# C++概述

* 1. C++间接

C++中的++是递增运算符，叫“C with class”。C++程序可以使用已有的C程序库。C++语言在C语言的基础上添加了面向对象编程和泛型编程的支持（两大编程思想）。C++是可移植的。C++是C语言的加强，是一种更好的C语言。

* 1. C++起源

C++起源于贝尔实验室。

* 1. 可移植性和标准

对于硬件，针对特定硬件编程的程序是不可移植的。对于编程语言，不同的系统下的实现不一定相同。因此美国国家标准局制定了一个C++标准。

C++的第一套标准即C++98，即1998年制定的。C++11同理。C++完全兼容C的程序。

* 1. 为什么C++会成功？

C语言转到C++花费的代价会比较小。C++是C语言的扩展。

# 初识C++

* 1. 简单的C++程序
     1. 引入头文件 #include<iostream> 标准输入输出流
     2. 使用标准命名空间 using namespace std;
     3. 标准输出流对象cout<< <<endl;
     4. 引入系统头文件，C中<stdio.h> <math.h> ，C++中<cmath> 。但是两种都可以。
     5. 面向对象的三大特性：封装、继承、多态。

在面向对象中，算法和数据结构被看作一个整体，称为对象。

* 1. 双冒号作用域运算符号
     1. ::代表作用域 如果前面什么都不加 代表全局作用域
  2. namespace命名空间
     1. 命名空间用途：解决名称冲突
     2. 命名空间下可以存放：变量、函数、结构体、类......
     3. 命名空间必须声明在全局作用域
     4. 命名空间可以嵌套命名空间
     5. 命名空间是开放的
     6. 命名空间可以是匿名的
     7. 命名空间可以起别名
  3. using声明及using编译指令
     1. using声明

using KingGlory::sunwukongId ;

使用KingGlory下的变量sunwukongId。后面就可以直接使用sunwukongId，而不需要使用双引号类型。当using声明与就近原则同时出现时，出错，应避免。

int sunwokongId=20;

using KingGlory::sunwukongId ;

cout<<sunwukongId; //这种是出错的

* + 1. using编译指令

using namespace std;

//当using编译指令 与 就近原则同时出现，优先使用就近

int sunwokongId=20;

using namespace KingGlory;

cout<<sunwokongId; // 结果是新定义的sunwukongId 为 20

//当using编译指令有多个，需要加作用域 区分

* 1. C++对C语言增强以及扩展
     1. 全局变量检测增强

int a

int a=10 C下可以 C++重定义 出错

* + 1. 函数检测增强
       1. 函数的返回值
       2. 参数类型
       3. 函数调用参数个数

C语言下这些要求不严格，C++下要求严格

getA(w,h){return w\*h;}//C语言下这个函数是可以编译的

int a=getA(1，2，3，4);//C语言下这个调用也是可行的

* + 1. 类型转换检测增强

char \*p=(char \* ) malloc(64);

C++下必须等号左右一致类型，C语言下则不需要强制类型转化

* + 1. struct增强
       1. C++可以在结构体中放函数
       2. C++创建结构体变量，可以简化关键字struct
    2. bool数据类型扩展
       1. C++才有bool类型
       2. true代表真，false代表假
       3. sizeof(bool)=1
    3. 三目运算符增强 ？：
       1. C语言下返回的是值
       2. C++下返回的是变量
    4. const增强

const 代表声明一种常量，而常量是不可更改的

* + - 1. C语言下（伪常量）
         1. 全局const，直接修改，失败；间接修改，语法通过，运行失败。（直接修改是变量名修改，间接修改是通过指针修改）
         2. 局部const，直接修改，失败；间接修改，成功。
      2. C++语言下
         1. 全局const和C结论一样
         2. 局部const，直接修改失败，间接修改也失败
         3. C++下const可以称为常量。可以 int arr[num];//num为const int 常量。
    1. const链接属性
       1. C语言下const修饰的全局变量默认是外部链接属性。
       2. C++下的const修饰的全局变量默认是内部链接属性，可以加extern提高作用域。
    2. const分配内存情况

const定义的常量是在符号表上，类似于键值对的 key value

* + - 1. 对const变量取地址，会分配临时内存
      2. 使用普通变量 初始化const变量
      3. 对于自定义数据类型
    1. 尽量使用const代替define 宏
       1. #define的宏常量，没有数据类型，不重视作用域；const是有数据类型的
    2. 引用
       1. 目的：起别名
       2. 语法：类型（与原名类型必须一致） &别名 = 原名
       3. 引用必须初始化
       4. 引用一旦初始化后，就不可以引向其他变量，但是值可以改变
       5. 建立对数组引用
          1. 直接建立引用
          2. int arr[10];
          3. int(&pArr)[10]=arr;// pArr 就是 arr
       6. 先定义出数组类型，再通过类型 定义引用
          1. typedef int(array\_type)[10];//array\_type是一种数组类型
          2. array\_type &pArr2=arr;//实现引用