Math and User Input

Now let's move on to some simple math programs. Let's start by writing a program that converts a person's weight on earth to his or her weight on the moon. An object that weighs one pound on Earth would weigh 0.1654 pounds on the moon, according to Wikipedia. Therefore, if we know the weight of something on Earth, we can multiply that weight by 0.1654 to get its weight on the moon. In C++, we represent multiplication with the * symbol.

In order to write this program, we will need to ask the user of our program for his or her weight, store that information in the computer, do the conversion, and then display the moon weight back to the user. The easiest way to do this is to save all the numbers in the program as variables. Just like in math class, a variable is a letter or a word that stands for a number or a mathematical formula. For example, we can say

```
x = 4;
y = 5 * x;
```

and whenever we use the variable x in our program, the computer will know that the x means 4. Likewise, the computer knows that y means 20. When you write variables, try to use better names than x and y. A good variable name describes the value that it stands for. So, if we have a variable that stands for a person's weight on earth, we should call that variable *earthWeight*. Keep in mind that variables can't have spaces and should not start with numbers.

In C++ , we are required to define the *type* of our variables. For example, if we know that a variable will store a decimal point number, we declare it to be a *float*, which stands for floating (decimal) point number. We always declare our variables inside a function before writing any other code (so for the main function, it would be the first line under *int main(){*). We do this by saying the variable type and then the names of all the variables that are that type, followed by a semi-colon. For example, we would say *float earthWeight* for the variable *earthWeight* described above. For our program, all our variables will be floats, since they are all decimal point numbers. Here's a quick reference of types in C++:

Type	Example
`float`	`0.124` **or** `4.0`
`int`	`4` **or** `134`
`string`	`"Hello World"`
`bool`	`true`/`false`

Let's write out an algorithm for this program, so that we make sure the computer has all the information it needs to do the program.

- Save the conversion factor (0.1654) as the variable *conversionFactor*
- Ask the user's weight
- Save the weight to a variable called *earthWeight*
- Multiply *earthWeight* and *conversionFactor* and save the result as the variable *moonWeight*
- Display *moonWeight* to the user

The only tricky part of this program will be getting input from the user. In order to do that, we will need to use a special command called *cin*. This command takes whatever the user types and saves it to a variable. The *syntax* (the structure) of the command looks like this:

```
cin >> myVariable;
```

When the computer reaches this point in the program, it will display a prompt for the user to type. The user can then type and hit the [Enter] key, and the computer saves whatever the user typed to the variable called *myVariable*. In our program, we will replace *myVariable* with *earthWeight*.
The last thing we have to address is how to print out the person's weight on the moon after we do the conversion. We can use *cout* again and simply surround the variable with double-left angle brackets, like this:

```
cout << "You would weigh " << moonWeight << " lbs on the moon.\n";
```

This puts together the two strings and the variable *moonWeight* and sends the whole thing to cout to be displayed.

Now we're ready to write the program. The text preceded by "//" is a comment --- just notes to anyone reading the code. Comments are ignored by the computer and are just used by programmers to explain parts of their code. It's good practice to have short comments in your code.

```
#include <iostream>
using namespace std;

int main(){
      //define variables as floats
      float conversionFactor, earthWeight, moonWeight;
      //prompt the user to enter weight
      cout << "Enter your weight on earth:";
      cin >> earthWeight; //store what the user types as earthWeight
      conversionFactor = 0.1654;
      moonWeight = earthWeight*conversionFactor;
      //print out conversion
      cout << "You would weigh " << moonWeight << " lbs on the moon.\n";
}
```

There are lots of math operators available in C++. The most common are the following:

Symbol	Math
+	addition
-	subtraction
*	multiplication
/	division
%	modulus (remainder)

The % operator gives you the remainder of integer division. So if you want the remainder of 5/2 (which is 1), you would say:

```
rem = 5 % 2;
```

This is very useful in programming, since you can easily decide is a number is even, odd, or a multiple of some other number.

Challenge 1:

Write a program that reverses the program above. This program will take the user's weight on the moon and convert it to Earth weight.

Hint: What's the mathematical opposite of multiplication?

Challenge 2:

Write a program that asks the user for two numbers, and then prints out the sum of the two numbers.

Hint: Use multiple *cin* commands.

Source