

C 代码优化 (2) PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/459/2021_2022_C___E4_BB_A3_E7_A0_81_E4_c97_459857.htm 尽可能使用常量(const) 尽可能使用常量(const)。C 标准规定，如果一个const声明的对象的地址不被获取，允许编译器不对它分配储存空间。这样可以使代码更有效率，而且可以生成更好的代码。提升循环的性能 要提升循环的性能，减少多余的常量计算非常有用（比如，不随循环变化的计算）。不好的代码(在for()中包含不变的if()) 推荐的代码 for(i ... { if(CONSTANT0 { DoWork0(i ; // 假设这里不改变CONSTANT0的值 } else { DoWork1(i ; // 假设这里不改变CONSTANT0的值 } } if(CONSTANT0 { for(i ... { DoWork0(i ; } } else { for(i ... { DoWork1(i ; } } 如果已经知道if()的值，这样可以避免重复计算。虽然不好的代码中的分支可以简单地预测，但是由于推荐的代码在进入循环前分支已经确定，就可以减少对分支预测的依赖。把本地函数声明为静态的(static) 如果一个函数在实现它的文件外未被使用的话，把它声明为静态的(static)以强制使用内部连接。否则，默认的情况下会把函数定义为外部连接。这样可能会影响某些编译器的优化比如，自动内联。考虑动态内存分配 动态内存分配（C 中的" ; new" ; ）可能总是为长的基本类型（四字对齐）返回一个已经对齐的指针。但是如果不能保证对齐，使用以下代码来实现四字对齐。这段代码假设指针可以映射到 long 型。例子 double* p = (double*)new BYTE[sizeof(double) * number_of_doubles 7L] ; double* np = (double*)((long(p) 7L) & amp. ; 8L) ; 现在，你可以使用 np 代替 p 来访问数据。注

意：释放储存空间时仍然应该用0delete p。 100Test 下载频道
开通，各类考试题目直接下载。详细请访问 www.100test.com