C++结构体的赋值

变量的赋值和初始化是不一样的，初始化是在变量定义的时候完成的，是属于变量定义的一部分，赋值是在变量定义完成之后想改变变量值的时候所采取的操作。

还是给定结构体A:

struct A {

int b;

int c;

}

**注意：**结构体变量的赋值是不能采用大括号的方式进行赋值的，例如下面的赋值是不允许的。

struct A a;

//错误赋值

a={1,2}; （区分初始化和赋值。）

1>定义一个**普通结构体**类型X：

1 struct X

2 {

3 int i;

4 float f;

5 char c;

6 };

 使用X创建对象时，可以这样初始化其每个成员的值：

1 X x1 = { 1， 2.2, 'c' };

使用X来创建对象数组时，可以用嵌套的大括号 {{obj 1},{obj 2},{obj 3},…,{obj n}} 来初始化每一个对象：

1 X x2[3] = { {1, 1.1, 'a'}, {2, 2.2, 'b'}};

这里只列出了两个对象的初值，所以第三个对象中的数据成员分别被初始化为对应数据类型的*缺省初值①*

2>对于**包含数组的结构体**来说，初始化与上述类似，需要说明的一点是，当数组成员在结构体中定义在首位且给出不完全初始化表的情况下，初值是按照 数组的元素顺序 + 之后的成员表 依次被赋初值的，此处举例说明：

1 #include <iostream>

2 #include <cstdio>

3

4 using namespace std;

5

6 struct Y

7 {

8 int i[2];

9 float f;

10 char c;

11 };

12

13 int main()

14 {

15 Y yy={1};

16 Y y[3] = {1,2,3};

17 printf("%d %d %f #%c#%d#\n", yy.i[0], yy.i[1], yy.f, yy.c, yy.c);

18 /\*输出结果：1 0 0.000000 # #0# \*/

19 printf("%d %d %f\n", y[0].i[0], y[0].i[1], y[0].f);

20 /\*输出结果：1 2 3.000000 \*/

21 printf("%d %f #%d#\n", y[2].i[0], y[2].f, y[2].c);

22 /\*输出结果：0 0.000000 #0# \*/

23 return 0;

24 }

对于更**复杂结构体**也是如此，用{}逐级表明赋值关系即可。