Loops

Section 5.2 - 5.5
Page 255

14.1 Running totals (5.5)

• How many hours did you work on assignment 1?

```
                              // initialize the counter
int days;                     // number of days


// let the user tell us how many times to loop
cout << "How many days did you work on assignment 1? ";
cin >> days;

while(                )   // test the counter
{
  cout << "How many hours did you work on day " <<    << "? ";



                              // increment the counter
}
```

14.2 Practice

• Write a snippet of code that will ask the user for a number. Print the *sum* of all the numbers from 0 to the number the user supplied.

1

14.3 Practice

- Write a snippet of code that will ask the user for a number. Print the sum of all the *even* numbers from 0 to the number the user supplied.

Increment and Decrement Operators

- C++ provides a shortcut to increment or decrement a variable by 1

```cpp
int x = 99, y = 90;
x++; // this is equivalent to x += 1
x--; // this is equivalent to x -= 1
```

In a Loop

- Often, this is used to increment a loop counter

```cpp
int x = 1;
while(x < 100)
{
    cout << " x : " << x << endl;
    x++;
}
```

14.4 Examples

a. This can be used in an expression:
```
y = x++ + 9;
```

This is equivalent to:
```
y = x + 9;
x += 1;
```

b. This can also be used in a conditional

```
( x-- > 9 )
```

is equivalent to:
```
( x > 9); x -= 1;
```

14.5 Practice

- Write one statement of code to do the following:

```
int x = 0, y = 1;
```

- Add x + 9 to y and increment x by 1

- Add x * 4 to y and increment x by 1

- Add y – 13 to x and decrement y by 1

Prefix vs Postfix

- Another way to write x++ (x--) is: ++x (--x)
  - ++x is *prefix*
    - The x += 1 happens *before* the expression is evaluated
  - x++ is *postfix*
    - the x += 1 happens *after* the expression is evaluated

```
int y = 0, x = 0;
x = y++ + 1; // x = y + 1; y += 1;
y = ++x + 1; // x += 1; y = x + 1;
```

Operator Precedence

()	**L->R**	Parentheses
++, --, **static_cast<type>()**	**L->R**	Unary prefix (++x)
++, --, !, +, -	**R->L**	Negation, Unary postfix (x++)
*,/,%	**L->R**	Mult, div, mod
+, -	**L->R**	Add, Subtract
<<, >>	**L->R**	Insertion/extraction
<, <=, >, >=	**L->R**	Relational
==, !=	**L->R**	Equality
&&	**L->R**	And
||	**L->R**	Or
?:	**R->L**	Conditional
=, +=, -=, *=, /=, %=	**R->L**	Assignment

14.6 Examples

```
int x = 0, y = 0;

x = y++ * 2;
y = ++x / 2;

x = x++ + 1;
x = ++x + 1;

y = (y+ x++) * 2;
x = y++ + ++x;
```

14.7 Example

• What is the output if i = 2?

```
cout << "Value of i is" << i;

cout << "Value of i++ is" << i++;

cout << "Value of ++i is" << ++i;

cout << "Value of --i is" << --i;

cout << "Value of i-- is" << i--;
```