Combining Conditions

- A loop may depend on multiple conditions, just like with `if…else`
  - Can use a complex logical statement (using `&&`, `||`)
  - Can also use an intermediate boolean *flag* variable

Example Case: Flag Variable

- Use a `while` loop to do something until a certain boolean variable becomes false
  1. Initialization (before the loop)
     - Declare a boolean *flag* variable
     - Assign it an initial value of true
  2. Condition (the while condition)
     - Check to see if the flag is true
       - If it is not, quit the loop
  3. Update (in the body of the loop)
     - Do the repetitive tasks
     - Something in the body must potentially set the flag to false, otherwise the loop will never end
  4. Steps 2 and 3 repeat

do…while Loop

- while loop executes 0 or more times
- do…while loop executes 1 or more times

```
do
{
   // do some stuff then repeat
   //  as long as condition is true
}
while( condition )
```

while vs. do…while Loop

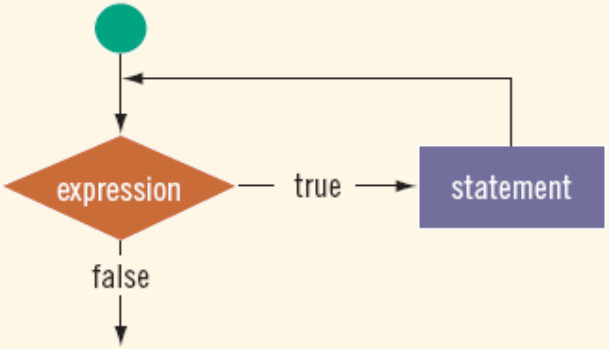

FIGURE 5-1 while loop

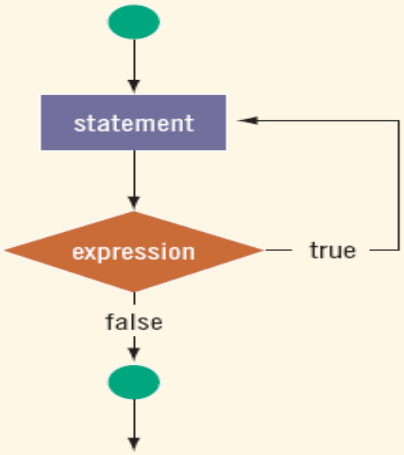

FIGURE 5-3 do...while loop

do...while Loop Examples

EXAMPLE 5-15

```
i = 0;

do
{
    cout << i << " ";
    i = i + 5;
}
while (i <= 20);
```

The output of this code is:

```
0 5 10 15 20
```

EXAMPLE 5-16

Consider the following two loops:

a.
```
i = 11;
while (i <= 10)
{
    cout << i << " ";
    i = i + 5;
}
cout << endl;
```

b.
```
i = 11;
do
{
    cout << i << " ";
    i = i + 5;
}
while (i <= 10);

cout << endl;
```

In (a), the `while` loop produces nothing. In (b), the `do...while` loop outputs the number `11` and also changes the value of `i` to 16.

Exercise

- Write the output of the following loops:

a:
```
int i = 5;
while ( i > 0 )
{
      cout << i << " ";
      cout << endl;
      i--;
}
```

b:
```
int i = 0;
do
{
      cout << i << " ";
      i++;
}
while ( i <= 5 );
cout << endl;
```

c:
```
int i = 0;
while ( i < 10 )
{
      i = i + 2;
      cout << i << " ";
}
cout << endl;
```

d:
```
int i = 7;
do
{
      cout << i << " ";
      i++;
}
while ( i < 5 );
cout << endl;
```