

## **Ingeniero en Software y tecnologías emergentes**

**Materia:** Programación Estructurada / Clave 36276

**Alumno:** Solano Meza Angel Daniel

**Matrícula:** 372453

**Maestro:** Pedro Núñez Yépiz

**Actividad No.** : 7

**Tema - Unidad :** Cadenas de Caracteres - Unidad 1

**Ensenada Baja California a 9 de Septiembre del 2023**

## PARTE 1

### 1.1

```
void Salida1(void)
{
    // VARIABLES LOCALES
    int i, j;
    char cadena[100];
    system("CLS");
    // AQUI DESARROLLO PROGRAMA
    printf("    SALIDA 1\n");
    printf("Ingresa una frase: \n");
    fflush(stdin);
    gets(cadena);
    for (i = 0; cadena[i] != '\0'; i++) // Convierte a mayusculas
    {
        if (cadena[i] >= 'a')
        {
            if (cadena[i] <= 'z')
            {
                cadena[i] = cadena[i] - 32;
            }
        }
    }
    for (j = 0; j <= i; j++) // Imprime cadena
    {
        printf("%c", cadena[j]);
    }
    printf("\n");
    system("PAUSE");
}
```

```
SALIDA 1
Ingresa una frase:
ensenada
ENSENADA
Presione una tecla para continuar . . .
```

### 1.2

```

void Salida2(void)
{
    //  VARIABLES LOCALES
    int i, j;
    char cadena[100];
    system("CLS");
    //  AQUI DESARROLLO PROGRAMA
    printf("  SALIDA 2\n");
    printf("Ingresa una frase: \n");
    fflush(stdin);
    gets(cadena);
    for (i = 0; cadena[i] != '\0'; i++) // Convierte a mayusculas
    {
        if (cadena[i] >= 'a')
        {
            if (cadena[i] <= 'z')
            {
                cadena[i] = cadena[i] - 32;
            }
        }
    }
    for (j = i; j >= 0; j--)
    {
        printf("%c", cadena[j]); // Imprime la cadena al reves
    }
    printf("\n");
    system("PAUSE");
}

```

```

SALIDA 2
Ingresa una frase:
ensenada
ADANESNE
Presione una tecla para continuar . . .

```

```

void Salida3(void)
{
    // VARIABLES LOCALES
    int i, j;
    char cadena[100];
    system("CLS");
    // AQUI DESARROLLO PROGRAMA
    printf("    SALIDA 3\n");
    printf("Ingresa una frase: \n");
    fflush(stdin);
    gets(cadena);

    for (i = 0; cadena[i] != '\0'; i++) // Convierte a mayusculas
    {
        if (cadena[i] >= 'a')
        {
            if (cadena[i] <= 'z')
            {
                cadena[i] = cadena[i] - 32;
            }
        }
    }
    for (j = 0; j <= i; j++)
    {
        printf("%c\n", cadena[j]); // Imprime la cadena letra por letra hacia abajo
    }
    system("PAUSE");
}

```

```

SALIDA 3
Ingresa una frase:
ensenada
E
N
S
E
N
A
D
A

Presione una tecla para continuar . . .

```

```

void Salida4(void)
{
    // VARIABLES LOCALES
    int i, j;
    char cadena[100];
    system("CLS");
    // AQUI DESARROLLO PROGRAMA
    printf("    SALIDA 4\n");
    printf("Ingresa una frase: \n");
    fflush(stdin);
    gets(cadena);
    for (i = 0; cadena[i] != '\0'; i++) // Convierte a mayusculas
    {
        if (cadena[i] >= 'a')
        {
            if (cadena[i] <= 'z')
            {
                cadena[i] = cadena[i] - 32;
            }
        }
    }
    for (j = i; j >= 0; j--)
    {
        printf("%c\n", cadena[j]); // Imprime la cadena letra por letra al revés
    }
    system("PAUSE");
}

```

```

SALIDA 4
Ingresa una frase:
ensenada

```

```

A
D
A
N
E
S
N
E
Presione una tecla para continuar . . .

```

```

void Salida5(void)
{
    // VARIABLES LOCALES
    int i, j, k, h;
    char cadena[100];
    system("CLS");
    // AQUI DESARROLLO PROGRAMA
    printf("  SALIDA 5\n");
    printf("Ingresa una frase: \n");
    fflush(stdin);
    gets(cadena);

    for (i = 0; cadena[i] != '\0'; i++) // Convierte a mayusculas
    {
        if (cadena[i] >= 'a')
        {
            if (cadena[i] <= 'z')
            {
                cadena[i] = cadena[i] - 32;
            }
        }
    }
    k = i;
    for (h = i; h >= 0; h--) // Controla las veces que se imprimira la cadena
    {
        j = 0;
        cadena[k] = '\0';          // Modifica el carecter vacio
        for (j = 0; j <= i; j++) // Imprime la cadena en mayuscula en el orden del usuario
        {
            printf("%c", cadena[j]);
        }
        k--;
        printf("\n");
    }
    system("PAUSE");
}

```

```

SALIDA 5
Ingresa una frase:
ensenada
ENSENADA
ENSENAD
ENSENA
ENSEN
ENSE
ENS
EN
E

Presione una tecla para continuar . . .

```

```

void Salida6(void)
{
    // VARIABLES LOCALES
    int i, j, k, h;
    char cadena[100];
    system("CLS");
    // AQUI DESARROLLO PROGRAMA
    printf("    SALIDA 6\n");
    printf("Ingresa una frase: \n");
    fflush(stdin);
    gets(cadena);

    for (i = 0; cadena[i] != '\0'; i++)
    {
        if (cadena[i] >= 'a')
        {
            if (cadena[i] <= 'z')
            {
                cadena[i] = cadena[i] - 32;
            }
        }
    }
    k = 0;
    for (h = i; h >= 0; h--) // Controla las veces que se imprimira la cadena
    {
        for (j = i; j >= 0; j--) // Imprime la cadena
        {
            printf("%c", cadena[j]);
        }
        cadena[k] = '\0'; // Modifica el carecter vacio
        k++;
        printf("\n");
    }
    system("PAUSE");
}

```

```

SALIDA 6
Ingresa una frase:
ensenada
ADANESNE
ADANESN
ADANES
ADANE
ADAN
ADA
AD
A

Presione una tecla para continuar . . .

```

```

void Salida7(void)
{
    // VARIABLES LOCALES
    int i, j, k, h;
    char cadena[100];
    system("CLS");
    // AQUI DESARROLLO PROGRAMA
    printf("    SALIDA 7\n");
    printf("Ingresa una frase: \n");
    fflush(stdin);
    gets(cadena);

    for (i = 0; cadena[i] != '\0'; i++) // Convierte a mayusculas
    {
        if (cadena[i] >= 'a')
        {
            if (cadena[i] <= 'z')
            {
                cadena[i] = cadena[i] - 32;
            }
        }
    }
    k = 0;
    for (h = i; h >= 0; h--) // Controla las veces que se imprimira la cadena
    {
        j = 0;
        for (j = 0; j <= i; j++) // Imprime la cadena
        {
            printf("%c", cadena[j]);
        }
        cadena[k] = '\0'; // Modifica el carecter vacio

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

```

```

Ingresa una frase:
ensenada
ENSENADA
NSENADA
SENADA
ENADA
NADA
ADA
DA
A

```



```

void Salida8(void)
{
    // VARIABLES LOCALES
    int i, j, k, h;
    char cadena[100];
    system("CLS");
    // AQUI DESARROLLO PROGRAMA
    printf("    SALIDA 8\n");
    printf("Ingresa una frase: \n");
    fflush(stdin);
    gets(cadena);

    for (i = 0; cadena[i] != '\0'; i++) // Convierte a mayusculas
    {
        if (cadena[i] >= 'a')
        {
            if (cadena[i] <= 'z')
            {
                cadena[i] = cadena[i] - 32;
            }
        }
    }
    k = i;
    for (h = i; h >= 0; h--) // Controla las veces que se imprimira la cadena
    {
        j = 0;
        cadena[k] = '\0'; // Modifica el carecter vacio
        for (j = i; j >= 0; j--) // Imprime la cadena al revés
        {
            printf("%c", cadena[j]);
        }
        k--;
        printf("\n");
    }
}

```

```

SALIDA 8
Ingresa una frase:
ensenada
ADANESNE
DANESNE
ANESNE
NESNE
ESNE
SNE
NE
E

```

```

Presione una tecla para continuar . . .

```

```

void Salida9(void)
{
    // VARIABLES LOCALES
    int i, j;
    char cadena[100];
    system("CLS");
    // AQUI DESARROLLO PROGRAMA
    printf("    SALIDA 9\n");
    printf("Ingresa una frase: \n");
    fflush(stdin);
    gets(cadena);

    for (i = 0; cadena[i] != '\0'; i++) // Convierte a mayusculas
    {
        if (cadena[i] >= 'a')
        {
            if (cadena[i] <= 'z')
            {
                cadena[i] = cadena[i] - 32;
            }
        }
    }

    for (j = 0; j <= i; j++)
    {
        // Si el caracter no es vocal la imprime
        if ((cadena[j] != 'A') && (cadena[j] != 'E') && (cadena[j] != 'I') && (cadena[j] != 'O') && (cadena[j] != 'U'))
        {
            printf("%c", cadena[j]);
        }
    }
    printf("\n");
    system("PAUSE");
}

```

```

SALIDA 9
Ingresa una frase:
ensenada
NSND
Presione una tecla para continuar . . .

```

```

void Salida10(void)
{
    // VARIABLES LOCALES
    int i, j;
    char cadena[100];
    system("CLS");
    // AQUI DESARROLLO PROGRAMA
    printf("    SALIDA 10\n");
    printf("Ingresa una frase: \n");
    fflush(stdin);
    gets(cadena);

    for (i = 0; cadena[i] != '\0'; i++) // Convierte a mayusculas
    {
        if (cadena[i] >= 'a')
        {
            if (cadena[i] <= 'z')
            {
                cadena[i] = cadena[i] - 32;
            }
        }
    }

    for (j = 0; j <= i; j++)
    {
        // Si el caracter es vocal lo imprime
        if ((cadena[j] == 'A') || (cadena[j] == 'E') || (cadena[j] == 'I') || (cadena[j] == 'O') || (cadena[j] == 'U'))
        {
            printf("%c", cadena[j]);
        }
    }
    printf("\n");
    system("PAUSE");
}

```

```

SALIDA 10
Ingresa una frase:
ensenada
EEAA
Presione una tecla para continuar . . .

```

## PARTE 2

### FUNCION MENU

```
void menu()
{
    int op, largo;
    char cadena[100];
    do
    {
        op = msges();
        switch (op)
        {
            case 1:
                system("CLS");
                printf("    MAYUSCULAS\n");
                printf("Ingresa una sentencia:\n");
                fflush(stdin);
                gets(cadena);
                Mayusculas(cadena);
                break;
            case 2:
                system("CLS");
                printf("    MINUSCULAS\n");
                printf("Ingresa una sentencia:\n");
                fflush(stdin);
                gets(cadena);
                Minusculas(cadena);
                break;
```

```
case 3:
    system("CLS");
    printf("    CAPITAL\n");
    printf("Ingresa una sentencia:\n");
    fflush(stdin);
    gets(cadena);
    Capital(cadena);
    break;
case 4:
    system("CLS");
    printf("    LARGO CADENA\n");
    printf("Ingresa una sentencia:\n");
    fflush(stdin);
    gets(cadena);
    largo = Caracteres(cadena);
    printf("La cadena tiene %d caracteres.\n", largo);
    break;
case 5:
    system("CLS");
    printf("    CADENA INVERSA\n");
    printf("Ingresa una sentencia:\n");
    fflush(stdin);
    gets(cadena);
    Inversa(cadena);
    break;
case 6:
    system("CLS");
    printf("    SIN ESPACIOS\n");
    printf("Ingresa una sentencia:\n");
    fflush(stdin);
    gets(cadena);
    Espacios(cadena);
    break;
```

```

        case 7:
            system("CLS");
            printf("    SOLO CARACTERES A - Z\n");
            printf("Ingresa una sentencia:\n");
            fflush(stdin);
            gets(cadena);
            Alfabetica(cadena);
            break;
        case 8:
            system("CLS");
            printf("    COMBINACION\n");
            printf("Ingresa una sentencia:\n");
            fflush(stdin);
            gets(cadena);
            Todas(cadena);
            break;
        case 9:
            system("CLS");
            printf("    PALINDROMO\n");
            printf("Ingresa una sentencia:\n");
            fflush(stdin);
            gets(cadena);
            Palindromo(cadena);
            break;
    }
} while (op != 0);
}

```

```

//
void Mayusculas(char cadena[])
{
    // VARIABLES LOCALES
    int i, j;
    // AQUI DESARROLLO PROGRAMA
    i = 0;
    for (i = 0; cadena[i] != '\0'; i++) // Convierte a mayusculas
    {
        if (cadena[i] >= 'a')
        {
            if (cadena[i] <= 'z')
            {
                cadena[i] = cadena[i] - 32;
            }
        }
    }
    for (j = 0; j <= i; j++) // Imprime la cadena
    {
        printf("%c", cadena[j]);
    }
    printf("\n");
    system("PAUSE");
}

```

MAYUSCULAS

Ingresa una sentencia:

Hola

HOLA

Presione una tecla para continuar . . . █

```

void Minusculas(char cadena[])
{
    // VARIABLES LOCALES
    int i, j;
    // AQUI DESARROLLO PROGRAMA
    i = 0;
    for (i = 0; cadena[i] != '\0'; i++) // Convierte a minusculas
    {
        if (cadena[i] >= 'A')
        {
            if (cadena[i] <= 'Z')
            {
                cadena[i] = cadena[i] + 32;
            }
        }
    }
    for (j = 0; j <= i; j++) // Imprime la cadena
    {
        printf("%c", cadena[j]);
    }
    printf("\n");
    system("PAUSE");
}

```

```

MINUSCULAS
Ingresa una sentencia:
HOLA
hola
Presione una tecla para continuar . . . 

```



```

void Capital(char cadena[])
{
    // VARIABLES LOCALES
    int i, j;
    // AQUI DESARROLLO PROGRAMA
    for (i = 0; cadena[i] != '\0'; i++) // Cuenta los caracteres totales
    {
        if (i == 0) // Convierte a mayusculas el primer caracter
        {
            if (cadena[i] != ' ')
            {
                if (cadena[i] >= 'a')
                {
                    if (cadena[i] <= 'z')
                    {
                        cadena[i] = cadena[i] - 32;
                    }
                }
            }
        }
        else // Si empieza con un espacio convierte la segunda letra en mayuscula
        {
            if (cadena[i + 1] >= 'a')
            {
                if (cadena[i + 1] <= 'z')
                {
                    cadena[i + 1] = cadena[i + 1] - 32;
                }
            }
        }
    }
    if (cadena[i] == ' ')

```

```

    {
        i++;
        if (cadena[i] >= 'a')
        {
            if (cadena[i] <= 'z')
            {
                cadena[i] = cadena[i] - 32;
            }
        }
    }
}
for (j = 0; j <= i; j++) // Imprime la cadena
{
    printf("%c", cadena[j]);
}
printf("\n");
system("PAUSE");
}

```

```
CAPITAL
Ingresa una sentencia:
hola como estas
Hola Como Estas
Presione una tecla para continuar . . .
```

2.4

```
int Caracteres(char cadena[])
{
    // VARIABLES LOCALES
    int i;
    // AQUI DESARROLLO PROGRAMA
    for (i = 0; cadena[i] != '\0'; i++) // Cuenta caracteres
    {
    }
    return i;
}
```

```
LARGO CADENA
Ingresa una sentencia:
Hola como estas
La cadena tiene 15 caracteres.
Presione una tecla para continuar . . .
```

2.5

```
void Inversa(char cadena[])
{
    // VARIABLES LOCALES
    int i, j;
    // AQUI DESARROLLO PROGRAMA
    for (i = 0; cadena[i] != '\0'; i++) // Cuenta caracteres
    {
    }

    for (j = i; j >= 0; j--) // Imprime la cadena inversa
    {
        printf("%c", cadena[j]);
    }
    printf("\n");
    system("PAUSE");
}
```

```
CADENA INVERSA
Ingresa una sentencia:
Hola como estas
satse omoc aloH
Presione una tecla para continuar . . .
```

## 2.6

```

void Espacios(char cadena[])
{
    // VARIABLES LOCALES
    int i, j, k, h, q;
    char copia[100];
    // AQUI DESARROLLO PROGRAMA
    for (i = 0; cadena[i] != '\0'; i++) // Cuenta caracteres
    {
    }

    for (q = 0; q <= i; q++) // Copia la cadena original
    {
        copia[q] = cadena[q];
    }

    for (j = 0; j <= i; j++)
    {
        if (copia[j] == ' ') // Si hay un espacio vacio, todo el contenido se recorre un espacio a la izquierda
        {
            for (k = j; k < i; k++)
            {
                copia[k] = copia[k + 1];
            }
        }
    }

    for (h = 0; h <= i; h++) // Imprime la cadena
    {
        printf("%c", copia[h]);
    }
    printf("\n");
    system("PAUSE");
}

```

SIN ESPACIOS

Ingresa una sentencia:

Hola como estas

Holacomoestas

Presione una tecla para continuar . . .

## 2.7

```

void Alfabetica(char cadena[])
{
    // VARIABLES LOCALES
    int i, j;
    char copia[100];
    // AQUI DESARROLLO PROGRAMA
    for (i = 0, j = 0; cadena[i] != '\0'; i++, j++)
    {
        // Verifica si es caracter o espacio
        if ((cadena[i] == ' ') || (cadena[i] >= 'a' && cadena[i] <= 'z') || (cadena[i] >= 'A' && cadena[i] <= 'Z'))
        {
            // Verifica inicio y terminacion en espacio
            if (cadena[i] != ' ' || (i > 0 && cadena[i + 1] != '\0' && cadena[i - 1] != ' '))
            {
                copia[j] = cadena[i];
                printf("%c", copia[j]);
            }
        }
    }
    printf("\n");
    system("PAUSE");
}

```

```
SOLO CARACTERES A - Z
Ingresa una sentencia:
56 Hola como estas
    Hola como estas
Presione una tecla para continuar . . .
```

2.8

```
void Todas(char cadena[])
{
    printf("    MAYUSCULAS\n");
    Mayusculas(cadena);
    printf("    MINUSCULAS\n");
    Minusculas(cadena);
    printf("    CAPITAL\n");
    Capital(cadena);
    printf("    SIN ESPACIOS\n");
    Espacios(cadena);
    printf("    AL REVES\n");
    Inversa(cadena);
    system("PAUSE");
}
```

```
COMBINACION
Ingresa una sentencia:
Hola como estas
    MAYUSCULAS
HOLA COMO ESTAS
Presione una tecla para continuar . . .
    MINUSCULAS
hola como estas
Presione una tecla para continuar . . .
    CAPITAL
Hola como estas
Presione una tecla para continuar . . .
```

```
SIN ESPACIOS
Holacomoestas
Presione una tecla para continuar . . .
    AL REVES
satse omoc aloH
Presione una tecla para continuar . . .
```

2.9

```

void Palindromo(char cadena[])
{
    // VARIABLES LOCALES
    int largo, j, valor, i, palindromo;
    // AQUI DESARROLLO PROGRAMA
    valor = Validar(cadena);
    if (valor == 0)
    {
        printf("Cadena valida\n");
        largo = 0;
        while (cadena[largo] != '\0')
        {
            largo++;
        }

        i = 0;
        j = largo - 1;
        palindromo = 1;
        while (i < j)
        {
            if (cadena[i] != cadena[j])
            {
                palindromo = 0;
            }
            i++;
            j--;
        }
        if (palindromo == 1)
        {
            printf("La cadena es palindromo\n");
        }
        else
    }
}

```

```

printf("La cadena no es palindromo\n");
system("PAUSE");
}
else
{
printf("Cadena invalida.\n");
system("PAUSE");
}
}
//*****
int Validar(char cadena[])
{
for (int i = 0; cadena[i] != '\0'; i++)
{
if (cadena[i] >= '0' && cadena[i] <= '9')
{
return 1; // Son numeros
}
if (cadena[i] >= 'a' && cadena[i] <= 'z')
{
return 2; // Es minuscula
}
if (cadena[i] == ' ' && cadena[i + 1] == ' ')
{
return 3; // Hay doble espacio
}
}
return 0;
}

```

#### PALINDROMO

Ingresa una sentencia:

ANA

Cadena valida

La cadena es palindromo

Presione una tecla para continuar . . .