C++补充

* 补充知识
  + 语言对比
    - C和C++
      * 设计思想
        + 面向对象和面向过程
      * 语法
        + C++有继承多态封装
        + 增加许多类型安全功能，比如强制类型转换
        + C++支持范式编程，比如函数模板，类模板
        + C语言不支持重载

函数指针

传入void\*指针，参数无类别化

可变参数列表

* + - * + 有异常处理机制
    - Java和C++
      * Java为解释性语言，程序源代码经过Java编译器编译成字节码，然后由JVM解释执行。而C/C++为编译型语言，源代码经过编译和链接后生成可执行的二进制代码
      * C++用析构函数回收垃圾，写C和C++程序时一定要注意内存的申请和释放 Java语言不使用指针，内存的分配和回收都是自动进行的，程序员无须考虑内存碎片的问题
    - Python和C++
      * Python是一种脚本语言，是解释执行的，而C++是编译语言，是需要编译后在特定平台运行的。python可以很方便的跨平台，但是效率没有C++高。
      * Python使用缩进来区分不同的代码块，C++使用花括号来区分
      * C++中需要事先定义变量的类型，而Python不需要，Python的基本数据类型只有数字，布尔值，字符串，列表，元组等等
      * Python的库函数比C++的多，调用起来很方便
  + C和C++中struct
    - C语言中：struct是用户自定义数据类型（UDT）
    - C++中struct是抽象数据类型（ADT）
    - 在C中必须在结构标记 前加上struct，才能做结构类型名
    - C++中的struct能继承，能实现多态
  + 判断结构体是否相等
    - 重载“==”操作符
    - 使用memcmp
  + 可变参数模板
  + switch中case里不能自定义变量
  + char和int转换
    - char c='5';int n = c-'0';
    - int n=5;char c=n+'0';
  + malloc的底层实现
  + C语言检索内存情况
  + 异常处理try/catch
  + fork()
  + epoll