

hanchaoman的专栏

目录视图

摘要视图

RSS 订阅

个人资料



hanchaoman



访问： 233619次
积分： 3932分
排名： 第2194名

原创： 173篇 转载： 40篇
译文： 0篇 评论： 83条

文章搜索

文章分类

- C++ (16)
- C语言+编程 (51)
- DSP (6)
- gdb调试 (7)
- linux 多线程 (7)
- linux多线程同步 (11)
- linux系统 (26)
- makefile (7)
- QT (2)
- TCP/IP (18)
- VxWorks (1)
- XML (7)
- 数据结构 (2)
- 计算机原理 (5)
- linux驱动开发 (33)
- com (1)
- Python (2)
- Bootstrap和Uboot (10)
- 汇编语言 (7)
- stm32&rt_thread (0)

文章存档

2014年07月 (1)

有奖征资源，博文分享有内涵 6月推荐文章汇总 微软Azure•英雄会编程大赛题 关注CSDN社区微信，福利多多 社区问答：叶劲峰 游戏引擎架构

c语言中的的内联(inline)函数

分类： C语言+编程 2012-02-18 18:43 902人阅读 评论(0) 收藏 举报

语言 c 编译器 recursion 性能优化 gcc

目录(?) [+]

c语言中的的内联(inline)函数

发表于 2011 年 01 月 26 日 由 admin

c/c++中的inline，使用在函数声明处，表示程序员请求编译器在此函数的被调用处将此函数实现插入，而不是像普通函数那样生成调用代码(申请是否有效取决于编译器)。一般地说，这样作的优点是省掉了调用函数的开销；缺点则是可能会增加代所生成目标代码的尺寸(二班的除外，二班情况下，inline函数甚至会返过来降低程序的性能)。

实际上，即使没有手工指定inline函数，编译器一般还会选择一些代码量较小但使用频繁的函数作为inline函数，以此作为性能优化的途径之一。

1. 和带参宏定义(Parameterized Macro)的比较

与带参宏定义相比，inline函数具备以下优点：

- 参数类型检查：宏定义中所使用的参数仅仅是在宏定义中被替换，不进行任何的类型检查
- 返回值：宏定义中无法使用return返回
- 便于调试

2. 不同编译器下的inline关键字

尽管c/c++有着自己的语言标准和规范，但不同编译器实现中总会有着这样或那样的区别。inline即是一例。

c99标准

- inline：用于同一c/cpp文件内部被调用处展开；对外部文件来说函数不可用
- static inline：用于在同一c/cpp文件内部被调用处展开；一般情况下，编译器并不会为此函数生成单独的目标代码；如遇到内联函数无法展开，或内联函数以地址形式被调用，则编译器将会为此内联函数生成单独的代码；

简单地讲，c99中inline关键字申明的函数一般仅用于同一文件，函数本身不会生成单独的目标代码；static关键字修订后，如果需要，则会生成单独的目标代码。

gcc

- inline：对同一c/cpp文件，函数将会在被调用处展开；对外部文件，此函数等同于"extern"函数
- static inline：与c99标准中相同
- extern inline：仅用于同一c/c++文件内部，在被调用处展开

gcc中的inline关键字与c99中不同，默认情况下(仅使用inline)，在同一文件中被调用处当作内联函数展开，而在外部文件调用中等同于普通extern函数(也就是说会生成单独的目标代码)；加static关键字修订后，反而不可应用于外部文件，但如果需要可以生成单独的目标代码；gcc扩展的extern inline模式更是缩小函数的使用仅限于在同文件中展开。

ms vc

ms vc中inline的含义基本与c99中相同，此基础上，作了一定扩展

2013年12月 (2)

2013年11月 (1)

2013年07月 (2)

2013年06月 (11)

展开

阅读排行

c语言结构体用法

(29261)

C语言转义符

(9334)

非阻塞connect()和accept

(5701)

C语言多个源文件编译成

(4864)

一个makefile调用多个m

(4571)

C语言指针的指针作用 (

(4568)

wait_event_interruptible

(4346)

gdb多线程调试1

(3784)

LINUX 多线程互斥量和读

(3326)

浅谈C中静态变量与全局

(3006)

评论排行

C语言转义符

(10)

c语言结构体用法

(8)

Makefile实例

(5)

指针的指针作用 (申请空

(3)

浅谈C中静态变量与全局

(3)

C语言多个源文件编译成

(3)

关于TCP流模式与UDP数

(3)

Makefile编译选项

(2)

Linux设备驱动编程之pol

(2)

kernel 中断详解

(2)

推荐文章

▮ `__inline` : 等同于`inline`

▮ `__forceinline` : 强制编译器将函数作为内联函数, 除非以下情况

▮ 使用了`/Ob0`选项编译 (debug编译默认取值)

▮ 函数使用了可变参数

▮ 函数本身为递归函数, 且未使用`#pragma inline_recursion(on)`

▮

▮ 函数为虚(virtual)函数

▮ 程序中使用了地址方式调用了函数

▮ ...

3. 其他

一般地, 内联函数不能是递归函数或调用递归函数 (递归调用会给函数体展开带来麻烦)。

`inline`应当是用于函数声明(Declaration)而非函数实现(Implementation)。但由于由于不同的编译器下`inline`关键字所修饰函数的调用范围不一致, 因此, 一种简单的作法是, 将`inline`函数的声明和实现合一。如果确定函数仅应用于同一文件, 我们可以将函数的声明和实现都放在`.c/.cpp`中; 否则将其放入`.h`中 (这样, 包含了此头文件的任何文件都可以使用它)。

更多 1

上一篇 强制内联和强制不内联

下一篇 高级字符设备驱动--中断下半部机制之tasklet (一)

主题推荐

c语言 性能优化 recursion 程序员 递归

猜你在找

TI-Davinci开发系列之三测试XDS560V2仿真器与一擦烙饼的排序 (0621)

MMX指令集



携手聚力合作 共赢智慧云机

IBM SoftLayer 渠道合作伙伴招募会

上海站

8月12日 (星期二)

上海裕景大酒店

点击报名



Bison生成文件分析

UNIX域套接字

Bluetooth: ATT and GATT

error: unknown field 'ioctl' specified in initializer

Source not found

T培训,120天入门到精通

30000年薪就业!我们真退费

果吧】10分钟爱上编程

3 西部数码云服务器, 45元!

6 智能家居加盟首选科比迪

9 工资不够花, 10000元小本开店

来源: c语言

shenck1992: 第四条不正确, 设置是有效的。

I/O 端口和 I/O 内存

zchyag: LDD里面的内容, 做了一些扩展

I/O 端口和 I/O 内存

jackyzjk: 介绍的很详细, 受教了, 非常感谢。真没想到博主居然是一位girl, 太了不起了。

C++构造函数初始化列表

Cyan_: 赞!

浅谈C中静态变量与全局变量初始

z427222: 涨姿势了

组合, 关联, 聚合的区别

秦伟H: 你所有的“原创”都是copy的吗?

DOM和SAX比较和选择

夜小楼: 了解一下!

C++ virtual关键字

Tangbzh: 不错

返回const指针与返回const引用

Tangbzh: good

C++ 命名空间 using namespace

Tangbzh: 很好, get

您还没有登录,请[登录]或[注册]

* 以上用户言论只代表其个人观点, 不代表CSDN网站的观点或立场

核心技术类目

全部主题	Hadoop	AWS	移动游戏	Java	Android	iOS	Swift	智能硬件	Docker	OpenStack
VPN	Spark	ERP	IE10	Eclipse	CRM	JavaScript	数据库	Ubuntu	NFC	WAP
jQuery	BI	HTML5	Spring	Apache	.NET	API	HTML	SDK	IIS	Fedora
XML	LBS	Unity	Splashtop	UML	components	Windows Mobile	Rails	QEMU	KDE	Cassandra
CloudStack	FTC	coremail	OPhone	CouchBase	云计算	iOS6	Rackspace	Web App	SpringSide	Maemo
Compuware	大数据	apttech	Perl	Tornado	Ruby	Hibernate	ThinkPHP	HBase	Pure	Solr
Angular	Cloud Foundry	Redis	Scala	Django	Bootstrap					

2014年8月2日

c语言中的的内联(inline)函数 - hanchaoman的专栏 - 博客频道 - CSDN.NET

网站客服 杂志客服 微博客服 webmaster@csdn.net 400-600-2320

京 ICP 证 070598 号

北京创新乐知信息技术有限公司 版权所有

江苏乐知网络技术有限公司 提供商务支持

Copyright © 1999-2014, CSDN.NET, All Rights Reserved 