

1 C++入门

C++ 读作“C加加”，是“C Plus Plus”的简称。顾名思义，C++ 是在C语言的基础上增加新特性，玩出了新花样，所以叫“C Plus Plus”，就像 iPhone 12 Plus 和 iPhone 11、Win10 和 Win7 的关系。

从语法上看，C语言是 C++ 的一部分，C语言代码几乎不用修改就能够以 C++ 的方式编译，这给很多初学者带来了不小的困惑，学习 C++ 之前到底要不要先学习C语言呢？

不过可以明确地说：学了C语言就相当于学了 C++ 的一半，从C语言转向 C++ 时，不需要再从头开始，接着C语言往下学就可以，所以码牛强烈建议先学C语言再学 C++。

1.1 C++和C语言的血缘关系

现在看来，C++ 和C语言虽然是两门独立的语言，但是它们却有着扯也扯不清的关系。

早期并没有“C++”这个名字，而是叫做“带类的C”。“带类的C”是作为C语言的一个扩展和补充出现的，它增加了很多新的语法，目的是提高开发效率，如果你有 [Java](#) Web 开发经验，那么你可以将它们的关系与 [Servlet](#) 和 [JSP](#) 的关系类比。

这个时期的 C++ 非常粗糙，仅支持简单的面向对象编程，也没有自己的编译器，而是通过一个预处理程序（名字叫 cfront），先将 C++ 代码“翻译”为C语言代码，再通过C语言编译器合成最终的程序。

随着 C++ 的流行，它的语法也越来越强大，已经能够很完善的支持面向过程编程、面向对象编程（OOP）和泛型编程，几乎成了一门独立的语言，拥有了自己的编译方式。

我们很难说 C++ 拥有独立的编译器，例如 Windows 下的微软编译器（cl.exe）、Linux 下的 [GCC](#) 编译器、Mac 下的 Clang 编译器（已经是 Xcode 默认编译器，雄心勃勃，立志超越 GCC），它们都同时支持C语言和 C++，统称为 C/C++ 编译器。对于C语言代码，它们按照C语言的方式来编译；对于 C++ 代码，就按照 C++ 的方式编译。

从表面上看，C、C++ 代码使用同一个编译器来编译，所以上面我们说“后期的 C++ 拥有了自己的编译方式”，而没有说“C++ 拥有了独立的编译器”。

1.2 C++类和对象到底是什么意思？

[C++](#) 是一门面向对象的编程语言，理解 C++，首先要理解**类 (Class)** 和**对象 (Object)** 这两个概念。

C++ 中的类 (Class) 可以看做C语言中结构体 (Struct) 的升级版。结构体是一种构造类型，

可以包含若干成员变量，每个成员变量的类型可以不同；可以通过结构体来定义结构体变量，每个变量拥有相同的性质。例如：

```
#include <stdio.h>
//定义结构体 Student
struct Student{
    //结构体包含的成员变量
    char *name;
    int age;
    float score;
};
//显示结构体的成员变量
void display(struct Student stu){
    printf("%s的年龄是 %d, 成绩是 %f\n", stu.name, stu.age, stu.score);
}
```

```

int main(){
    struct Student stu1;
    //为结构体的成员变量赋值
    stu1.name = "小明";
    stu1.age = 15;
    stu1.score = 92.5;
    //调用函数
    display(stu1);
    return 0;
}

```

运行结果:

小明的年龄是 15, 成绩是 92.500000

1.3 C++ 中的类也是一种构造类型

同时也对C++进行了一些扩展, 类的成员不但可以是变量, 还可以是函数;

通过类定义出来的变量也有特定的称呼, 叫做“对象”。例如:

```

#include <stdio.h>
//通过class关键字类定义类
class Student{
public:
    //类包含的变量
    char *name;
    int age;
    float score;
    //类包含的函数
    void say(){
        printf("%s的年龄是 %d, 成绩是 %f\n", name, age, score);
    }
};
int main(){
    //通过类来定义变量, 即创建对象
    class Student stu1; //也可以省略关键字class
    //为类的成员变量赋值
    stu1.name = "小明";
    stu1.age = 15;
    stu1.score = 92.5f;
    //调用类的成员函数
    stu1.say();
    return 0;
}

```

运行结果与上例相同。

class 和 public 都是 C++ 中的关键字, 初学者请先忽略 public (后续会深入讲解), 把注意力集中在 class 上。

c和c++的区别

1. C语言中的 struct 只能包含变量,
2. 而 C++ 中的 class 除了可以包含变量, 还可以包含函数。display() 是用来处理成员变量的函数, 在C语言中, 我们将它放在了 struct Student 外面, 它和成员变量是分离的;

3. 而在 C++ 中，我们将它放在了 class Student 内部，使它和成员变量聚集在一起，看起来更像一个整体。

结构体和类都可以看做一种由用户自己定义的复杂数据类型，

在C语言中可以通过结构体名来定义变量，在 C++ 中可以通过类名来定义变量。

不同的是：

通过结构体定义出来的变量还是叫变量，

而通过类定义出来的变量有了新的名称，叫做对象（Object）。

在第二段代码中，我们先通过 class 关键字定义了一个类 Student，然后又通过 Student 类创建了一个对象 stu1。变量和函数都是类的成员，创建对象后就可以通过点号 . 来使用它们。

可以将类比喻成图纸，对象比喻成零件，图纸说明了零件的参数（成员变量）及其承担的任务（成员函数）；

一张图纸可以生产出多个具有相同性质的零件，不同图纸可以生产不同类型的零件。

重点：

类只是一张图纸，起到说明的作用，不占用内存空间；

对象才是具体的零件，要有地方来存放，才会占用内存空间。

在 C++ 中，通过类名就可以创建对象，即将图纸生产成零件，这个过程叫做类的实例化，因此也称对象是类的一个实例（Instance）。

有些资料也将类的成员变量称为属性（Property），将类的成员函数称为方法（Method）。

面向对象编程（Object Oriented Programming, OOP）

类是一个通用的概念，C++、Java、C#、PHP 等很多编程语言中都支持类，都可以通过类创建对象。可以将类看做是结构体的升级版，C语言的晚辈们看到了C语言的不足，尝试加以改善，继承了结构体的思想，并进行了升级，让程序员在开发或扩展大中型项目时更加容易。

因为 C++、Java、C#、PHP 等语言都支持类和对象，所以使用这些语言编写程序也被称为面向对象编程，这些语言也被称为面向对象的编程语言。C语言因为不支持类和对象的概念，被称为面向过程的编程语言。

在C语言中，我们会把重复使用或具有某项功能的代码封装成一个函数，将拥有相关功能的多个函数放在一个源文件，再提供一个对应的头文件，这就是一个模块。使用模块时，引入对应的头文件就可以。

而在 C++ 中，多了一层封装，就是类（Class）。类由一组相关联的函数、变量组成，你可以将一个类或多个类放在一个源文件，使用时引入对应的类就可以。下面是C和C++项目组织方式的对比：

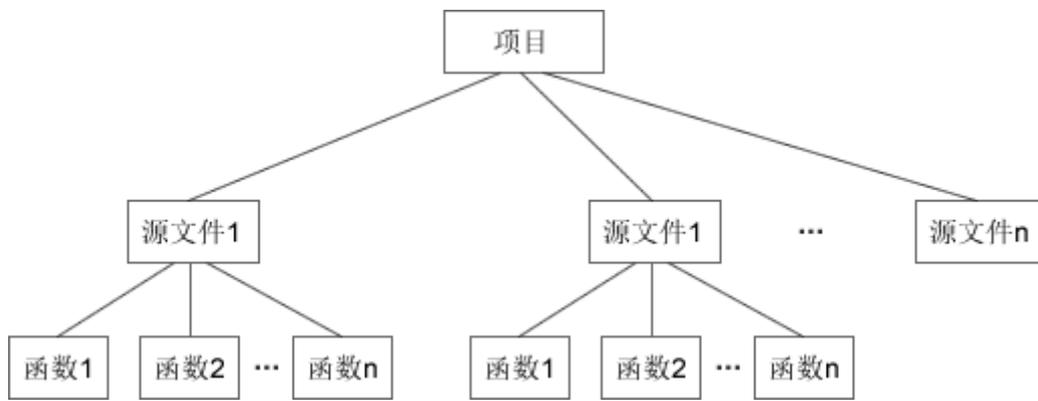


图1: C语言中项目的组织方式

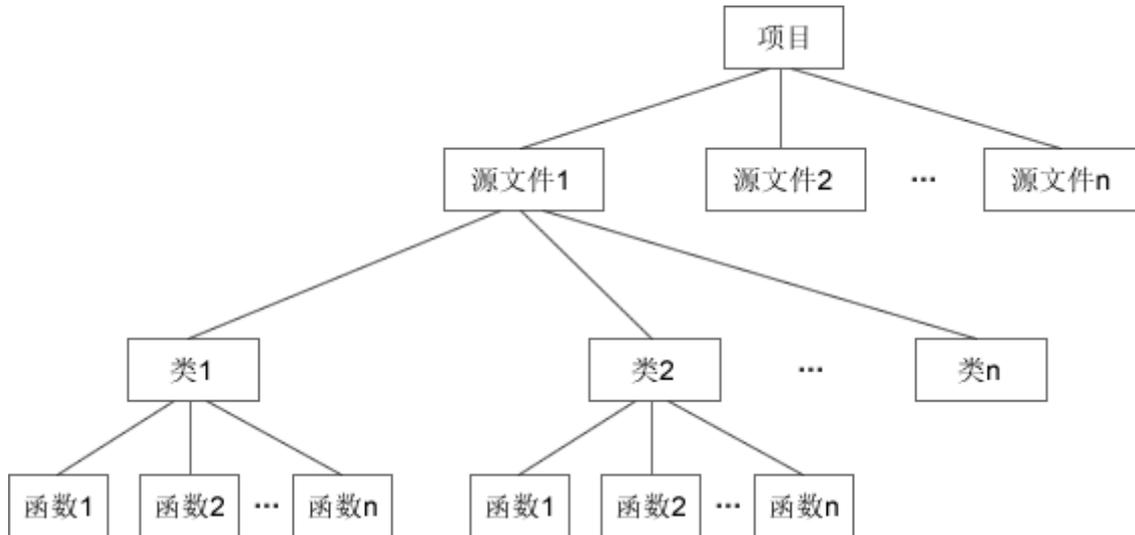


图2: C++中项目的组织方式

不要小看类 (Class) 这一层封装，它有很多特性，极大地方便了中大型程序的开发，它让 C++ 成为面向对象的语言。

面向对象编程在代码执行效率上绝对没有任何优势，它的主要目的是方便程序员组织和管理代码，快速梳理编程思路，带来编程思想上的革新。

面向对象编程是针对开发中大规模的程序而提出来的，目的是提高软件开发的效率。不要把面向对象和面向过程对立起来，面向对象和面向过程不是矛盾的，而是各有用途、互为补充的。如果你希望开发一个贪吃蛇游戏，类和对象或许是多余的，几个函数就可以搞定；但如果开发一款大型游戏，那你绝对离不开面向对象。

2 cin和cout简介

定义:

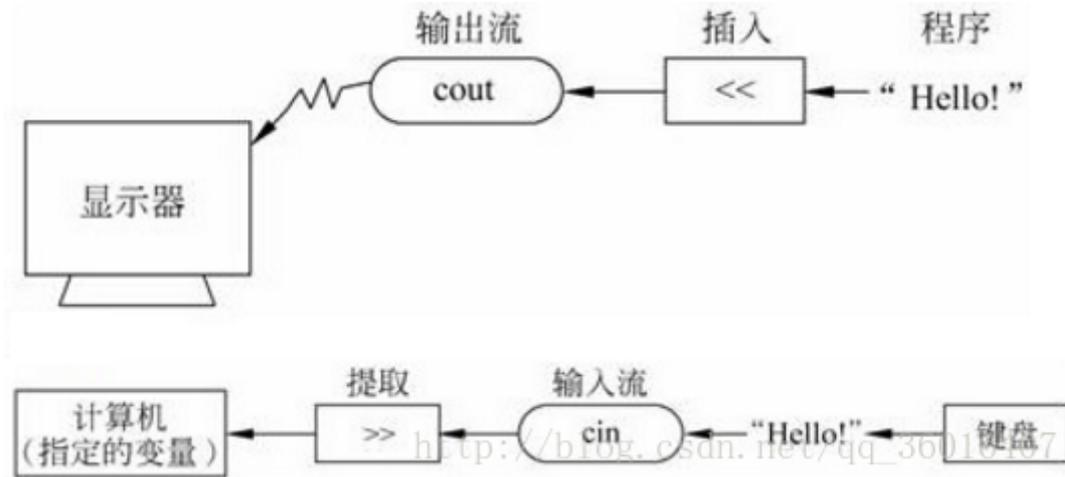
1. cin是C++编程语言中的标准输入流对象，即**istream类的对象**。cin主要用于从标准输入读取数据，这里的标准输入，指的是终端或键盘。
2. cout是流的对象，即**ostream类的对象**
3. cerr是标准错误输出流的对象，也是ostream类的对象。这里的标准输出指的是终端键盘，标准错误输出指的是终端的屏幕。

在理解cin功能时，不得不提标准输入缓冲区。当我们从键盘输入字符串的时候需要敲一下回车键才能够将这个字符串送入到缓冲区中，

那么敲入的这个回车键(\r)会被转换为一个换行符\n，这个换行符\n也会被存储在cin的缓冲区中并且被当成一个字符来计算！

比如我们在键盘上敲下了123456这个字符串，然后敲一下回车键 (\r) 将这个字符串送入了缓冲区中，那么此时缓冲区中的字节个数是7，而不是6。

cin读取数据也是从缓冲区中获取数据，缓冲区为空时，cin的成员函数会阻塞等待数据的到来，一旦缓冲区中有数据，就触发cin的成员函数去读取数据。



有关流对象cin、cout和流运算符的定义等信息是存放在C++的输入输出流库中的，因此如果在程序中使用cin、cout和流运算符，就必须使用预处理命令把头文件iostream包含到本文件中，并使用命名空间std：

1.2.cin和cout的基本操作

cout表达式

cout语句的一般格式为：

```
cout<<表达式1<<表达式2<<表达式3...<<表达式n;
```

cin表达式

cin语句的一般格式为：

```
cin>>变量1>>变量2>>变量3>>.....>>变量n;
```

1.2.1cout 操作

一个cout语句可以分成若干行

```
cout<<"this is" //注意没有分号
    <<"a C++"
    <<"program."
    <<endl;
```

或者这样写 也是等效的

```
cout<<"this is "; //有分号
cout<<"a C++";
cout<<"program.";
cout<<endl;
```

输出结果差不多。

不能用一个插入运算符"<<"插入多个输出项：

1.2.2 在用cout输出时，用户不必通知计算机按何种类型输出

系统会自动判别输出数据的类型，使输出的数据按相应的类型输出。如已定义a为int型，b为float型，c为char型，则：cout<<a<<' '<<b<<' '<<c<<endl;

cin: 与cout类似，cin语句可以分成若干行

```
1)cin>>a>>b>>c>>d;
```

```
cin>>a //这样的写法比较清晰
  >>b
  >>c
  >>d;
```

```
cin>>a;
cin>>b;
cin>>c;
```

这几种写法都是一样的效果

1.2.3在用cin输入时,系统也会根据变量的类型从输入流中提取相应长度的字节。

如有：

```
char c1,c2;int a;

float b;

cin>>c1>>c2>>a>>b;
```

不能用cin语句把空格字符和回车换行符作为字符输入给字符变量，他们将被跳过。