

4. C语言和C++的区别与联系

差异1:

1. C语言面向过程, C++面向对象 **面向过程**: 面向过程编程就是分析出解决问题的步骤, 然后把这此步骤一步一步的实现, 使用的时候一个一个的依次调用就可以了。

1. 考虑的是实际地实现.
2. 一般从上往下步步求精.
3. 最重要的是模块化的思想方法. 当程序规模不是很大时, 面向过程的方法还会体现出一种优势, 因为程序的流程很清楚, 按着模块与函数的方法可以很好的组织. **面向对象**: 面向对象编程就是把问题分解成各个对象, 建立对象的目的不是为了完成一个步骤, 而是为了描述某个事物在整个解决问题的步骤中的行为.
4. 封装: 把客观事物抽象成类, 每个类对自己的数据和方法进行访问权限保护
5. 继承: 可使用现有类的所有功能, 在无需重新编写原来的类的情况下对这些功能进行扩展.
6. 多态: 可以简单地概括为“一个接口, 多种方法”, 程序在运行时才决定调用的函数, 面向对象的核心, 多态的目的则是为了接口重用.

面向过程就是自顶向下的编程 (步骤划分)、面向对象就是高度实物抽象化 (功能划分)

差异2:

语法上

1. c++具有重载, 继承, 多态三种特性.
 2. c++增加了许多类型安全功能 (如强制类型转换)
- 类型安全代码指访问被授权可以访问的内存位置.
 - 例如, 类型安全代码不能从其他对象的私有字段读取值. 它只从定义完善的允许方式访问类型才能读取. c++中:
 - (1)操作符new返回的指针类型严格与对象匹配, 而不是void*;
 - (2)C中很多以void*为参数的函数可以改写为C++模板函数, 而模板是支持类型检查的;
 - (3)引入const关键字代替#define constants, 它是有类型、有作用域的, 而#define constants只是简单的文本替换;
 - (4)一些#define宏可被改写为inline函数, 结合函数的重载, 可在类型安全的前提下支持多种类型, 当然改写为模板也能保证类型安全;
 - (5) C++提供了dynamic_cast关键字, 使得转换过程更加安全, 因为dynamic_cast比 static_cast涉及更多具体的类型检查.

基于上述的例子, 二者的优劣总结如下:

- **面向过程语言**

优点: 性能比面向对象高, 因为类调用时需要实例化, 开销比较大, 比较消耗资源;比如单片机、嵌入式开发、Linux/Unix等一般采用面向过程开发, 性能是最重要的因素。

缺点: 没有面向对象易维护、易复用、易扩展

- **面向对象语言:**

优点：易维护、易复用、易扩展，由于面向对象有封装、继承、多态性的特性，可以设计出低耦合的系统，使系统 更加灵活、更加易于维护

缺点：性能比面向过程低

参考资料：

1. [C语言和C++的区别与联系（详细）_cherrydreamsover的博客-CSDN博客_c语言和c++的区别和联系](#)
2. [c++和c的区别_哈哈哈哈哈哈哈的博客-CSDN博客_c++和c的区别](#)