

While Construct in C++

Instructor: Andy Abreu

Repeat Decision Making

- Computers are great at repeating tasks
 - Building cars on the assembly line
 - You can picture this is the task of building one car being repeated, as long as there are parts
- Give user three attempts at entering password
- Calculating averages for each student for an entire class.

While Loop Model

- Model:

```
while ( comparison_is_true )  
{  
    //do something here  
}
```

While Loop Model Example

- Example:

```
int number;
```

```
cout << "Enter a positive number: ";
```

```
cin >> number;
```

```
while ( number <= 0 )
```

```
{
```

```
    cout << "Try again: ";
```

```
    cin >> number;
```

```
}
```

Counting in While Loop

- Sometimes we need to repeat a task specific number of times:
 - We will have to setup a counter before the loop
 - Check to see if counter has been met before and each time we loop
 - Update the counter each time we loop

While Loop Model (counter)

- Model:

//initialize counter

while (comparison on counter)

{

//do something processing here

//update counter, prevent infinite loop

}

While Loop Example (counter)

- Example:

```
int counter = 1; //initialize
while ( counter <= 10 ) //compare
{
    cout << counter << endl; //do something
    counter = counter + 1; //update
}
```

Parts of the While Loop

- For every while loop to function properly three components must exist:
 - Initialization
 - Comparison
 - done repeatedly to check for termination condition
- Update
 - Changes the condition every time, makes loop meaningful

Incrementing in C++

- Suppose we want to increment a number, how do we do so?

```
int counter = 1;  
counter = counter + 1;
```

- You can also do any of the following:

```
++counter;  
counter++;  
counter += 1;
```

Decrementing in C++

- Suppose we want to decrement a number, how do we do so?

```
int counter = 10;  
counter = counter - 1;
```

- You can also do any of the following:

```
--counter;  
counter--;  
counter -= 1;
```