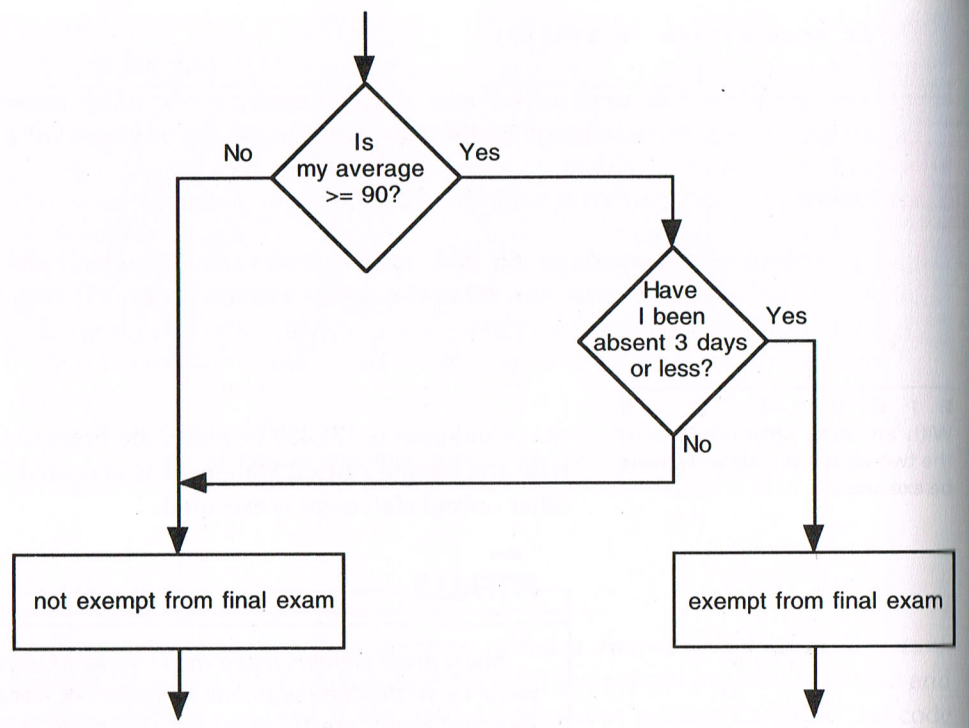

FIGURE 7-9
This flowchart can be programmed using nested if structures.

```
exempt_from_final = FALSE;
if (my_average >= 90)
  {
    if (my_days_absent <= 3)
      { exempt_from_final = TRUE; }
  }
```

FIGURE 7-10
Nested if structures can be used to check two requirements before making a final decision.

Algorithms involving nested if structures can get more complicated than the one in Figure 7-9. Figure 7-11 shows the flowchart from Figure 7-9 expanded to include another way to be exempted from the final exam. In this expanded algorithm, students can also be exempted if they have an 80 or higher average, as long as they have been present every day or missed only once.

Note

In the code segment in Figure 7-10, TRUE and FALSE have been declared as constants with the values 1 and 0 respectively. The variable that tells whether the person is exempt from the final is set to false. The program assumes that the student fails to meet the exemption qualification and tests to determine otherwise.

As you can probably imagine, programming the algorithm in Figure 7-11 will require careful construction and nesting of if and if/else structures. Figure 7-12 shows you how it is done.

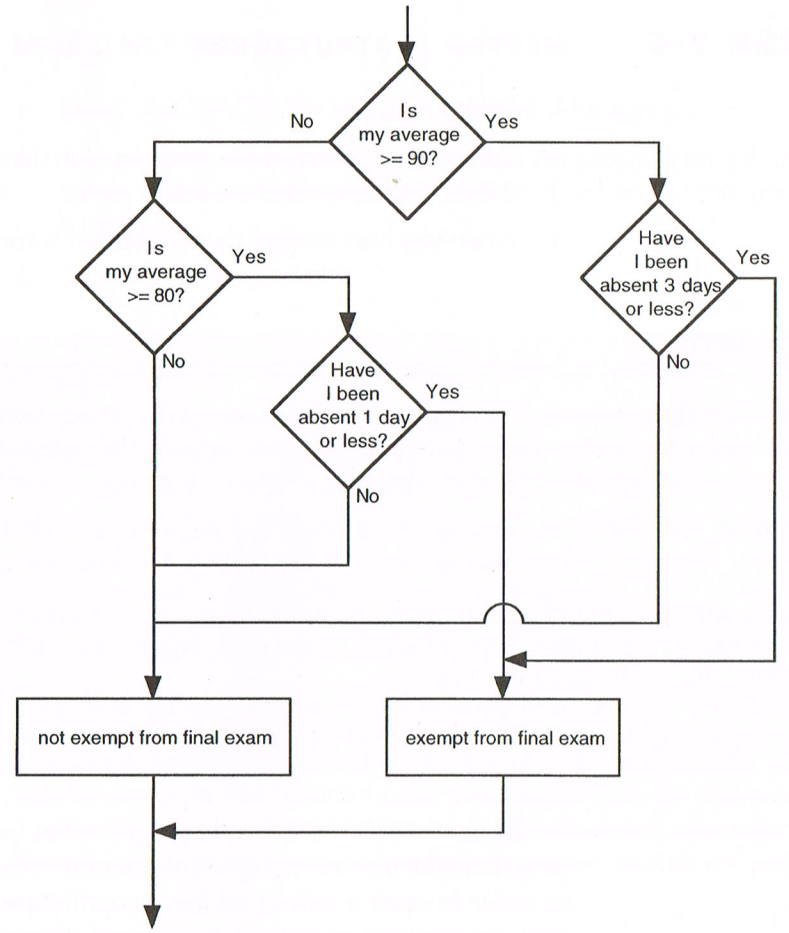

FIGURE 7-11
This algorithm provides two paths to exemption from the final.

Note

Earlier you learned that it is a good idea to always use braces with if structures. Figure 7-12 illustrates another reason why you should do so. Without the braces, the compiler may assume that the else clause goes with the nested if structure rather than the first if.

```
if (my_average >= 90)
  {                                  // if your average is 90 or better
   if (my_days_absent <= 3)          // and you have missed three days
     { exempt_from_final = TRUE; } // or less, you are exempt.
  }
else  // if you don't have a 90+ average, you still have a chance
  { if (my_average >= 80)
     {                                 // if your average is 80 or
      if (my_days_absent <= 1)         // better and you have missed
        { exempt_from_final = TRUE; } // one day or less, you are
     }                                 // exempt.
  }
```

FIGURE 7-12
Nested if structures can require careful construction.