Loops

section 5.1, 5.2, 5.4

Increment and Decrement Operators (5.1)

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

```
int x = 99;

x++; // this is equivalent to x += 1
x--; // this is equivalent to x -= 1
```

Prefix and Postfix

Prefix	Postfix
`k = --x;`	`k = x--;`
`k = ++x;`	`k = x++;`
Increment/decrement x then assign value of x to k	Assign value of x to k, then increment or decrement x

Example

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

x = y++;

cout << x << " " << y << " "
    << z << endl;

y = ++z;

cout << x << " " << y << " "
    << z << endl;

z = x ++ + 1;

cout << x << " " << y << " "
    << z << endl;
```
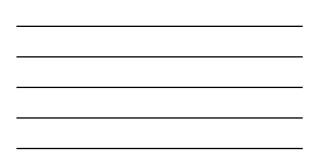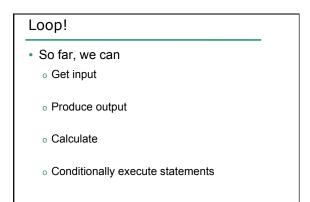
4

Loop!

- So far, we can
  - Get input

  - Produce output

  - Calculate

  - Conditionally execute statements

Loops

- Perform the same bit of code many times

```
statement1;
statement2;
. . .
statement3;
```

  - Why might we want to do this?

While Loop (5.2)

- **while** the **expression** is **true**, loop!

```
while ( expression )
{
    statement1;
    statement2;
    . . .
    statement3;
}
statement4;
```

[1]Test the expression

[2] Either

Perform the statements
in the loop if expression is true

[3]Repeat

or

Move past the loop if expression
is false

Example: What happens?

```
int count = 0;

while (count < 4)
{
  cout << "I'm looping! ";
  cout << count << endl;
  count ++;
}
cout << endl << "I'm done looping. ";
cout << count << endl;
```

Counters (5.4)

- Counter: A variable that is incremented or decremented each time a loop

```
int theCounter = 0;     // initialize the counter

while (theCounter < 5)  // test the counter
{
  cout << "*****" << endl;
  theCounter ++;         // udpate the counter
}
```
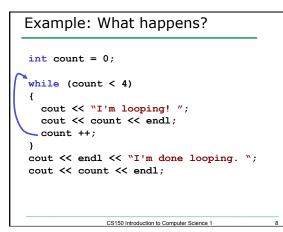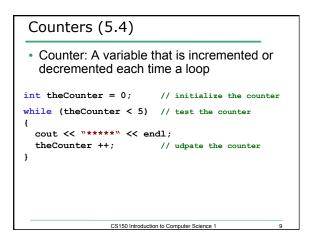
Key Ingredients of `while` loops

- Initialize

- Test

- Update

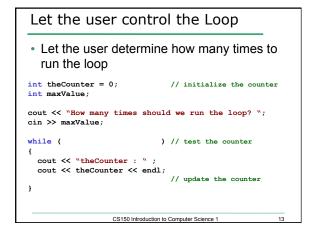*If any one of these is missing or incorrect, your loop won't run properly--not at all, too many/ few times or infinitely.*

What is the Output?

```cpp
int theCounter = 0;      // initialize the counter


while (theCounter < 5)  // test the counter
{
  cout << "*****" << endl;
}
```

Practice

- Write a snippet of code that will print all the integers from 0 to 20 using a while loop.

- Write a snippet of code that will print a table containing the integers from 0 to 20 and their squares.

Let the user control the Loop

- Let the user determine how many times to run the loop

```cpp
int theCounter = 0;            // initialize the counter
int maxValue;

cout << "How many times should we run the loop? ";
cin >> maxValue;

while (                    ) // test the counter
{
   cout << "theCounter : " ;
   cout << theCounter << endl;
                            // update the counter
}
```

Practice

- Write a snippet of code that will ask the user for an integer. Print the integers from 0 to the square of the number the user supplied.

On Friday we Solved

- Write a C++ program segment that allows the user the ability to input an integer from the keyboard.

- If the integer is positive, increment a variable **posCount** by 1. If the integer is **negative**, increment a variable **negCount** by 1. If neither, increment **zeroCount** by 1

```cpp
int posCount=0, negCount=0, zeroCount=0;
```

- **How can we modify the program so that the user inputs 20 integers and the program outputs the number of positive integers, the number of negative integers, and the number of zeros that the user entered?**

Practice

- Write a piece of code that will ask the user for two integers and display them to the screen. If the sum of the two integers is less than 100, repeat the above action.

_____

_____

_____

_____

_____

_____

_____

_____