

平凡的程序员

喜欢编程、算法、虚拟机、搜索引擎、编译器、网络、游戏设计、操作系统等所有计算机中好玩的东西

目录视图

摘要视图

RSS 订阅

个人资料



feixiaoxing

恒

访问：1555211次

积分：19072分

排名：第111名

原创：296篇 转载：0篇

译文：0篇 评论：2085条

文章搜索

文章分类

代码测试 (4)

C/C++ (31)

数据结构和算法 (65)

多线程编程 (20)

c语言和设计模式 (26)

搜索引擎的那些事 (8)

Linux开发 (17)

操作系统 (28)

随想录 (62)

第三方库编译 (3)

linux驱动编写 (5)

verilog学习记 (4)

从skyyeye学习arm (5)

fltk学习记 (17)

文章存档

2014年06月 (3)

2014年05月 (16)

2014年04月 (2)

2014年03月 (1)

有奖征资源，博文分享有内涵 5月推荐博文汇总 大数据读书汇--获奖名单公布 2014 CSDN博文大赛

linux下的C语言开发（makefile编写）

分类：Linux开发 2012-01-12 18:22 11772人阅读 评论(9) 收藏 举报

makefile

linux

语言

c

工具

测试

【声明：版权所有，欢迎转载，请勿用于商业用途。联系信箱：feixiaoxing @163.com】

对于程序设计员来说，makefile是我们绕不过去的一个坎。可能对于习惯Visual C++的用户来说，是否会编写makefile无所谓。毕竟工具本身已经帮我们做好了全部的编译流程。但是在Linux上面，一切变得不一样了，没有人会为你做这一切。编码要靠你，测试要靠你，最后自动化编译设计也要靠你自己。想想看，如果你下载了一个开源软件，却因为自动化编译失败，那将会在很大程度上打击你学习代码的自信心了。所以，我的理解是这样的。我们要学会编写makefile，至少会编写最简单的makefile。

首先编写add.c文件，

```
[cpp]
01. #include "test.h"
02. #include <stdio.h>
03.
04. int add(int a, int b)
05. {
06.     return a + b;
07. }
08.
09. int main()
10. {
11.     printf(" 2 + 3 = %d\n", add(2, 3));
12.     printf(" 2 - 3 = %d\n", sub(2, 3));
13.     return 1;
14. }
```

再编写sub.c文件，

```
[cpp]
01. #include "test.h"
02.
03. int sub(int a, int b)
04. {
05.     return a - b;
06. }
```

最后编写test.h文件，

```
[cpp]
01. #ifndef _TEST_H
02. #define _TEST_H
03.
04. int add(int a, int b);
05. int sub(int a, int b);
06. #endif
```

那么，就是这三个简单的文件，应该怎么编写makefile呢？

2014年02月 (2) 展开

阅读排行

linux下的C语言编程（总

(50299)

一步一步写算法（之 算法

(47193)

一步一步写算法（之hasl

(43308)

用汇编的眼光看C++（之

(29959)

多线程的那点儿事（基础

(28561)

linux下的C语言开发（网

(23823)

多线程的那点儿事（之大

(23799)

一步一步写算法（之单向

(23744)

一步一步写算法（之快速

(23255)

随想录（程序员和收入）

(22290)

推荐文章

最新评论

一步一步写算法（之hash表）

Brightziu: data % 10 这个为啥要取余数呢？

多线程的那点儿事（基础篇）

EbowTang: 草，牛逼了！我就邪恶的全盘学习了！

随想录（移动app下的生活）

zistym: 为什么民营企业的app要比国营公司的app好用得多？像农业银行的app,太烂了。国营的基本是服务领...

随想录（从apple的swift语言说起

w8708812: 博主，你好，我从去年开始拜读博文的文章了，收获很大，“我们要记住技术是为业务服务的，只有真正给客户创...

linux下的C语言开发（动态库）

AmelieDan: 简单，干练

多线程的那点儿事（之数据互斥）

springontime: 或者我认为应该写成while(flag && (turn == index))

C语言和设计模式（继承、封装、

qq584253394: 恩。其实都是主要靠指针来实现抽象了。

一步一步写算法（之A*算法）

ios-开发者: 不知道 那两个函数传入的value数组从哪里来得，想请教

用汇编的眼光看C++（之虚函数）

xueerfei: LZ这块是不是有点小问题，调用顺序应该是：edx->->[, ecx里的内容是ptr，所以是两...

一步一步写算法（之洗牌算法）

wocaonilaozi: 楼主的程序每次运行后的结果都是一样的，因为他使用的是系统默认的种子，rand()使用前要有种子；应该...

友情链接

酷壳

云风

李先静

刘未鹏

高德纳

孟岩

潘爱民

周伟明

gamers

codeproject

[cpp]

01. test: add.o sub.o

02. gcc -o test add.o sub.o

03.

04. add.o: add.c test.h

05. gcc -c add.c

06.

07. sub.o: sub.c test.h

08. gcc -c sub.c

09.

10. clean:

11. rm -rf test

12. rm -rf *.o

上一篇

linux下的C语言开发（开篇）

下一篇

linux下的C语言开发（gdb调试）

更多

0

顶

13

踩

1

主题推荐

c语言

makefile

linux

开源软件

自动化

猜你在找

搜索引擎的那些事（网页下载）

一步一步写算法（之双向链表）

一步一步写算法（之爬楼梯）

使用socket的Linux上的C语言文件传输顺序服务器和

【Boost】boost::string_algo详解7——join的应用

用Powershell调用DLL文件

一步一步写算法（之二叉树广度遍历）

如何删除数据库中的冗余数据

ubuntu 10.10 添加系统调用全过程

如何在Linux下用eclipse开发C/C++程序

免费学习IT4个月,月薪12000

中国[官方授权]IT培训与就业示范基地,学成后名企直接招聘,月薪12000起!

查看评论

7楼 dragonAhao 2014-04-21 08:12发表

看不懂哦，亲

6楼 小塔-皂荚花 2013-12-01 19:16发表

clean:

rm -rf test

rm -rf *.o

我的最后这两句怎么没成功呢，在目录下还是生产了.o文件啊。

5楼 syrchina 2012-02-29 22:38发表

C 学习!

4楼 guoxiaojie_415 2012-02-29 10:34发表

C 楼主总结的很详细，有利于初学者学习巩固呀！顶！

3楼 nmgzhangds 2012-02-26 15:53发表

转载学习了。

2楼 practiceperfect 2012-02-19 17:48发表

期待更详细的总结

http://blog.csdn.net/feixiaoxing/article/details/7197095

2/3

华清远见
51cto
IT 人生
raw-os



1楼 [ybgwxf2008](#) 2012-01-18 09:25发表



这个太简单啦

编译器参数选项，动态库，静态库的包含，编译优化，头文件包含，**makefile**调用**makefile**都没有，实用性不高...

Re: [qinxiongxu](#) 2012-02-20 09:32发表



回复[ybgwxf2008](#)：人家是介绍最简单的**makefile**的编写，复杂的**makefile**再相应的拓展开来即可啦~

Re: [dfsgsfid](#) 2013-03-01 22:34发表



回复[qinxiongxu](#)：`test: add.o sub.o`
`gcc -o test add.o sub.o`
不过怎么也要说说上面一行是指文件依赖吧
进来的肯定是**makefile**小白啊
不过网络上**MAKEFILE**的教程不少
虽然我是楼主的粉丝 不过楼主这篇文章没啥内容

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 [Java](#) [VPN](#) [Android](#) [iOS](#) [ERP](#) [IE10](#) [Eclipse](#) [CRM](#) [JavaScript](#) [Ubuntu](#) [NFC](#)
[WAP](#) [jQuery](#) [数据库](#) [BI](#) [HTML5](#) [Spring](#) [Apache](#) [Hadoop](#) [.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](#) [Spark](#) [HBase](#) [Pure](#) [Solr](#) [Angular](#) [Cloud Foundry](#) [Redis](#) [Scala](#)
[Django](#) [Bootstrap](#)

公司简介 | 招贤纳士 | 广告服务 | 银行汇款帐号 | 联系方式 | 版权声明 | 法律顾问 | 问题报告 | 合作伙伴 | 论坛反馈
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

