

6.1 关系操作符 (Comparison Operators)

关系操作符，用于比较同类型数据（不同类型的基础数据需要进行类型转换，不同长度的向量，不能进行比较）之间的大小关系或者等价关系。Cg中有6种 关系操作符，如表 1所示, 关系操作符运算后的返回类型均为bool类型。



Cg语言表达式允许对向量使用所有的boolean operator，如果是二元操作符，则被操作的两个向量的长度必须一致。表达式中向量的每个分量都进行一对一的 运算，最后返回的结果是一个bool类型的向量，长度和操作数向量一致。例如：`float3 a = float4(0.5, 0.0, 1.0); float3 b = float4(0.6, -0.1, 0.9); bool3 c = a < b;`

运算后向量c的结果为`float3(true, false, true);`

6.2 逻辑操作符 (Logical Operators)



Cg语言中有3种逻辑操作符(也被称为boolean Operators)，逻辑操作符运算后的返回类型均为bool类型。逻辑操作符也可以对向量使用，返回的变量类型是同样长度的内置bool向量。有一点需要注意：Cg中的逻辑与（&&）和逻辑或（||）不存在C中的短路现象（short-circuiting，即只用计算一个操作数的bool值即可），而是参与运算的操作数据都进行bool分析。

在Cg中，由于关系操作符以及逻辑操作符，都返回bool类型结果，所以这两种操作符有时也被统一称为boolean operator。

6.3 数学操作符 (Math Operators)

Cg语言对向量的数学操作提供了内置的支持，Cg中的数学操作符有：*乘法；/除法；-取反；+加法；-减法；%求余；++；——；*=；/=；+=；-=。

求余操作符%。只能在int类型数据间进行，否则编译器会提示错误信息：error C1021: operands to “%” must be integral.

6.4 移位操作符

Cg语言中的移位操作符，功能和C语言中的一样，也可以作用在向量上，但是向量类型必须是int类型。

6.5 Swizzle 操作符

可以使用Cg语言中的swizzle操作符 (.) 将一个向量的成员取出组成一个新的向量。swizzle操作符被GPU硬件高效支持。swizzle操作符后接x、y、z、w，分别表示原始向量的第一个、第二个、第三个、第四个元素；swizzle操作符后接r、g、b和a的含义与前者等同。不过为了程序的易读性，建议对于表示颜色值的向量，使用swizzle操作符后接r、g、b和a的方式。举例如下：float4(a, b, c, d).xyz 等价于 float3(a, b, c) float4(a, b, c, d).xxyy 等价于 float3(a, b, b) float4(a, b, c, d).wzyx 等价于 float4(d, c, b, a) float4(a, b, c, d).w 等价于 float d

Cg语言中float a 和float1 a是基本等价的，两者可以进行类型转换；float、bool、half等基本类型声明的变量也可以使用swizzle操作符。例如： float a = 1.0; float4 b = a.xxxx;:

swizzle操作符只能对结构体和向量使用，不能对数组使用,要从数组中取值必须使用[]符号。例如：
float a[3] = {1.0,1.0,0.0}; float b = a[0]; //正确 float c = a.x; //编译会提示错误信息(error C1010: expression left of "." x" is not a struct or array)

6.6 条件操作符 (Conditional Operators)

条件操作符的语法格式为： expr1 ? expr2 : expr3; expr1的计算结果为true或者false，如果是true,则expr2执行运算，否则expr3被计算。

Cg中的条件操作符一个独特的性能是：支持向量运算。即，expr1的计算结果可以是bool型向量，expr2和expr3必须是与expr1长度相同的向量。举例如下：

```
float3 h = float3(-1.0,1.0,1.0); float3 i = float3(1.0,0.0,0.0); float3 g = float3(1.0,1.0,0.0); float3 k; k = (h < float3(0.0,0.0,0.0))?(i):(g);
```

三元向量h与float3(0.0, 0.0, 0.0)做比较运算后结果为 (true, false, false) ,所以i的第一个数据赋值给K的第一个数据，g的第二个和第三个数据赋值给k的第二个和第三个数据，K的值为(1.0, 1.0, 0.0)。

6.7 操作符优先顺序

Cg语言中操作符的优先顺序从上到下表示从高级到低级的优先级；同一行的操作符具有同等优先级。



6.8 控制流语句 (Control Flow Statement)

Cg中的控制流语句和循环语句与C语言类似：

条件语句有：if、if-else；循环语句有：while、for。break语句可以和在for语句中使用。

如果没有确切的把握，尽量不要在低级的profiles中使用 循环控制语句。return只能作为最后一条语句出现。函数的递归调用（recursion）在Cg语言中是被禁止的。Switch、case和default在Cg中作为保留关键字存在，但是它们目前不被任何profile所支持。