

## U2T1\_PERSONAL

### 1. DATOS INFORMATIVOS

**Carrera:** Electrónica y Automatización

**Asignatura:** Fundamentos de Programación

**Tema del taller:** Uso de SWITCH

**Docente:** Ing. Jenny Ruíz

**Integrantes:** Mendoza Cortez Christopher Alexander

**Fecha:** 20/11/2025      **NRC:** 29583

### 2. DESARROLLO

#### Objetivo

Comprender el uso de la estructura de selección múltiple SWITCH en C utilizando condiciones anidadas, basado en el código proporcionado.

#### Introducción

En este taller el estudiante analizará un programa en C que identifica si un carácter ingresado es vocal, consonante, número o símbolo. Se utilizará la estructura SWITCH como herramienta principal para clasificar caracteres.

#### Código base de análisis

```
#include <stdio.h>

void main()
{
    char car;
    printf("Ingrese caracter: ");
    scanf("%c",&car);
    if
    (car>='A' && car<='Z')
    {
        switch(car)
        {
            case 'A': case 'E': case 'I':
            case 'O': case 'U': printf("Es vocal mayuscula\n");
            break;
            default: printf("Es consonante mayuscula\n");
            break;
        }
    }
}
```

```

}
else
{
    if (car>='a' && car<='z')
    {
        switch(car)
        {
            case 'a': case 'e': case 'i':
            case 'o': case 'u': printf("Es vocal minuscula\n");
        break;
            default: printf("Es consonante minuscula\n");
        break;
        }
    }
    else
    {
        if (car>='0' && car<='9')
printf("Es un numero\n");      else
            printf("Es un simbolo\n");
        }
    }
}

```

## Actividades del Taller

Actividad 1: Explique en sus propias palabras cómo funciona el SWITCH dentro del programa.

El programa utiliza el switch para **examinar** la variable car (el carácter ingresado) y compararla directamente con una lista de valores constantes (case).

Actividad 2: Modifique el programa para que también identifique vocales acentuadas (á, é, í, ó, ú).

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

```

void main() {

    char car;

    printf("Ingrese caracter: ");

    scanf("%c", &car);

    // Verificar si es letra mayúscula
    if ((car >= 'A' && car <= 'Z') || (car == 'Á' || car == 'É' || car == 'Í' || car == 'Ó' || car == 'Ú')) {

        switch (car) {

            case 'A': case 'E': case 'I': case 'O': case 'U':

            case 'Á': case 'É': case 'Í': case 'Ó': case 'Ú':

                printf("Es vocal mayúscula\n");
        }
    }
}
```

```

break;

default:

printf("Es consonante mayúscula\n");

break;

}

}

// Verificar si es letra minúscula

else if ((car >= 'a' && car <= 'z') || (car == 'á' || car == 'é' || car == 'í' || car == 'ó' || car == 'ú')) {

switch (car) {

case 'a': case 'e': case 'i': case 'o': case 'u':

case 'á': case 'é': case 'í': case 'ó': case 'ú':

printf("Es vocal minúscula\n");

break;

default:

printf("Es consonante minúscula\n");

break;

}

}

// Verificar si es número

else if (car >= '0' && car <= '9') {

printf("Es un número\n");

}

// Si no es letra ni número, es símbolo

else {

printf("Es un símbolo\n");

}

}

```

### Enunciado del ejercicio

Cree una versión del programa donde el SWITCH identifique colores primarios: R, G, B.



```
#include <stdio.h>

int main() {
    char color;
    printf("Ingrese una letra (R, G, B): ");
    scanf(" %c", &color); // Nota: el espacio antes de %c evita problemas con saltos de línea

    switch (color) {
        case 'R':
        case 'r':
            printf("Rojo\n");
            break;
        case 'G':
        case 'g':
            printf("Verde\n");
            break;
        case 'B':
        case 'b':
            printf("Azul\n");
            break;
        default:
            printf("Color no válido. Use R, G o B.\n");
            break;
    }

    return 0;
}
```

### 3. Preguntas de reflexión

#### 1. ¿Por qué es útil el SWITCH en lugar de múltiples if-else?

El switch es mucho más útil ya que nos permite ordenar mejor el código, y de esta forma, evitamos usar muchos if - else, mejorando la legibilidad del código.

#### 2. ¿Qué sucede si no incluimos la instrucción break en cada caso?

Si no incluimos el break en cada caso, el programa se va a seguir ejecutando, incluso los casos que no correspondan.

#### 3. ¿Cómo se comporta el programa si el usuario ingresa más de un carácter?

El programa solo va a tomar el 1er carácter ingresado, y los que le siguen los va a ignorar

### 4. Mini ejercicio práctico

Escriba un programa en C utilizando SWITCH que permita ingresar un número del 1 al 7 y muestre el día de la semana correspondiente.

```
//MINI EJERCICIO DE LA SEMANA
```

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



```
int main() {
    int numero;
    printf("Ingrese un número del 1 al 7: ");
    scanf("%d", &numero);

    switch (numero) {
        case 1:
            printf("Lunes\n");
            break;
        case 2:
            printf("Martes\n");
            break;
        case 3:
            printf("Miércoles\n");
            break;
        case 4:
            printf("Jueves\n");
            break;
        case 5:
            printf("Viernes\n");
            break;
        case 6:
            printf("Sábado\n");
            break;
        case 7:
            printf("Domingo\n");
            break;
        default:
            printf("Número inválido. Debe ser entre 1 y 7.\n");
            break;
    }

    return 0;
}
```

## 5. RUBRICA

Criterio	Descripción del desempeño esperado	Puntaje
Comprensión del funcionamiento del SWITCH	Explica con claridad en sus propias palabras cómo opera la estructura SWITCH dentro del programa base,	0 – 5 pts



	incluyendo casos, break y flujo lógico.	
Modificación del programa para incluir vocales acentuadas	Integra correctamente los caracteres á, é, í, ó, ú en la lógica del programa, manteniendo coherencia, sintaxis y funcionamiento.	0 – 5 pts
Versión alternativa del programa (R, G, B)	Implementa un nuevo SWITCH funcional que identifique colores primarios (R, G, B). Se evalúa creatividad, sintaxis y correcto uso de casos.	0 – 4 pts
Respuesta a preguntas de reflexión	Responde con criterio propio y claridad conceptual a las tres preguntas de reflexión del taller, demostrando comprensión real del tema.	0 – 3 pts
Mini ejercicio: días de la semana	Programa en C un SWITCH que muestre el día de la semana según un número del 1 al 7. El código debe compilar y mostrar todos los casos de forma ordenada.	0 – 3 pts

#### 4. REFERENCIAS

##### Referencias

Joyanes Aguilar, L. (2013). *Fundamentos de programación: algoritmos, estructuras de datos y objetos*.

Madrid: McGraw-Hill.