Switch Statement

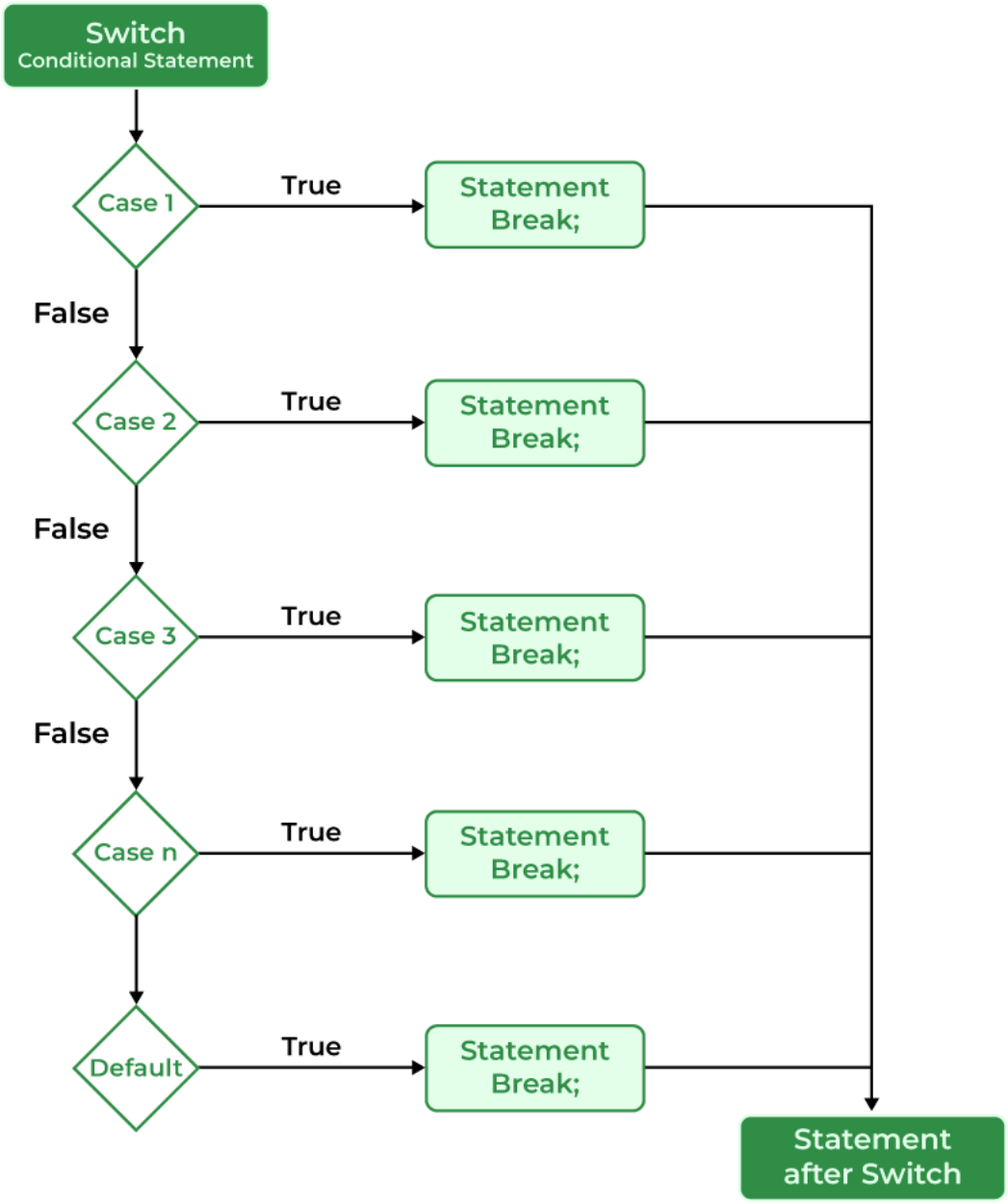
Switch statement is a control statement that allows us to choose only one choice among the many given choices. The expression in switch evaluates to return an integral value, which is then compared to the values present in different cases. It executes that block of code which matches the case value. If there is no match, then **default** block is executed (if present). The switch statement is a **multi-way** branch statement . It provides an easy way to dispatch execution to different parts of code based on the value of the expression.

The general form of switch statement is

```
switch(expression)
{
    case [constant-expression] : statement(s);
                                 break;
    case [constant-expression] : statement(s);
                                 break;
    …..
    …..
    default : statement(s);
}
```

Important Points about Switch Case Statements:

1. The expression provided in the switch should result in a **integral constant value** otherwise it would not be valid.
2. Duplicate case values are not allowed.
3. The default statement is optional. Even if the switch case statement does not have a default statement, it would run without any problem.
4. The break statement is used inside the switch to terminate a statement sequence.When a break statement is reached, the switch terminates, and the flow of control jumps to the next line following the switch statement.
5. The break statement is optional. If omitted, execution will continue on into the next case. The flow of control will fall through to subsequent cases until a break is reached.
6. Nesting of switch statements are allowed, which means you can have switch statements inside another switch. However nested switch statements should be avoided as it makes program more complex and less readable.



Following is a simple C program to demonstrate syntax of switch.

```
#include <stdio.h>
int main()
{
    char ch;
    double first, second;
    printf("Enter an operator (+, -, *, /): ");
    scanf("%c", &ch);
    printf("Enter two operands: ");
    scanf("%lf %lf", &first, &second);
    switch (ch)
    {
        case '+':printf("%.1lf + %.1lf = %.1lf", first, second, first + second);
                 break;
```

```
        case '-':printf("%.1lf - %.1lf = %.1lf", first, second, first - second);
                break;
        case '*':printf("%.1lf * %.1lf = %.1lf", first, second, first * second);
                break;
        case '/':printf("%.1lf / %.1lf = %.1lf", first, second, first / second);
                break;
        // operator doesn't match any case constant
        default : printf("Error! operator is not correct");
    }
    return 0;
}
```

Output :-

```
Enter an operator (+, -, *, /): +
Enter two operands: 45
34
45.0 + 34.0 = 79.0
```