

## Taller de Fundamentos de Programación – Uso de SWITCH

Nombre: Dylan Tutillo

Fecha: 19/11/2025

NRC: 29583

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

### 1. 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.

### 2. 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");
}
}
}

```

### 3. Actividades del Taller

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

**Res:** En este algoritmo se usan dos switches, una para letras mayúsculas y otra para letras minúsculas. Cada switch tiene si o si un if que verifica el tipo de carácter. En el primer switch si el carácter que se ingresa es una letra mayúscula, el switch compara ese carácter con las vocales mayúsculas. Si llega a coincidir, muestra por pantalla que es una vocal y si no cumple es una consonante mayúscula, todo gracias al default. Y así también sucede con las minúsculas, si el carácter que se ingresa es una letra minúscula, el switch compara ese carácter con las vocales minúsculas. Si llega a coincidir, muestra por pantalla que es una vocal y si no cumple es una consonante minúscula, todo gracias al default.

**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);

    if (car >= 'A' && car <= 'Z')

    {

```

```
switch(car)
{
    case 'A': case 'Á':
    case 'E': case 'É':
    case 'I': case 'Í':
    case 'O': case 'Ó':
    case 'U': case 'Ú':
        printf("Es vocal mayuscula\n");
        break;

    default:
        printf("Es consonante mayuscula\n");
        break;
}

else
{
    if (car >= 'a' && car <= 'z')
    {
        switch(car)
        {
            case 'a': case 'á':
            case 'e': case 'é':
            case 'i': case 'í':
```

```
        case 'o': case 'ó':

        case 'u': case 'ú':

            printf("Es vocal minuscula\n");

            break;


        default:

            printf("Es consonante minuscula\n");

        }

    }

else

{

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

        printf("Es un numero\n");

    else

        printf("Es un simbolo\n");

    }

}

}

#include <stdio.h>


void main()

{

    char car;

    printf("Ingrese caracter: ");
```

```
scanf("%c", &car);
```

```
if (car >= 'A' && car <= 'Z')
```

```
{
```

```
    switch(car)
```

```
    {
```

```
        case 'A': case 'Á':
```

```
        case 'E': case 'É':
```

```
        case 'I': case 'Í':
```

```
        case 'O': case 'Ó':
```

```
        case 'U': case 'Ú':
```

```
            printf("Es vocal mayuscula\n");
```

```
            break;
```

```
        default:
```

```
            printf("Es consonante mayuscula\n");
```

```
            break;
```

```
    }
```

```
}
```

```
else
```

```
{
```

```
    if (car >= 'a' && car <= 'z')
```

```
    {
```

```
        switch(car)
```

```

{
    case 'a': case 'á':
    case 'e': case 'é':
    case 'i': case 'í':
    case 'o': case 'ó':
    case 'u': case 'ú':
        printf("Es vocal minuscula\n");
        break;

    default:
        printf("Es consonante minuscula\n");
}
}
else
{
    if (car >= '0' && car <= '9')
        printf("Es un numero\n");
    else
        printf("Es un simbolo\n");
}
}
}

```

El codeblocks no lee acentos y por ello el compilador no me permite poner acentos, solo hay una excepción son las vocales mayúsculas acentuadas.

**Actividad 3: Cree una versión del programa donde el SWITCH identifique colores primarios: R, G, B.**

```
#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;
```

```
            case 'R':
```

```
                printf("R es de red (rojo)\n");
```

```
                break;
```

```
            case 'G':
```

```
                printf("G es de green (verde)\n");
```

```
                break;
```

```
case 'B':
```

```
    printf("B es de blue (azul)\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");

    }

}

}

```

#### 4. Preguntas de reflexión

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

Por que así se pueden usar varios casos de variables, porque si se usa varios if luego queda muy desordenado, pero en si el if es bueno para cuando se quiere establecer condiciones.

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

Básicamente si no incluimos el break el switch que pusimos antes seguirá hasta que nuestro programa acabe y seguirá así con mas switch que se pongan en el programa, pero si se pone break sabrá el programa que vamos a usar otro tipo de casos nuevos y para eso sirve el break.

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

El programa fácilmente leerá el primer carácter por ejemplo si pone hola el programa solo leerá la letra h y así continuará con el programa.

#### 5. 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.

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

```

int main() {
    int num;

```

```

printf("Ingrese un número del 1 al 7: ");
scanf("%d", &num);

switch (num) {
    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 fuera de rango (1-7).\n");
}

return 0;
}

```

## 6. 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