



Universidad Autónoma de Baja California

Facultad de Ingeniería, Arquitectura y Diseño

Ingeniero en Computación

Asignatura:

Programación Estructurada

Actividad 7:

Brayan Arturo Rocha Meneses

Matricula:

371049

Ensenada Baja California 25 de Septiembre del 2023

```

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

int msges();
void menu();

void datos(char nombre[]);
int validar(char mensj[], int ri, int rf);

void salida1(char nombre[]);
void salida2(char nombre[]);
void salida3(char nombre[]);
void salida4(char nombre[]);
void salida5(char nombre[]);
void salida6(char nombre[]);
void salida7(char nombre[]);
void salida8(char nombre[]);
void salida9(char nombre[]);
void salida10(char nombre[]);

int main()
{
    menu();
    return 0;
}

```

```

int msges()
{
    int op;

    system ("CLS");
    printf("\n MENU PARA LAS DIFERENTES SALIDAS \n");
    printf("\n OPCIONES: \n");
    printf("1.- MAYUSCULAS\n");
    printf("2.- MAYUSCULAS AL REVES\n");
    printf("3.- UNA LETRA POR RENGLO\n");
    printf("4.- UNA LETRA POR RENGLO AL REVES\n");
    printf("5.- PALABRA MENOS UNA LETRA\n");
    printf("6.- LA PALABRA MENOS UNA LETRA AL REVES\n");
    printf("7.- LA PALABRA MENOS UNA LETRA DEL INICIO\n");
    printf("8.- LA PALABRA MENOS UNA LETRA DEL INICIO AL REVES\n");
    printf("9.- SOLO CONSONANTES\n");
    printf("10.-SOLO VOCALES\n");
    printf("11.- SALIR \n");
    op = validar ("ESCOGE UNA OPCION: ", 0, 10);

    return op;
}

void menu()
{
    system ("CLS");
    int op;
    char nombre[50];
    do {
        op = msges();
        switch(op)

```

```

{
    case 1:
        datos(nombre);
        salida1(nombre);
        break;
    case 2:
        datos(nombre);
        salida2(nombre);
        break;
    case 3:
        datos(nombre);
        salida3(nombre);
        break;
    case 4:
        datos(nombre);
        salida4(nombre);
        break;
    case 5:
        datos(nombre);
        salida5(nombre);
        break;
    case 6:
        datos(nombre);
        salida6(nombre);
        break;
    case 7:
        datos(nombre);
        salida7(nombre);
        break;
    case 8:
        datos(nombre);
        salida8(nombre);
        break;
}

```

```

    case 9:
        datos(nombre);
        salida9(nombre);
        break;
    case 10:
        datos(nombre);
        salida10(nombre);
        break;
    case 0:
        printf("Saliste del programa.\n");
        break;
    default:
        printf("Opcion Invalida.\n");
        break;
}
} while(cp != 0);
system("PAUSE");
}

```

```

int validar(char mensj[], int ri, int rf)
{
    int num;

    char cadena[50];

    do
    {
        printf("%s", mensj);
        fflush(stdin);
        gets(cadena);
    }
    while (num != 0);
}

```

```

        num = atoi(cadena);
    } while (num < ri || num > rf);

    return num;
}

void datos(char nombre[])
{
    system("CLS");
    printf("Introduzca su nombre: ");
    fflush(stdin);
    gets(nombre);

    if (nombre[strlen(nombre) - 1] == '\n')
    {
        nombre[strlen(nombre) - 1] = '\0';
    }
}

void salida1(char nombre[])
{
    system("CLS");

    printf("Salida 1: MAYUSCULAS\n");

    for (int i = 0; nombre[i] != '\0'; i++)
    {
        if (nombre[i] >= 'a')
        {

```

```

            if (nombre[i] <= 'z')
            {
                nombre[i] = nombre[i] - ('a' - 'A');
            }
        }

        printf("%c", nombre[i]);
    }
    printf("\n");
    system("PAUSE");
}

void salida2(char nombre[])
{
    system("CLS");

    printf("Salida 2: MAYUSCULAS AL REVES\n");

    int length = strlen(nombre);

    for (int i = length - 1; i >= 0; i--)
    {
        if (nombre[i] >= 'a')
        {
            if (nombre[i] <= 'z')
            {
                nombre[i] = nombre[i] - ('a' - 'A');
            }
        }

        printf("%c", nombre[i]);
    }
}

```

```

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

void salida3(char nombre[])
{
    system ("CLS");

    printf("Salida 3: UNA LETRA POR RENGLON\n");

    for (int i = 0; nombre[i] != '\0'; i++)
    {
        if (nombre[i] >= 'a')
        {
            if (nombre[i] <= 'z')
            {
                nombre[i] = nombre[i] - ('a' - 'A');
            }
        }

        printf("%c\n", nombre[i]);
    }

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

void salida4(char nombre[])
{
    system ("CLS");

```

```

    int length = strlen(nombre);

    for (int i = length - 1; i >= 0; i--)
    {
        if (nombre[i] >= 'a')
        {
            if (nombre[i] <= 'z')
            {
                nombre[i] = nombre[i] - ('a' - 'A');
            }
        }

        printf("%c\n", nombre[i]);
    }

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

void salida5(char nombre[])
{
    system ("CLS");

    printf("Salida 5: LA PALABRA MENOS UNA LETRA\n");

    int length = strlen(nombre);
    char copia_nombre[50];

    strcpy(copia_nombre, nombre);

    for (int i = length; i >= 0; i--)
    {

```

```

        {
            if (copia_nombre[j] <= 'z')
            {
                copia_nombre[j] = copia_nombre[j] - ('a' - 'A');
            }
        }

        printf("%c", copia_nombre[j]);
    }

    printf("\n");
}

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

```

```

void salida6(char nombre[])
{
    system ("CLS");
    printf("Salida 6: LA PALABRA MENOS UNA LETRA AL REVES\n");

    int length = strlen(nombre);
    char copia_nombre[50];

    strcpy(copia_nombre, nombre);

    for (int i = length - 1; i > 0; i--)
    {
        copia_nombre[i] = '\0';
    }
}

```

```

    }
}

for (int i = 0; nombre[i] != '\0'; i++)
{
    if ( nombre[i] == 'A')
    {
        printf ( "%c", nombre[i] );
    }
    if (nombre[i] == 'E')
    {
        printf ( "%c", nombre[i] );
    }
    if ( nombre[i] == 'I')
    {
        printf ( "%c", nombre[i] );
    }
    if ( nombre[i] == 'O')
    {
        printf ( "%c", nombre[i] );
    }
    if ( nombre[i] == 'U')
    {
        printf ( "%c", nombre[i] );
    }
}

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

```

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

int msges();
void menu();

int es_numero(char character);

int es_espacio(char character);

int validarCadena(const char *cadena);

void datos(char cadena[]);

void ej1_pt2(char cadena[]);
void ej2_pt2(char cadena[]);
void ej3_pt2(char cadena[]);
void ej4_pt2(char cadena[]);
void ej5_pt2(char cadena[]);
void ej6_pt2(char cadena[]);
void ej7_pt2(char cadena[]);
void ej8_pt2(char cadena[]);
void ej9_pt2(char cadena[]);

int main()
{
    menu();
```

```
    return 0;
}

int msges()
{
    int op;

    system("CLS");
    printf("\n MENU \n");
    printf("\n PARTE 2 \n");
    printf("1.- Mayusculas \n");
    printf("2.- Minusculas \n");
    printf("3.- Capital \n");
    printf("4.- Cantidad \n");
    printf("5.- Inversa \n");
    printf("6.- Nueva Cadena \n");
    printf("7.- Alfabeto \n");
    printf("8.- TODO \n");
    printf("9.- PALINDROMO \n");
    printf("10.- SALIR \n");
    printf("SELECCIONA UNA OPCION: ");
    scanf("%d",&op);

    return op;
}

void menu()
{
    int op;
    char cadena[50];
    do{
```

```
op=msges();
switch (op)
{
    case 1:
        datos(cadena);
        ej1_pt2(cadena);
        break;
    case 2:
        datos(cadena);
        ej2_pt2(cadena);
        break;
    case 3:
        datos(cadena);
        ej3_pt2(cadena);
        break;
    case 4:
        datos(cadena);
        ej4_pt2(cadena);
        break;
    case 5:
        datos(cadena);
        ej5_pt2(cadena);
        break;
    case 6:
        datos(cadena);
        ej6_pt2(cadena);
        break;
    case 7:
        datos(cadena);
        ej7_pt2(cadena);
        break;
    case 8:
        datos(cadena);
```



```

int validarCadena(const char *cadena)
{
    int contieneNumero = 0;
    int contieneDobleEspacio = 0;

    for (int i = 0; cadena[i] != '\0'; i++)
    {
        if (es_numero(cadena[i]))
        {
            contieneNumero = 1;
        }
        if (es_espacio(cadena[i]))
        {
            if (cadena[i + 1] == ' ')
            {
                contieneDobleEspacio = 1;
            }
        }
    }

    if (contieneNumero || contieneDobleEspacio)
    {
        return 0;
    }
    else
    {
        return 1;
    }
}

```

```

int es_numero(char caracter)
{
    return (caracter >= '0' && caracter <= '9');
}

int es_espacio(char caracter)
{
    return (caracter == ' ');
}

void datos(char cadena[])
{
    system("CLS");
    printf( "Introduzca una palabra: " );
    fflush(stdin);
    gets(cadena);

    if (!validarCadena(cadena))
    {
        printf("La cadena no cumple con los requisitos.\n");
        printf("No uses dobles espacios y no ingrese numeros.\n");

        cadena[0] = '\0';
        system("PAUSE");
    }
}

if (cadena[0] != '\0')

```

```

        system ("CLS");

        printf("CONVERTIR A MAYUSCULAS\n");

        for (int i = 0; cadena[i] != '\0'; i++)
        {

            if (cadena[i] >= 'a')
            {
                if (cadena[i] <= 'z')
                {

                    cadena[i] = cadena[i] - ('a' - 'A');

                }

            }

            printf("%c", cadena[i]);

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

void ej2_pt2(char cadena[])
{
    if (cadena[0] != '\0')
    {
        system ("CLS");

        printf("CONVERTIR A MINUSCULAS\n");

        for (int i = 0; cadena[i] != '\0'; i++)

```

```

void ej3_pt2(char cadena[])
{
    if (cadena[0] != '\0')
    {
        system ("CLS");

        printf("CONVERTIR A CAPITAL\n");

        int primera_letra = 1;

        for (int i = 0; cadena[i] != '\0'; i++)
        {

            if (cadena[i] >= 'a')
            {
                if (cadena[i] <= 'z')
                {

                    if (primera_letra)
                    {

                        cadena[i] = cadena[i] - ('a' - 'A');

                        primera_letra = 0;

                    }

                }

            }

            printf("%c", cadena[i]);

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

```

```
        system("PAUSE");
    }

void ej6_pt2(char cadena[])
{
    system("CLS");
    printf("hola");
    system("PAUSE");
}

void ej7_pt2(char cadena[])
{
    system("CLS");
    printf("hola");
    system("PAUSE");
}

void ej8_pt2(char cadena[])
{
    system("CLS");
    printf("hola");
    system("PAUSE");
}

void ej9_pt2(char cadena[])
{
    system("CLS");
    printf("hola");
    system("PAUSE");
}
```