Lesson 6: Functions

3D Game Programming With C++

Digital Media Academy (Summer 2011)

Written by: Andrew Uzilov (andrew.uzilov@gmail.com)
Feel free to contact me with any questions.

Functions are pieces of code that have a name. They usually take an input and return an output (but not always). It is like you are asking a question (input) and getting and answer (output). To get a function to run, you have to call it, which we can do by using its name. Input is passed in as arguments (sometimes called parameters). But some functions take no input, or no output, or either! In that case, they usually modify the state of the program.

Function calls always have parentheses after their name. The parameters go between the parentheses.

Here are some sample functions calls:

```
// function “initGame” takes no input, gives no output
initGame();

// input is variable “score”, no output
updateHighScore (score);

// input to function convertTemp() is variable “tempC”
// output is saved into variable “tempF”
double tempF = convertTemp (tempC);
```

Just like variables, new functions have to be declared and defined:

```
// Declaration, no definition.
// This is called a “prototype”.
double convertTemp (double tempC);

// Declaration and definition.
double convertTemp (double tempC) {
    double tempF = (9.0 / 5.0) * tempC + 32.0;
    return tempF;
}
```

Now anytime we want to convert the temperature, we can just call the `convertTemp()` function, instead of having to remember the math!

You don’t have to worry about declaring and defining functions just yet. The important thing is that you recognize a function call when you see one, and that you know how to make function calls. You will be calling many useful functions in C++ and Panda3D libraries!