Multiway Branches

By: Mir Faraz Ali

Programming Fundamentals

Multiway Branches in Programming

- In programming, branching means choosing one path of execution out of many possible paths.
- When we have more than two choices, it is called a multiway branch.
- There are 3 common ways to write multiway branches:
- Nested if statements
- Multiway if-else statements
- Switch statements

Nested if Statements

- Definition:
  - This means putting one if statement inside another.
  - It is used when a decision depends on the result of a previous decision.
  - Example: First check if a student passed, then check if they got an A grade.

Example:

```
#include <iostream>
using namespace std;
int main() {
    int marks = 85;
    if (marks >= 50) {
        if (marks >= 80) {
            cout << "Grade: A" << endl;
        } else {
            cout << "Grade: B" << endl;
        }
    } else {
        cout << "Fail" << endl;
    }
    return 0;
}
```

Multiway if-else statements

- Definition:
  - This is a chain of conditions using if, else if, and else.
  - It is used when we need to choose from many possible options.
  - Only one condition will be true, and its block will run.
  - Example: Checking if a number is positive, negative, or zero.

Example:

```
#include <iostream>
using namespace std;
int main() {
    int number = 0;
    cout << "Enter a number: ";
    cin >> number;

    if (number > 0) {
        cout << "Positive number" << endl;
    } else if (number < 0) {
        cout << "Negative number" << endl;
    } else {
        cout << "Zero" << endl;
    }
    return 0;
}
```

Switch statement

- Definition:
  - This allows us to test the value of a single variable against many fixed cases.
  - It is easier and cleaner than writing many if-else statements.
  - Example: Printing the day of the week based on the number entered (1 = Monday, 2 = Tuesday, etc.).

Example:

```
#include <iostream>
using namespace std;
int main() {
    int day = 3;
    switch(day) {
        case 1:
            cout << "Monday" << endl;
            break;
        case 2:
            cout << "Tuesday" << endl;
            break;
        case 3:
            cout << "Wednesday" << endl;
            break;
        case 4:
            cout << "Thursday" << endl;
            break;
        case 5:
            cout << "Friday" << endl;
            break;
        default:
            cout << "Weekend" << endl;
    }
    return 0;
}
```

End Of Class