Output Parameters

Let's take a quick look at the code for `getCoefficients()` to see how output parameters work.

```
void getCoefficients(double& x, double& y, double& z)
{
    cout << "Enter 3 coefficients: ";
    cin >> x >> y >> z;
}
```

If a function **returns more than one** piece of information, then you can use **reference parameters** to return that information to the caller.

> *Note that when you call **`getCoefficients`**, information **does not** flow from **`main`** **into** the function; instead, information **flows out of the function** back to **`main`**, through the three **output parameters** **`x`**, **`y`**, and **`z`**, which **are not new variables**, but are **new names** or **aliases** for the variables **`a`**, **`b`**, and **`c`** used when calling it.*

Instead of separate inputs, this function reads three variables using a **single input statement**. The values entered by the user must be **separated from each other by whitespace**, not commas. Spaces, tabs or newlines all work fine.

When **documenting your parameters**, you may **annotate** each of the parameters with the direction of the information flow: `@param[in]`, `@param[in,out]`, `@param[out]`. If you don't annotate the parameter, it is **assumed** to be an input parameter.