for loop

for vs. while loop

The `for` Loop vs. the `while` loop

- Often you will need to execute a sequence of statements a given number of times.

  You could use a `while` loop:

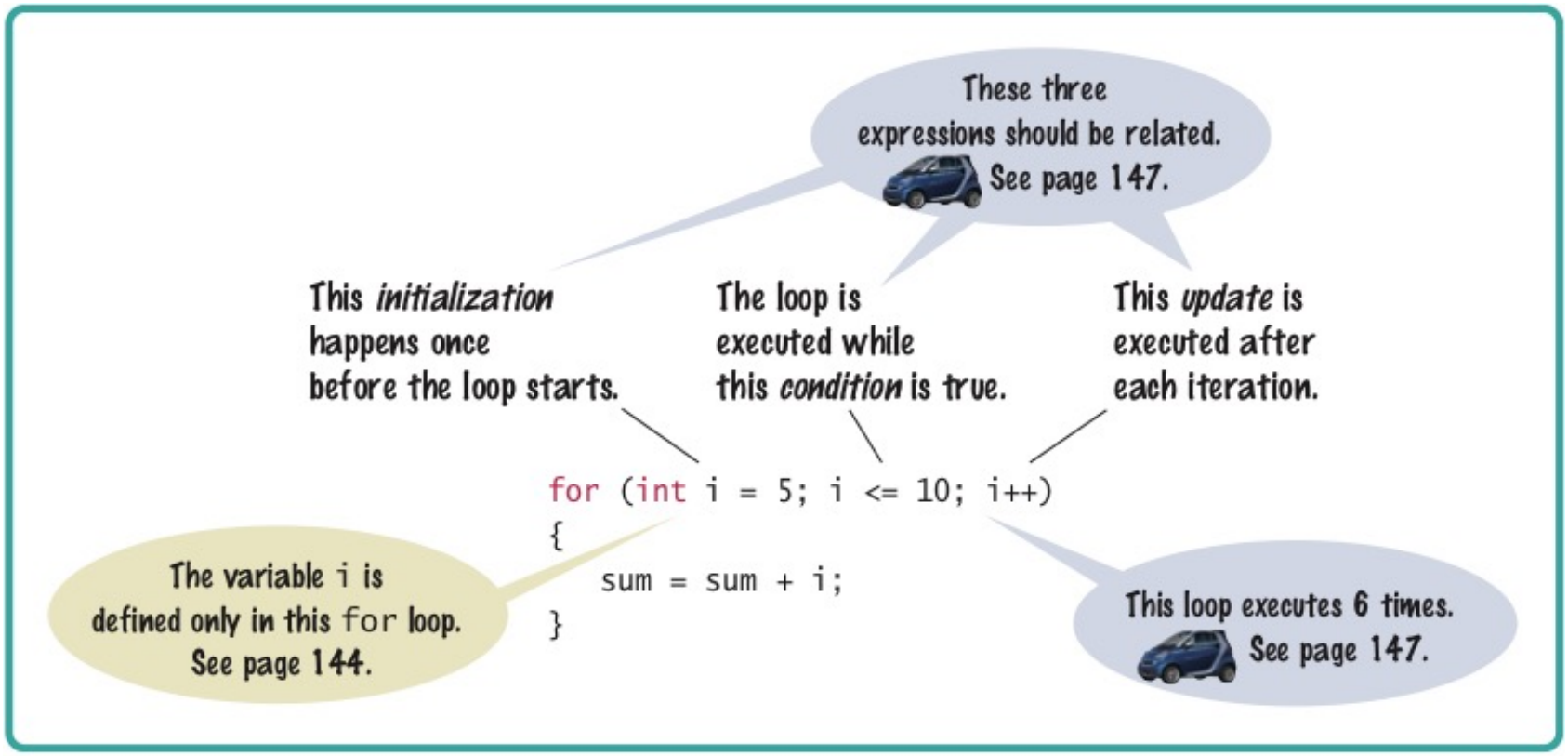
```
num = 1; // Initialize the variable
while (num <= 10) // Check the variable
{
   cout << num << endl;
   num++; // Update the variable
}
```

The `for` Loop

- C++ has a statement custom made ***for*** this sort of processing: the **`for`** loop.

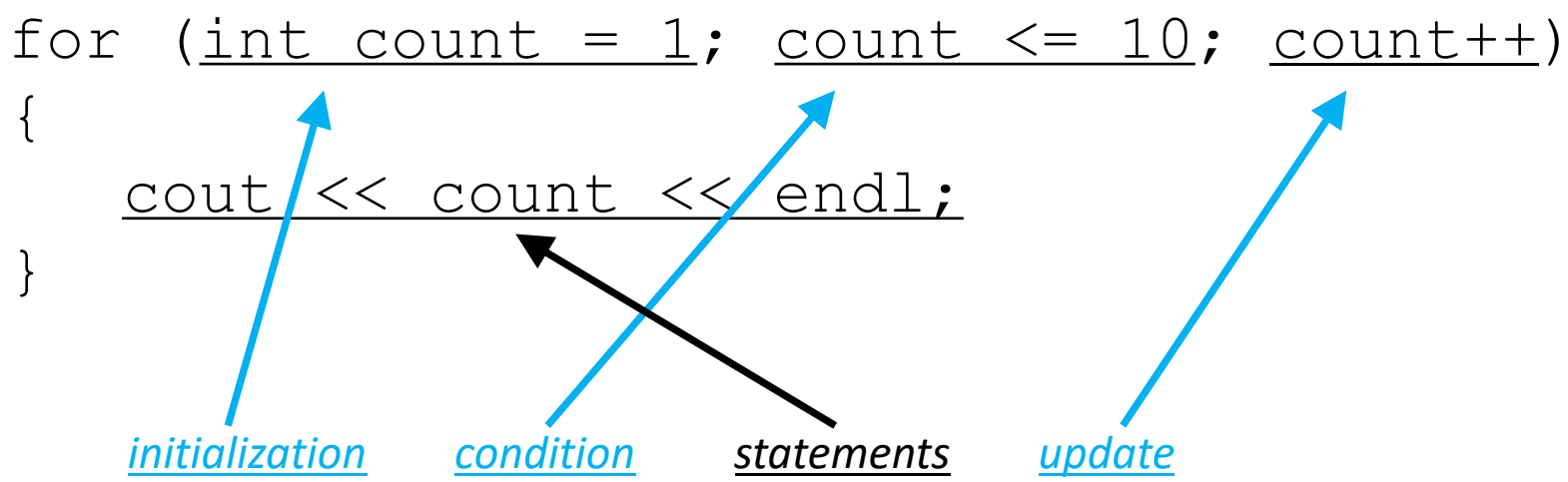
```
for (num = 1; num <= 10; num++)
{
   cout << num << endl;
}
```

The `for` Loop Syntax



The `for` Loop Is Better than `while` for Certain Things

- Doing something a known number of times or causing a variable to take on a sequence of values is so common, C++ has a statement just for that:



`for()` loop execution

```
for (initialization; condition; update)
   {
      statements;
   }
```

- The ***initialization*** is code that happens once, before the check is made, to set up counting how many times the *statements* will happen. The loop variable may be created here, or before the `for()` statement.
- The ***condition*** is a comparison to test if the loop is done. When this test is false, we skip out of the `for()`, going on to the next statement.
- The ***update*** is code that is executed at the bottom of each iteration of the loop, immediate before re-testing the condition. Usually it is a counter increment or decrement.
- The ***statements*** are repeatedly executed until the condition is false. These also are known as the "loop body".

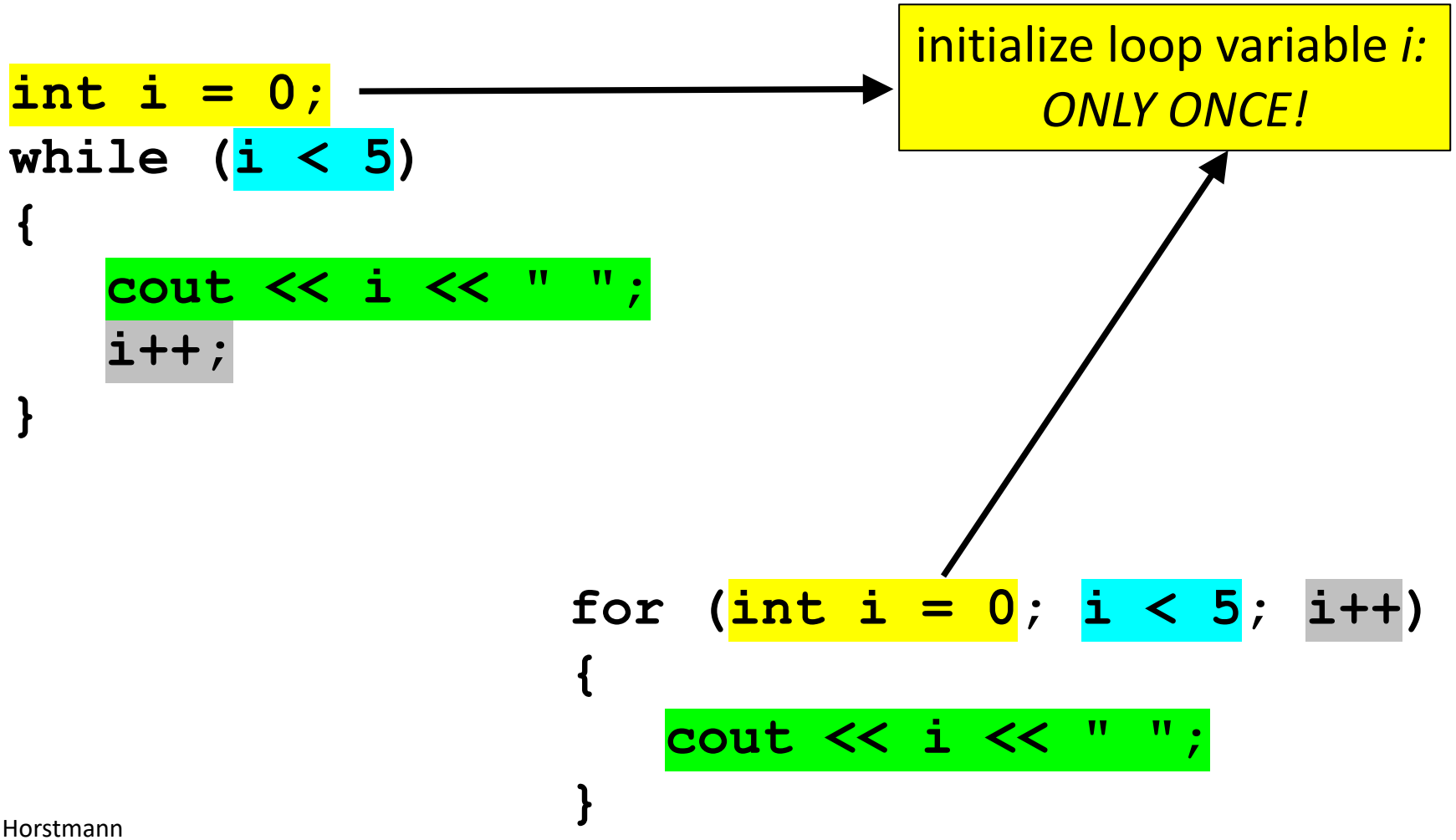
The `for` Can Count Up or Down

- A `for` loop can count down instead of up:

```
for (int counter = 10; counter >= 0; counter--)…
```

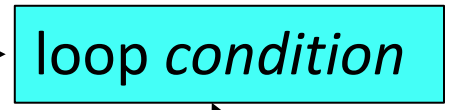
- Notice that in this examples, the loop variable is defined **in** the *initialization* (where it really should be!).

Converting from a *while* loop to a *for* loop



Converting from a *while* loop to a *for* loop

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




```
for (int i = 0; i < 5; i++)
{
   cout << i << " ";
}
```

Converting from a *while* loop to a *for* loop

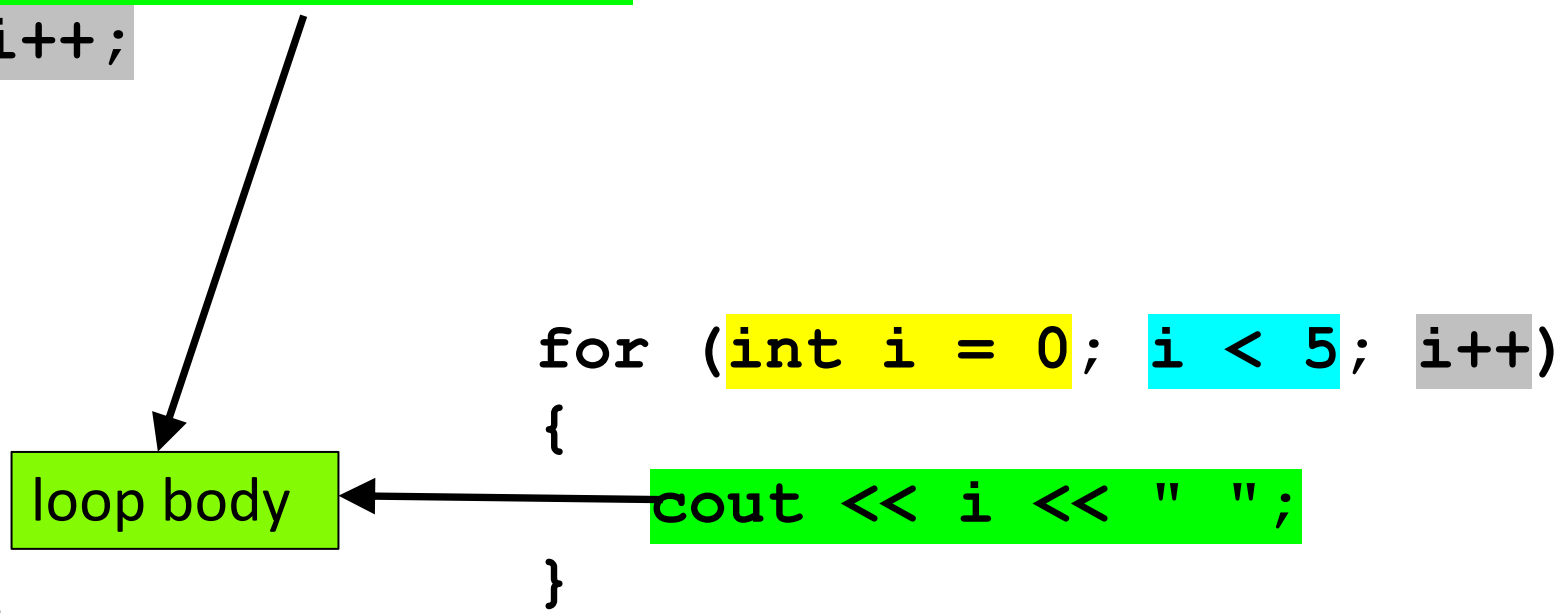
```
int i = 0;
while (i < 5)
{
   cout << i << " ";
   i++;
}
```

update loop variable *i*

```
for (int i = 0; i < 5; i++)
{
   cout << i << " ";
}
```

Converting from a *while* loop to a *for* loop

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



Converting from a *while* loop to a *for* loop