

## Chapter 18 – More on Nested Ifs

*If* and *if-else* statements are like any other statement in that they may be used inside of other *if* and *if-else* statements. This is referred to as nesting. Nested *if* and *if-else* statements can be used to avoid complex Boolean expressions or to add reliability in the construction of decision trees (*i.e. places where a single choice must be made to follow one out of multiple but mutually exclusive paths of logic*).

For example, the following single *if* statement can be rewritten with three *ifs*, but with simpler Boolean logic.

```
if (a == b && a == c && a == d)
    statement;
```

can be rewritten

```
if (a == b)
    if (a == c)
        if (a == d)
            statement;
```

In either example, the statement following the *if* or *nested-ifs* will be performed only if the variable objects *a*, *b*, *c* and *d* share the same value.

In the following example, there are three possible but mutually exclusive outcomes based on the age of the program user. Nesting two *if-else* statements provides a reliable way to pick only one of the possible outcomes.

```
cout << "Enter your age: ";
int age;
cin >> age;
if (age < 0)
    cout << "The age entered must be 0 or greater." << endl;
else if (age < 18)
    cout << "You are too young to vote." << endl;
else
    cout << "You are old enough to vote." << endl;
```

If the user enters an invalid age (less than 0), the Boolean expression of the first *if* becomes true and its *cout* statement (`cout << "The age entered must be 0 or greater." << endl;`) is performed, with the rest being ignored.

If the user enters an age between 0 and 17 inclusive, the Boolean expression of the first *if* is false and the program logic proceeds to the *else*, where the second *if* is encountered. With an age value between 0 and 17, the Boolean expression of the second *if* is true and its *cout* statement (`cout << "You are too young to vote." << endl;`) is performed, with the last *cout* being ignored.

If the user enters an age 18 or greater, the Boolean expression of the first *if* is false and the program logic proceeds to the *else*, where the second *if* is encountered. As the value in age is 18 or greater, the Boolean expression of the second *if* is also false and the program logic proceeds to the second *else*, where the last *cout* statement (`cout << "You are old enough to vote." << endl;`) is encountered.

*Nested ifs* may be taken to any level of complexity.

### **Programming Exercise 18.1**

Implement a program that checks the voting age of the user by completing the nested if voting example.

### **Programming Exercise 18.2**

Create a program that accepts the age input and determines if the user is too young to drive, old enough to be required to drive only in daylight hours (81 or older), or may drive at any time.

### **Programming Exercise 18.3**

Create and use a function that receives a value  $n$  and returns  $(2*n*n)$  when  $n$  is less than 0, 1 if  $n$  is 0, and  $(-2*n*n)$  when  $n$  is greater than 0.