLA MUNDO CRU  
HOLA MUNDO CR  
HOLA MUNDO C  
HOLA MUNDO  
HOLA MUNDO  
HOLA MUND  
HOLA MUN  
HOLA MU  
HOLA M  
HOLA  
HOLA  
HOL  
HO  
H

SALIDA 6:  
LEURC ODNUM ALOH  
LEURC ODNUM ALO  
LEURC ODNUM AL  
LEURC ODNUM A  
LEURC ODNUM  
LEURC ODNUM  
LEURC ODN  
LEURC ODN  
LEURC OD  
LEURC O  
LEURC  
LEURC  
LEUR  
LEU  
LE  
L

SALIDA 8:  
LEURC ODNUM ALOH  
EURC ODNUM ALOH  
URC ODNUM ALOH  
RC ODNUM ALOH  
C ODNUM ALOH  
ODNUM ALOH  
ODNUM ALOH  
DNUM ALOH  
NUM ALOH  
UM ALOH  
M ALOH  
ALOH  
ALOH  
LOH  
OH  
H

SALIDA 9:  
HL MND CRL

SALIDA 10:  
OAUQUE

```
#include "libreria.h"
void funcion1(char cad[]);
void funcion2(char cad[]);
void funcion3(char cad[]);
void funcion4(char cad[]);
void funcion5(char cad[]);
void funcion6(char cad[]);
void funcion7(char cad[]);
void funcion8(char cad[]);
void funcion9(char cad[]);
int main()
{
    system("CLS");
    char cad[30];
    int tam, i, j;
    printf("Dame una palabra: ");
    fflush(stdin);
    gets(cad);
    tam = tam_cad(cad);

    funcion1(cad);
    funcion2(cad);
    funcion3(cad);
    funcion4(cad);
    funcion5(cad);
    funcion6(cad);
    funcion7(cad);
    funcion8(cad);
    funcion9(cad);

    return 0;
}
void funcion1(char cad[])
{
    printf("MAYUSCULAS\n");
    mayusculas(cad);
    puts(cad);
    printf("\n");
}
void funcion2(char cad[])
{
    printf("minusculas\n");
    minusculas(cad);
    puts(cad);
    printf("\n");
}
```

```

}
void funcion3(char cad[])
{
    printf("Capital\n");
    capital(cad);
    puts(cad);
    printf("\n");
}
void funcion4(char cad[])
{
    int i;
    printf("Cantidad de caracteres\n");
    for (i = 0; cad[i] != '\0'; i++);
    printf("Tiene un total de ",i);
    printf(" caracteres\n");
    printf("\n");
}
void funcion5(char cad[])
{
    printf("Palabra al reves\n");
    reves(cad,cad);
    puts(cad);
    printf("\n");
}
void funcion6(char cad[])
{
    printf("Palabrassinespacio\n");
    sinespacios(cad);
    puts(cad);
    printf("\n");
}
void funcion7(char cad[])
{
    printf("S0l0 L3tr4s\n");
    solo_az(cad);
    puts(cad);
    printf("\n");
}
void funcion8(char cad[])
{
    printf("De ToD0\n");
    combo(cad);
    puts(cad);
    printf("\n");
}

```

```

void funcion9(char cad[])
{
    printf("Es un palindromo?\n");
    palindromo(cad,cad);
    printf("\n");
}

```

```

Dame una palabra: Hakuna matata
MAYUSCULAS
HAKUNA MATATA

minusculas
hakuna matata

Capital
hakuna matata

Cantidad de caracteres
Tiene un total de  caracteres

Palabra al reves
atatam matata

Palabrassinespacio
atatammatataa

S0l0 L3tr4s
atatammatataa

De ToDO
atatammatataa

Es un palindromo?
aatatamatataa
es un palindromo

```

```

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

void mayusculas(char cad[]);
void minusculas(char cad[]);
void capital(char cad[]);
int tam_cad(char cad[]);
void reves(char cad1[],char cad2[]);
void sinespacios(char cad[]);

```

```

void solo_az(char cad[]);
void combo(char cad[]);
void palindromo(char cad1[],char cad2[]);

void mayusculas(char cad[])
{
    int i, tam;
    tam = tam_cad(cad);
    for (i = 0; i < tam; i++)
    {
        if (cad[i] >= 'a' && cad[i] <= 'z')
        {
            cad[i] = cad[i] - ('a' - 'A');
        }
    }
}

void minusculas(char cad[])
{
    int i, tam;
    tam = tam_cad(cad);
    for (i = 0; i < tam; i++)
    {
        if (cad[i] >= 'A' && cad[i] <= 'Z')
        {
            cad[i] = cad[i] + ('a' - 'A');
        }
    }
}

void capital(char cad[])
{
    int i, tam;
    tam = tam_cad(cad);
    for (i = 0; i<tam; i++)
    {
        if (cad[i] != '\0')
        {
            if (cad[i] > cad[0])
            {
                if (cad[i] > '\0' && cad[i] >= 'A' && cad[i] <= 'Z');

                if (cad[i] >= 'A' && cad[i] <= 'Z')
                {

```

```

        cad[i] = tolower(cad[i]);
    }
}
else
{
    printf(" ");
}
}
}

```

```

int tam_cad(char cad[])
{
    int i;
    for (i = 0; cad[i] != '\0'; i++);

    return i;
}

```

```

void revez(char cad1[], char cad2[])
{
    int i, tam;
    tam = tam_cad(cad1);
    tam--;
    for (i = 0; i <= tam; i++)
    {
        cad2[i] = cad1[tam - i];
    }
}

```

```

void sinespacios(char cad[])
{
    int i,j;
    i=0;
    j=0;
    while (cad[i]!='\0')
    {
        if(cad[i]!=' ')
        {
            cad[j]=cad[i];
            j++;
        }
        i++;
    }
}

```



```

void solo_az(char cad[])
{
}

void combo(char cad[])
{
    int i, j, tam;
    tam = tam_cad(cad);

    for (i = 0; i < tam; i++)
    {
        if (cad[i] >= 'a' && cad[i] <= 'z')
        {
            cad[i] = cad[i] - ('a' - 'A');
        }
    }
    for (i = 0; i < tam; i++)
    {
        if (cad[i] >= 'A' && cad[i] <= 'Z')
        {
            cad[i] = cad[i] + ('a' - 'A');
        }
    }
    i=0;
    j=0;
    while (cad[i]!='\0')
    {
        if(cad[i]!=' ')
        {
            cad[j]=cad[i];
            j++;
        }
        i++;
    }
}

void palindromo(char cad1[], char cad2[])
{
    int i, j, tam, band;
    band=0;
    j=0;
    tam=tam_cad(cad1);
    reves(cad1, cad2);
    for (i=0; i<tam; i++)

```

```
{  
    if (cad1[i]==cad2[i])  
    {  
        j++;  
    }  
}  
if (j==tam)  
{  
    puts(cad1);  
    printf(" es un palindromo\n");  
}  
}
```

## **Conclusión**

En esta practica se llevo a cabo las distintas funciones para cambiar la estructura de una cadena, tanto si fuera una mayúscula esta pasarla a minúscula o viceversa.