



*Programming*

**Fundamentals**

# Table of Contents

- SWITCH Construct
- BREAK Statement
- Nested Switch



*SECTION 1:*

**SWITCH Construct**



# What is Switch Construct?

---

- A **decision-making structure** in programming.
- Allows a **variable/expression** to be compared with **multiple fixed constant values**.
- In few cases, it provides a cleaner and more efficient alternative to **long if-else if** chains.

# SYNTAX

---

```
switch(expression) {  
    case constant1:  
        // code  
        break;  
    case constant2:  
        // code  
        break;  
    ...  
    default:  
        // code  
}
```

## Rules:

- expression must be an integer, or char type.
- case labels must be constant values (not variables or expressions).
- break exits the switch block.
- default is executed if no match is found.

# If-Else-If

vs

# Switch

```
int day = 3;
```

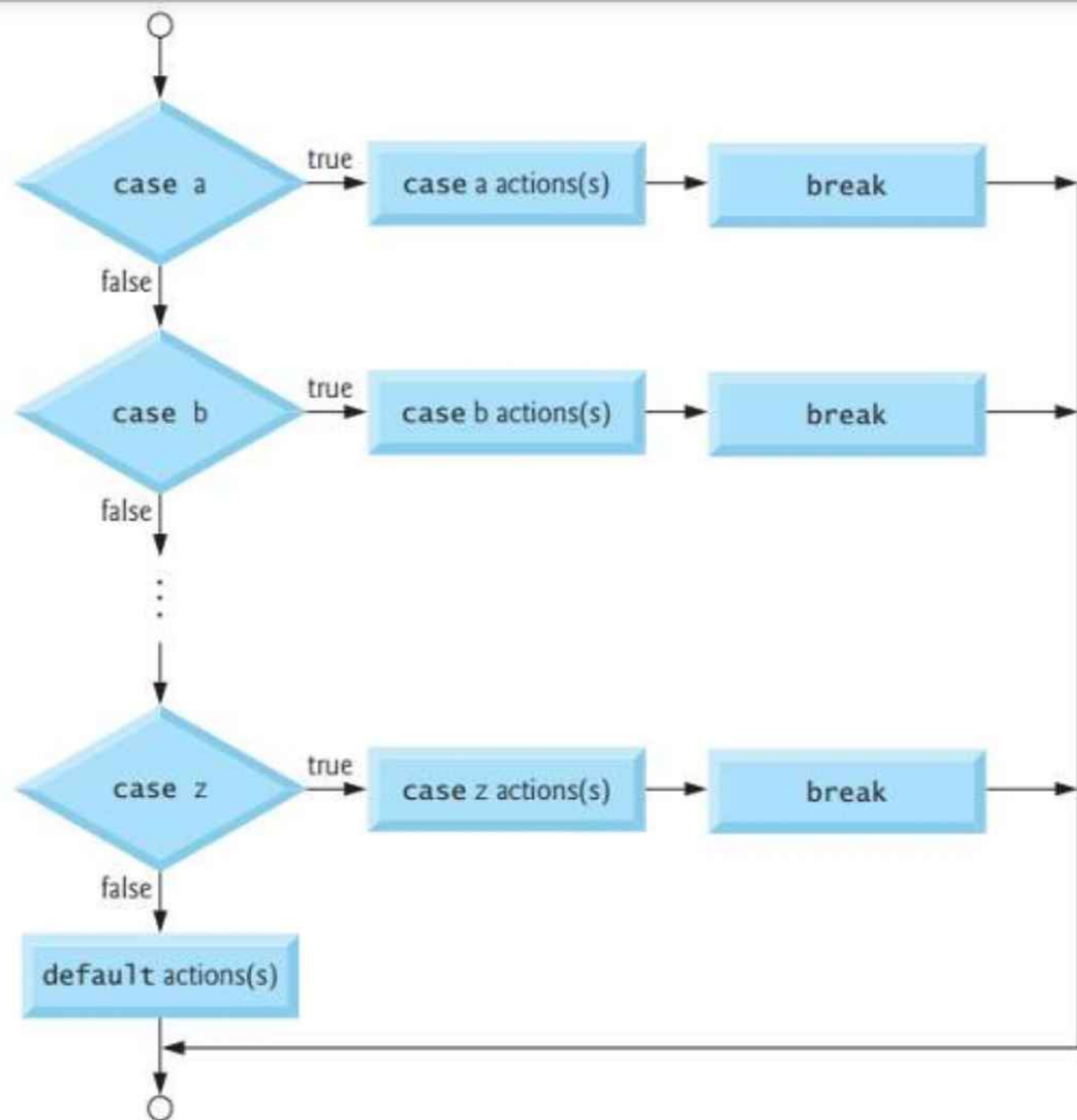
```
if(day == 1) cout << "Monday";  
else if(day == 2) cout << "Tuesday";  
else if(day == 3) cout <<  
"Wednesday";  
else cout << "Invalid";
```

```
int day = 3;
```

```
switch(day) {  
    case 1: cout << "Monday"; break;  
    case 2: cout << "Tuesday"; break;  
    case 3: cout << "Wednesday";  
    break;  
    default: cout << "Invalid";  
}
```



# FLOWCHART





## *SECTION 2:*

# **BREAK Statement**



# Break Statement

---

- break stops execution and exits the switch.
- Without break, program falls through to the next case.

## WITH BREAK

```
int x = 2;  
switch(x) {  
    case 1: cout << "One"; break;  
    case 2: cout << "Two"; break;  
    case 3: cout << "Three"; break;  
}
```

OUTPUT: Two

## WITHOUT BREAK

```
int x = 2;  
switch(x) {  
    case 1: cout << "One";  
    case 2: cout << "Two";  
    case 3: cout << "Three";  
}
```

OUTPUT: TwoThree

# **Q: Simple ATM Menu**

**Write a C program to simulate a simple ATM menu using a switch statement.**

- **The program should start with a balance of 1000.**
- **Display the following menu to the user:**

- 1. Withdraw**
- 2. Deposit**
- 3. Check Balance**
- 4. Exit**

- **If the user chooses 1 (Withdraw), ask for the withdrawal amount, deduct it from balance, and display the remaining balance.**
- **If the user chooses 2 (Deposit), ask for the deposit amount, add it to balance, and display the new balance.**
- **If the user chooses 3 (Check Balance), simply display the current balance.**
- **If the user chooses 4 (Exit), display "Exiting..." and end the program.**
- **If the user enters an invalid option, display "Invalid choice".**



# Q: Calculator

---

**Write a C program to implement a simple calculator using a switch statement.**

- **The program should take two integers and a mathematical operator (+, -, \*, /, %) as input.**
- **Based on the operator entered, the program should perform the corresponding calculation and display the result.**
- **Handle the case where the user tries to divide by zero or use modulus with zero.**
- **If the operator entered is not valid, display "Invalid operator".**





# *SECTION 3:* **Nested Switch**

# Q: Day and Activity Planner

---

**Write a C program using nested switch:**

- **Ask the user to enter a day of the week (1 = Monday, ..., 7 = Sunday).**
- **For weekdays (1–5):**
  - **Ask if the time is Morning (M) or Evening (E).**
  - **If Morning → Print "Go to Class"**
  - **If Evening → Print "Do Homework"**
- **For weekends (6–7):**
  - **Ask if the time is Morning (M) or Evening (E).**
  - **If Morning → Print "Play Sports"**
  - **If Evening → Print "Watch Movie"**
- **If day is invalid → Print error.**

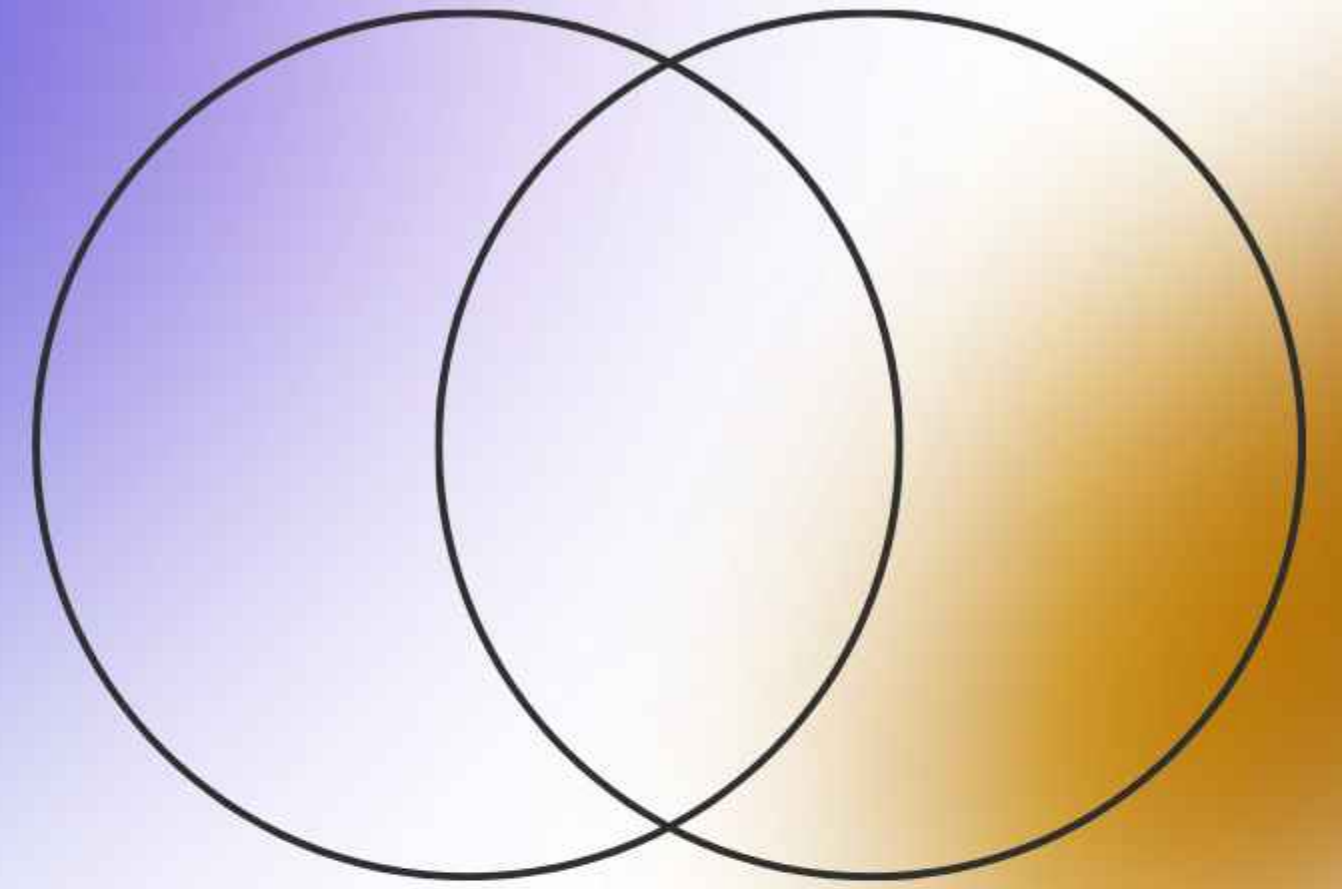
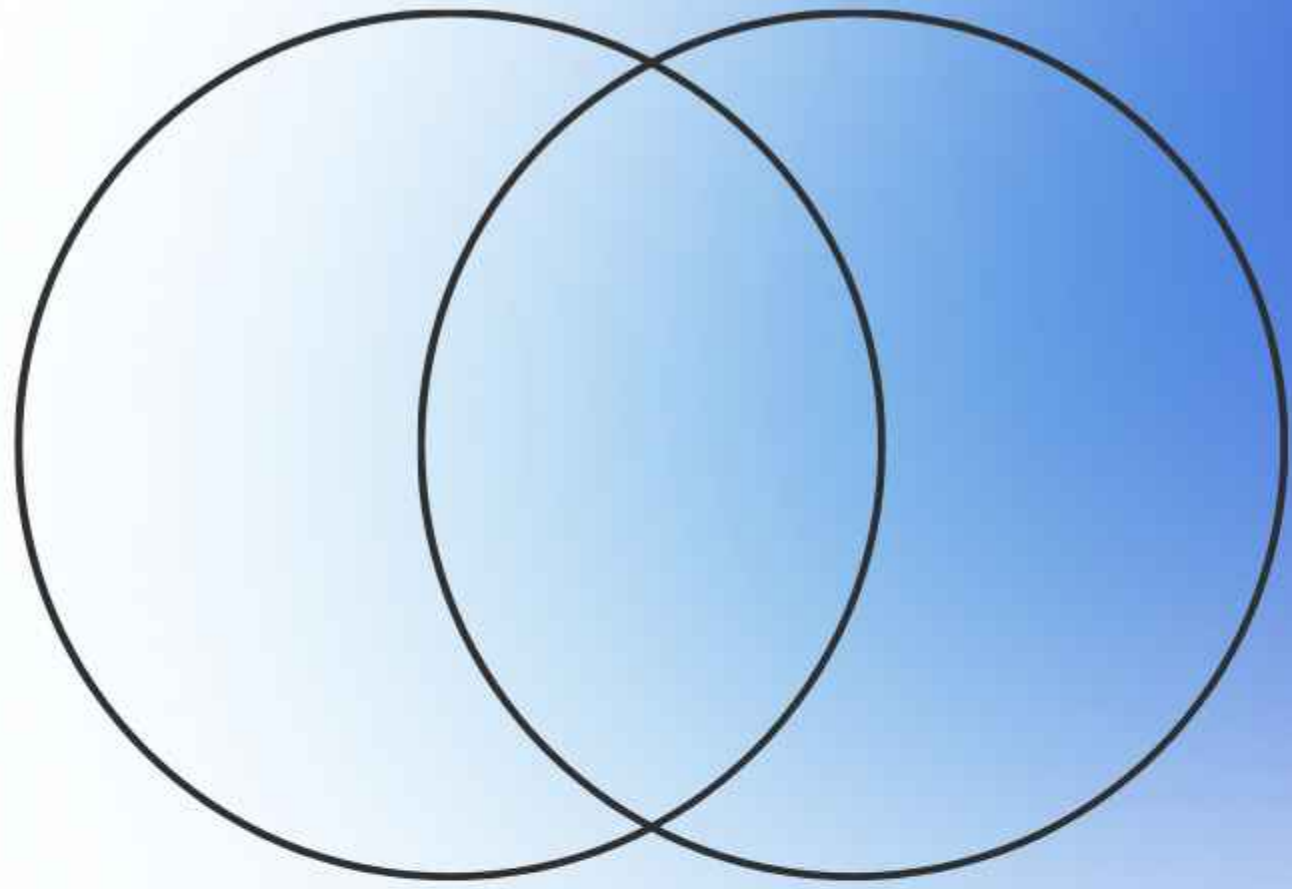
# Q: College Program and Courses

---

**Write a C program using nested switch:**

- **Ask the user to enter program type:**
  - **1 = Computer Science**
  - **2 = Electrical Engineering**
- **Inside each program, ask for year:**
  - **1 = First Year**
  - **2 = Second Year**
- **Display courses accordingly.**





**Thank You**