

8.1 函数

- 函数可以被看作一个由用户定义的操作。**Cg** 语言中的函数声明形式与**C\C++**中相同，由返回类型（**return type**）、函数名、形参列表（**parameter list**，位于括号中，并用逗号分隔的参数表）和函数体组成。函数体包含在花括号中。
- 如果没有返回值，则函数的返回类型是 **void**。
- 如果函数没有返回值，函数的返回类型一定要是 **void**，否则编译会出现大量的错误，错误信息的大概形式是：

```
error C0000: syntax error, unexpected ' (' at token "("  
error C0501: type name expected at token "("  
error C1110: function "main_v" has no return statement
```

8.1.1 数组形参

- 在 **C\C++**中，当一个数组作为函数的形参时，实际上传入的只是指向首元素的指针，并且数组边界被忽略。而在 **Cg** 语言中不存在指针机制（图形硬件不支持），数组作为函数形参，传递的是数组的完整拷贝。
- 数组类型变量作为函数形参，可以是一维的也可以是多维的，并且不必声明数组长度，即 **Unsize Array**。

```
float myFunc( float vals[])  
{  
    float sum = 0.0;  
    for(int i = 0; i < vals.length; i++)  
    {  
        sum += vals[i];  
    }  
    return sum;  
}
```

如果指定了函数中形参数组的长度，那么在调用该函数时实参数组的长度和形参数组的长度必须保持一致

8.2 函数重载

Cg 语言支持函数重载（**Function Overloading**），其方式和 **C++**基本一致，通过形参列表的个数和类型来进行函数区分。**Cg** 语言标准函数库中绝大部分函数都被重载过。

8.3 入口函数

所谓入口函数，即一个程序执行的入口，例如 **C\C++**程序中的 **main()**函数。

通常高级语言程序中只有一个入口函数，不过由于着色程序分为顶点程序和片断程序，两者对应着图形流水线上的不同阶段，所以这两个程序都各有一个入口函数。

8.4 CG 标准函数库

- 和 C 的标准函数库类似，Cg 提供了一系列内建的标准函数。这些函数用于执行数学上的通用计算或通用算法（纹理映射等），例如，需要求取入射光线的反射光线方向向量可以使用标准函数库中的 **reflect** 函数，求取折射光线方向向量可以使用 **refract** 函数，做矩阵乘法运算时可以使用 **mul** 函数。
- Cg 标准函数库主要分为五个部分：
 1. 数学函数（**Mathematical Functions**）；
 2. 几何函数(**Geometric Functions**)；
 3. 纹理映射函数(**Texture Map Functions**)；（作为单独的一章进行讲解）
 4. 偏导数函数(**Derivative Functions**)；
 5. 调试函数(**Debugging Function**)；

8.4.1 数学函数（**Mathematical Functions**）

下表中列举了 Cg 标准函数库中所有的数学函数，这些数学函数用于执行数学上常用计算，包括：三角函数、幂函数、圆函数、向量和矩阵的操作函数。这些函数都被重载，以支持标量数据和不同长度的向量作为输入参数。

8.4.2 几何函数（**Geometric Functions**）

几何函数，如下表所示，用于执行和解析几何相关的计算，例如根据入射光向量和顶点法向量，求取反射光和折射光方向向量。Cg 语言标准函数库中有3个几何函数会经常被使用到，分别是：

normalize 函数，对向量进行归一化；**reflect**函数，计算反射光方向向量；**refract** 函数，计算折射光方向向量。

重点#####

1. 着色程序中的向量最好进行归一化之后再使用，否则会出现难以预料的错误；
2. **reflect** 函数和 **refract** 函数都存在以“入射光方向向量”作为输入参数，注意这两个函数中使用的入射光方向向量，是从外指向几何顶点的；平时我们在着色程序中或者在课本上都是将入射光方向向量作为从顶点出发。