

[High一下!](#)

酷壳 – CoolShell.cn

享受编程和技术所带来的快乐 – <http://coolshell.cn>

- [首页](#)
- [推荐文章](#)
- [本站插件](#)
- [留言小本](#)
- [关于酷壳](#)
- [关于陈皓](#)
-

[首页](#) > [C/C++语言](#), [杂项资源](#), [编程语言](#) > Leetcode 编程训练

Leetcode 编程训练

2014年10月23日 [陈皓](#) [发表评论](#) [阅读评论](#) 89,594 人阅读



Leetcode这个网站上的题都是一些经典的公司用来面试应聘者的面试题，很多人通过刷这些题来应聘一些喜欢面试算法的公司，比如：Google、微软、Facebook、Amazon之类的这些公司，基本上是应试教育的功利主义。

我做这些题目的不是为了要去应聘这些公司，而是为了锻炼一下自己的算法和编程能力。因为我开始工作的时候基本没有这样的训练算法和编程的网站，除了大学里的“算法和数据结构”里的好些最基础最基础的知识，基本上没有什么训练。所以，当我看到有人在做这些题的时候，我也蠢蠢欲动地想去刷一下。

于是，我花了3-4个月的业余时间，我把[Leetcode的154道题](#)全部做完了。（这也是最近我没有太多的时间来写博客的原因，你可以看到我之前[做的那个活动](#)中有几个算法题来自于Leetcode）有人说我时间太多了，这里声明一下，我基本上都是利用了晚上10点以后的时间来做这些题的。

LeetCode的题大致分成两类：

1) 基础算法的知识。这些题里面有大量的算法题，解这些题都是有套路的，不是用递归（深度优先DFS，广度优先BFS），就是要用动态规划（Dynamic Programming），或是拆半查找（Binary Search），或是回溯（Back tracing），或是分治法（Divide and Conquer），还有大量的对树，数组、链表、字符串和hash表的操作。通过做这些题能让你对这些最基础的算法的思路有非常扎实的了解和训练。对我而言，Dynamic Programming 是我的短板，尤其是一些比较复杂的问题，在推导递推公式上总是有思维的缺陷（数学是我的硬伤），通过做了这些题后，我能感到我在DP的思路上有了很大的收获。

2) 编程题。比如：atoi, strstr, add two num, 括号匹配，字符串乘法，通配符匹配，文件路径简化，Text Justification, 反转单词等等，这些题的Edge Case, Corner Case有很多。这些题需要你想清楚了再干，只要你稍有疏忽，就会有几个case让你痛不欲生，而且一不小心就会让你的代码会写得又臭又长，无法阅读。通过做这些题，可以非常好的训练你对各种情况的考虑，以及你对程序代码组织的掌控（其实就是其中的状态变量）。还记得我在《[函数式编程](#)》中说的，程序中的状态是你程序变得复杂难维护的直接原因。

我觉得每个程序员都应该花时间和精力做这些题，因为你会从这些题中得到很大的收益。做完这些题后你一定会明白下面几个道理：

1) 想清楚了再干。这个观点我以前就在《[多些时间可以少些代码](#)》说过。如果你拿到题就上去直接写代码的话，你一定会被各种case打回来了。然后呢，你一着急，你就会进入那种我在《[开发团队的效率](#)》中说的那种毫无效率case by case的开发模式，而你也进入了“平庸模式”。于是你就会出现下图那样的情况。



Case-by-Case Development

2) 编程是脑力劳动，急不得。这个事情在这做这些题的时候你就会发现，要么是脑子转不过来了，要么就是明明就差一点了，但程序怎么都调不对。如果你越着急的话，你就会发现你会离目标越远，而花的时间也会更多。另外，你会发现这些题基本上都是50行代码内就可以搞定的，但是为了这50行以内的代码，你要花好多时间和精力。coding 50行代码在我们的日常工作中分分钟就完成，而Leetcode里的50行代码却没那么简单，也许，用这个你就可以区别什么是码农，什么是程序员了。

3) 加班要不得。因为我总是在晚上10点以后做题，所以，基本上都是在加班状态中工作。这种状态过上两三天，你就会发现，整个大脑已经不转了，而且不但不转，还会犯很多低级错误，很多事情都想不清楚，一个晚上都在和程序的状态控制做搏斗，代码写得越来越乱，越来越没条理。于是这种时候，我都会休息几天，不做题了，然后再做题的时候，就觉得非常地清楚。可见加班 是编程最致命的敌人！

我把我的C++代码放到了Github上，大家也帮我review一下，看看有没有可以改善的。

<https://github.com/haol/leetcode>

好了，不多说了，我希望大家有时间都去练练LeetCode，无论是找工作还是对你的编程能力会有非常大的提高。

(全文完)



关注CoolShell微信公众账号可以在手机端搜索文章

(转载本站文章请注明作者和出处 酷壳 - CoolShell.cn，请勿用于任何商业用途)

==== 访问 [酷壳404页面](#) 寻找遗失儿童。 ====



本广告收入已由广告主捐给Wikipedia

40

分类: [C/C++语言](#), [杂项资源](#), [编程语言](#) 标签: [Algorithm](#), [C++](#), [Leetcode](#), [Programmer](#), [Programming](#), [程序员](#), [算法](#), [面试](#)
★★★★★ (74 人打了分, 平均分: 4.78)

相关文章

- 2011年11月29日 [一些有意思的算法代码](#)
- 2012年08月22日 [为什么我反对纯算法面试题](#)
- 2011年04月06日 [一些有意思的文章和资源](#)
- 2011年02月28日 [打印质数的各种算法](#)
- 2012年10月14日 [Bret Victor – Learnable Programming](#)
- 2011年03月18日 [可视化的排序过程](#)
- 2010年07月12日 [一些重要的算法](#)
- 2011年07月12日 [给程序员新手的一些建议](#)

[评论 \(28\)](#) [Trackbacks \(4\)](#) [发表评论](#) [Trackback](#)



1.

lance_xian

2014年11月23日 10:31 | [#1](#)

[回复](#) | [引用](#)

很不错，每一次来总能看到新的东西!!! 赞楼主



2.

chenggl

2014年11月26日 09:02 | [#2](#)

[回复](#) | [引用](#)

<https://github.com/haol/leetcode/blob/master/src/LRUCache/LRUCache.cpp>

有错误,修正为:

```
1 void Unlink2(Node* n){
2     Node* before = n->prev;
3     Node* after = n->next;
4
5     if (before){
6         before->next = after;
7     }else{
8         pHead = after;
9     }
10
11    if (after){
12        after->prev = before;
13    }else{
14        pTail = before;
15    }
16
17    size--;
18 }
```



3.

xMAs

2014年12月2日 11:59 | [#3](#)

[回复](#) | [引用](#)

你好，感谢您分享的leetcode的解题思路，我在看您的解题的时候，注意到您在题目《Intersection of Two Linked Lists》中对一个私有函数int getListLength()进行了内联处理，即：

```
1 inline int getListLength(ListNode *head){
2     int len=0;
3     while(head!=NULL){
4         head = head->next;
5         len++;
6     }
7     return len;
8 }
```

自己学习c++的时候，看到很多资料都说如果函数内有循环体的不建议使用内联的方式处理，因为即使有inline关键字，编译器也会把该函数作为非内联函数处理。请问博主解决这个问题时是怎么思考的？还是笔误？



陈皓

2014年12月3日15:40 | [#4](#)

[回复](#) | [引用](#)

哪些资料说的？



xMAs

2014年12月4日11:12 | [#5](#)

[回复](#) | [引用](#)

@陈皓

我也是在学习的时候在一些程序员的博客（并非权威）里面看到，对这个也比较疑惑，所以请教下博主你有什么验证它是否起作用的方法。。。谢谢。。。顺带请教，博主的mac是什么配置的，用什么环境搞开发？？我刚买了个8G的air，想用它来搞python、c/c++的开发，有什么可以指点的？？？



frestyle

2014年12月14日07:32 | [#6](#)

[回复](#) | [引用](#)

谢谢博主分享高质量代码，学习了！



cloudfeng

2014年12月14日17:06 | [#7](#)

[回复](#) | [引用](#)

虽然还没有把题目做完，但是个人非常同意楼主的观点：（1）想清楚了再干，这一点深有同感，特别是做个种边界条件的题目的时候。（2）编程确实不能着急，这个观点很多国外的经典书籍都提到了，比如TPOP。但是人么，都会有心理焦虑症滴。其实个人特别疑惑，虽然做了几道题目，但是却忘记的差不多。嗨，这个记忆呀。最后特别感谢楼主的心得体会。



frestyle

2014年12月15日06:13 | [#8](#)

[回复](#) | [引用](#)

你好，permutations这个题可以写个简洁的dfs:

```
1 void dfs(int pos, vector &num, vector<vector> &result){
2     if(pos == num.size()){
3         result.push_back(num);
4     } else {
5         for(int i=pos; i<num.size(); i++){
6             swap(num[i], num[pos]);
7             dfs(pos+1, num, result);
8             swap(num[i], num[pos]);
9         }
10    }
11 }
```



devin

2014年12月22日16:53 | [#9](#)

[回复](#) | [引用](#)

博主要是能加上单元测试那就更加完美了，赞一个。



9.

jianggest

2014年12月25日 17:59 | [#10](#)[回复](#) | [引用](#)

我提个题外的话：搞不懂这个评论是怎么显示的？为何第二页的评论时间比第一页的部分早呢？



10.

lxw

2014年12月30日 16:12 | [#11](#)[回复](#) | [引用](#)

耗子哥推荐！必须看！



11.

小飞猪

2015年1月15日 17:46 | [#12](#)[回复](#) | [引用](#)

源代码如下：

```
$(document).ready(function() {
var params = {};
params.quality = "high";
params.bgcolor = "#ffffff";
params.allowscriptaccess = "sameDomain";
params.allowfullscreen = "true";
params.flashvars = 'a=ggssso9..rhrgtj-bnl.bnllnm.hl`fdr.`c.eku.i`u`rr,l-rv&b=qgsso9..rhrgtj-bnl.u0.ok`xEhkd-ir&c=ugsso9..rhrgtj-bnl.uhcdn.dwh&d=11&e=101&f=1&g=3501&h=0&i=2&j=egsso9..rhrgtj-bnl.uhcdn.qdbnq&k=1&l=10031519&m=baidu&x=XUTKNUBDDS2BC2TTZCBUQ576NJMPXL5Q3JGU5RAJERSFLIKTURLC';
var attributes = {};
attributes.id = "ssonlineplayer-2";
attributes.name = "ssonlineplayer-2";
attributes.align = "middle";
swfobject.embedSWF(
'/videomgr/portal/player/ssonlineplayer-2.swf',
"flashContent",
"640",
"520",
"10.1.0",
"/videomgr/portal/player/expressInstall.swf", "", params, attributes, function(e){if(!e.success){$("#flashContentError").show();}}
));
```



12.

小飞猪

2015年1月15日 17:49 | [#13](#)[回复](#) | [引用](#)

[小飞猪](#)：源代码如下：

```
$(document).ready(function() {var params = {};params.quality = "high";params.bgcolor = "#ffffff";params.allowscriptaccess = "sameDomain";params.allowfullscreen = "true";params.flashvars = 'a=ggssso9..rhrgtj-bnl.bnllnm.hl`fdr.`c.eku.i`u`rr,l-rv&b=qgsso9..rhrgtj-bnl.u0.ok`xEhkd-ir&c=ugsso9..rhrgtj-bnl.uhcdn.dwh&d=11&e=101&f=1&g=3501&h=0&i=2&j=egsso9..rhrgtj-bnl.uhcdn.qdbnq&k=1&l=10031519&m=baidu&x=XUTKNUBDDS2BC2TTZCBUQ576NJMPXL5Q3JGU5RAJERSFLIKTURLC';
attributes = {};attributes.id = "ssonlineplayer-2";attributes.name = "ssonlineplayer-2";attributes.align = "middle";swfobject.embedSWF('/videomgr/portal/player/ssonlineplayer-2.swf',"flashContent","640","520","10.1.0","/videomgr/portal/player/expressInstall.swf","", params, attributes, function(e){if(!e.success){$("#flashContentError").show();}}});
```

```
a=ggssso9..rhrgtj-bnl.bnllnm.hl`fdr.`c.eku.i`u`rr,l-rv&b=qgsso9..rhrgtj-bnl.u0.ok`xEhkd-ir&c=ugsso9..rhrgtj-bnl.uhcdn.dwh&d=11&e=101&f=1&g=3501&h=0&i=2&j=egsso9..rhrgtj-
```

bnl.uhcdn.qdbnq&k=1&l=10031519&m=baidu&x=XUTKNUBDDS2BC2TTZCBUQ576NJMPXL5Q3JGU5RAJERSFLIKTURLC
是如何编码和解码的。谁知道告诉一声哦，非常感谢你的回复和关注。



13. [wee](#)
2015年2月22日 10:10 | [#14](#)
[回复](#) | [引用](#)

不能着急，恩。



14. [java爱好者](#)
2015年2月22日 18:10 | [#15](#)
[回复](#) | [引用](#)

我自己也喜欢刷leetcode，其实就是为了复习一下算法，因为在开发的生涯中经常会忘记了这些算法内容。相对而言，自己因为经常开发，而且也经常做java开发的笔记，也做一些自己的博客站点来记录开发的心得，比如www.javaflush.com或者www.javafreer.com所以其实最缺的还是平时学习。看到陈浩这么努力积极刷题，看来我也要加油了。时间都是挤出来的。



15. [Hola](#)
2015年3月16日 15:58 | [#16](#)
[回复](#) | [引用](#)

挺好个网站,用来复习算法.



16. [latecomer](#)
2015年3月27日 13:05 | [#17](#)
[回复](#) | [引用](#)

[@陈皓](#)

前辈您好，这个我也看到过很多资料上面写了在内联函数中不能出现while等。在中有提到：“要牢记在心的一条是，inline指令就象register，它只是对编译器的一种提示，而不是命令。也就是说，只要编译器愿意，它就可以随意地忽略掉你的指令，事实上编译器常常会这么做。例如，大多数编译器拒绝内联”复杂”的函数（例如，包含循环和递归的函数）”；在<>中：“假如函数太复杂，编译器将不能够执行内联。”另外还有很多网上的博客讨论中都有人提出”在内联函数内不允许用循环语句和开关语句。”随便举出的网址有<http://www.cnblogs.com/singa/archive/2008/09/24/1297821.html>;
<http://www.yesky.com/329/1925329.shtml>;
还有内联函数百科中也有这样的说法。前辈能简单讲解析吗。谢谢了！



17. [魔王](#)
2015年4月8日 16:25 | [#18](#)
[回复](#) | [引用](#)

10点之后还要做这么费脑子的工作，牛人就是厉害啊



18. [damo](#)
2015年5月27日 15:17 | [#19](#)
[回复](#) | [引用](#)

[@xMAs](#)

[@latecomer](#)

在下面这个问题中：

<http://stackoverflow.com/questions/13190326/loops-and-inline-functions>

题主也见到了包含循环的函数不能内联这种说法，但是被众多答主否定了。不过这说明这个江湖流言确实存在，它来自哪呢？其中一位答主提到“I think the OP is reading some very old C++ books or using old C++ compilers. I also have the impression that some 90's C++ books claim that loops cannot be put into inline functions. – xuhdev Mar 2 '14 at 0:25”

结论就是:

1. 在201X年的今天, 不存在这种说法
2. 该流言可能源自90年代的C++书籍
3. 一个流言真的可以流传很多年, 而且被人很认真地记忆, 传颂.....



19.

宝贝小猪

2015年5月28日00:10 | [#20](#)

[回复](#) | [引用](#)

有个问题特别好奇, 特别期待lz的回复。这一百多道题目, 肯定有自己想不出来的, 这个时候我想知道您是怎么做的呢? 我找工作的时候为了做题而做题, 就是会看别人的答案, 然后背下来转换为自己的。您呢?



20.

Curried

2015年6月17日01:29 | [#21](#)

[回复](#) | [引用](#)

You solution for problem No.1 the TwoSum, has a bug.
Beacuse the order of array was unknown, so we can't use binary search
your code
+ while (low target ? high- : low++;
}
}
Thanks.



21.

Future

2015年7月20日20:58 | [#22](#)

[回复](#) | [引用](#)

确实厉害啊, 从04年写到现在。



22.

[JustYY.com](#) 小赖子的英国生活和资讯。

2015年7月22日18:11 | [#23](#)

[回复](#) | [引用](#)

NB, MARK一下。

DP也是我的短板。



23.

point

2015年11月11日19:53 | [#24](#)

[回复](#) | [引用](#)

最近也在刷算法题, 分享答案不如分享思路,
最近找到的这个网站, 答案和思路一并给上, 个人感觉还不错
<http://tianmaying.com/tutorials/tag/Leetcode>



24.

tosca

2015年12月20日15:38 | [#25](#)

[回复](#) | [引用](#)

[@xMAs](#)

你记反了吧 应该是编译遇见那种比较短小的函数 会作为inline来处理



25.

levy

2016年1月24日16:19 | [#26](#)

[回复](#) | [引用](#)

试试评论功能，最近想用nodejs写一个自己的博客，以作学习之用。

26.



levy

2016年1月24日16:25 | [#27](#)

[回复](#) | [引用](#)

这次我是在safari评论的，上次是在chrome里。

27.



Shawn

2016年9月20日20:19 | [#28](#)

[回复](#) | [引用](#)

这个评论怎么弄成倒序的呀~

评论分页

[« 上一页](#) [1](#) [2](#) 12052

1. 2015年9月2日11:02 | [#1](#)

[Leetcode | Hooya!](#)

2. 2015年9月2日21:28 | [#2](#)

[Cuckoo Filter: 设计与实现 | 星达红](#)

3. 2015年12月13日17:56 | [#3](#)

[让我们来谈谈分工 | 星达红](#)

4. 2016年10月6日23:53 | [#4](#)

[大家一起来 Leetcode – 天如](#)

昵称 (必填)

电子邮箱 (我们会为您保密) (必填)

网址

[订阅评论](#)

[vfork 挂掉的一个问题](#) [State Threads 回调终结者](#)

[订阅](#)

[Twitter](#)

本站公告



访问 [酷壳404页面](#) 寻找遗失儿童！



酷壳建议大家多使用RSS访问阅读（本站已经是全文输出，推荐使用cloud.feedly.com 或 digg.com）。有相关事宜欢迎电邮：[haoel\(at\)hotmail.com](mailto:haoel(at)hotmail.com)。最后，感谢大家对酷壳的支持和体谅！

最新文章

- [什么是工程师文化？](#)
- [关于高可用的系统](#)
- [这多年来我一直在钻研的技术](#)
- [缓存更新的套路](#)
- [为什么我不在微信公众号上写文章](#)
- [性能测试应该怎么做？](#)
- [让我们来谈谈分工](#)
- [Cuckoo Filter: 设计与实现](#)
- [Docker基础技术: DeviceMapper](#)
- [Docker基础技术: AUFS](#)
- [Docker基础技术: Linux CGroup](#)
- [Docker基础技术: Linux Namespace \(上\)](#)
- [Docker基础技术: Linux Namespace \(下\)](#)
- [关于移动端的钓鱼式攻击](#)
- [Linus: 为何对象引用计数必须是原子的](#)
- [DHH 谈混合移动应用开发](#)
- [HTML6 展望](#)
- [Google Inbox如何跨平台重用代码？](#)
- [vfork 挂掉的一个问题](#)
- [Leetcode 编程训练](#)
- [State Threads 回调终结者](#)
- [bash代码注入的安全漏洞](#)
- [互联网之子 - Aaron Swartz](#)
- [谜题的答案和活动的心得体会](#)
- [【活动】解谜题送礼物](#)
- [开发团队的效率](#)
- [TCP 的那些事儿 \(下\)](#)
- [TCP 的那些事儿 \(上\)](#)
- [「我只是认真」聊聊工匠精神](#)


- [面向GC的Java编程](#)

全站热门

- [程序员技术练级攻略](#)
- [简明 Vim 练级攻略](#)
- [做个环保主义的程序员](#)
- [如何学好C语言](#)
- [AWK 简明教程](#)
- [TCP 的那些事儿（上）](#)
- [应该知道的Linux技巧](#)
- [“21天教你学会C++”](#)
- [6个变态的C语言Hello World程序](#)
- [由12306.cn谈谈网站性能技术](#)
- [编程能力与编程年龄](#)
- [sed 简明教程](#)
- [28个Unix/Linux的命令行神器](#)
- [“作环保的程序员，从不用百度开始”](#)
- [我是怎么招聘程序员的](#)
- [性能调优攻略](#)
- [二维码的生成细节和原理](#)
- [Web开发中需要了解的东西](#)
- [Lua简明教程](#)
- [MySQL性能优化的最佳20+条经验](#)
- [C++ 程序员自信心曲线图](#)
- [Android将允许纯C/C++开发应用](#)
- [如何学好C++语言](#)
- [无插件Vim编程技巧](#)
- [如何写出无法维护的代码](#)
- [20本最好的Linux免费书籍](#)
- [Windows编程革命简史](#)
- [编程真难啊](#)
- [加班与效率](#)
- [深入理解C语言](#)

新浪微博

微博



左耳朵耗子 北京 朝阳区

加关注

标签

[agile](#) [AJAX](#) [Algorithm](#) [Android](#) [Bash](#) [C++](#) [Coding](#) [CSS](#) [Database](#) [Design](#) [design pattern](#) [ebook](#) [Flash](#) [Game](#) [Go](#) [Google](#) [HTML](#) [IE](#) [Java](#) [Javascript](#) [jQuery](#)
[Linux](#) [MySQL](#) [OOP](#) [password](#) [Performance](#) [PHP](#) [Programmer](#) [Programming](#) [programming language](#) [Puzzle](#) [Python](#) [Ruby](#) [SQL](#) [TDD](#) [UI](#) [Unix](#) [vim](#) [Web](#) [Windows](#)
[XML](#) [安全](#) [程序员](#) [算法](#) [面试](#)

分类目录

- [.NET编程](#) (3)
- [Ajax开发](#) (9)
- [C/C++语言](#) (71)
- [Erlang](#) (1)
- [Java语言](#) (32)
- [PHP脚本](#) (11)
- [Python](#) (23)
- [Ruby](#) (5)
- [Unix/Linux](#) (75)
- [Web开发](#) (103)

- [Windows](#) (12)
- [业界新闻](#) (26)
- [企业应用](#) (2)
- [技术新闻](#) (33)
- [技术管理](#) (15)
- [技术读物](#) (117)
- [操作系统](#) (49)
- [数据库](#) (11)
- [杂项资源](#) (270)
- [流程方法](#) (48)
- [程序设计](#) (87)
- [系统架构](#) (9)
- [编程工具](#) (65)
- [编程语言](#) (174)
- [网络安全](#) (27)
- [职场生涯](#) (34)
- [趣味问题](#) (19)
- [轶事趣闻](#) (147)

归档

- [2016年九月](#) (1)
- [2016年八月](#) (2)
- [2016年七月](#) (3)
- [2015年十二月](#) (1)
- [2015年九月](#) (1)
- [2015年八月](#) (2)
- [2015年四月](#) (4)
- [2014年十二月](#) (3)
- [2014年十一月](#) (2)
- [2014年十月](#) (2)
- [2014年九月](#) (2)
- [2014年八月](#) (2)
- [2014年六月](#) (1)
- [2014年五月](#) (4)
- [2014年四月](#) (4)
- [2014年三月](#) (5)
- [2014年二月](#) (3)
- [2014年一月](#) (2)
- [2013年十二月](#) (3)
- [2013年十一月](#) (1)
- [2013年十月](#) (6)
- [2013年八月](#) (1)
- [2013年七月](#) (8)
- [2013年六月](#) (2)
- [2013年五月](#) (3)
- [2013年四月](#) (3)
- [2013年三月](#) (3)
- [2013年二月](#) (5)
- [2013年一月](#) (1)
- [2012年十二月](#) (4)
- [2012年十一月](#) (4)
- [2012年十月](#) (3)
- [2012年九月](#) (4)
- [2012年八月](#) (8)
- [2012年七月](#) (4)
- [2012年六月](#) (7)
- [2012年五月](#) (6)
- [2012年四月](#) (6)
- [2012年三月](#) (6)
- [2012年二月](#) (3)
- [2012年一月](#) (6)
- [2011年十二月](#) (5)

- [2011年十一月](#) (9)
- [2011年十月](#) (6)
- [2011年九月](#) (5)
- [2011年八月](#) (14)
- [2011年七月](#) (6)
- [2011年六月](#) (12)
- [2011年五月](#) (5)
- [2011年四月](#) (18)
- [2011年三月](#) (16)
- [2011年二月](#) (16)
- [2011年一月](#) (18)
- [2010年十二月](#) (11)
- [2010年十一月](#) (11)
- [2010年十月](#) (19)
- [2010年九月](#) (15)
- [2010年八月](#) (10)
- [2010年七月](#) (20)
- [2010年六月](#) (9)
- [2010年五月](#) (13)
- [2010年四月](#) (12)
- [2010年三月](#) (11)
- [2010年二月](#) (7)
- [2010年一月](#) (9)
- [2009年十二月](#) (22)
- [2009年十一月](#) (27)
- [2009年十月](#) (17)
- [2009年九月](#) (14)
- [2009年八月](#) (21)
- [2009年七月](#) (18)
- [2009年六月](#) (19)
- [2009年五月](#) (27)
- [2009年四月](#) (53)
- [2009年三月](#) (43)
- [2008年十月](#) (1)
- [2007年十二月](#) (1)
- [2006年十一月](#) (1)
- [2004年六月](#) (1)

最新评论

- [dongdong](#): 真想在工程师文化的公司工作
- [fanhexin](#): 国内是码农文化
- [Meloyi](#): 说说我是如何看到这篇文章的：百度搜索“python”，点击搜索结果：“你是如何自学 Python 的？”，查看文章评论时看到这个“酷壳 - ...
- [asdf](#): 这是个非常复杂的投入和产出问题，每个公司可能模型都不一样。m s现在老板技术出生，可能他觉得QA太低效了，所以砍掉。
- [tangyikejun](#): @asd 确实是这个感觉，作者很有眼界
- [行之予之](#): 我看这个文章已经是好几年前了，受益匪浅，今天正式回复下，感谢 前辈笔耕多年。
- [Laohaoren](#): 不是都说 不该发明 null这个玩意儿么
- [zjh](#): @薛非 是的 你很聪明
- [zjh](#): 大家快点页面的头右侧有一个high一下!!!!!!!!!!!! ~~~~~!!!!!!!!!!!! ...
- [zjh](#): 我是最新评论
- [zjh](#): @Eson 其实文章题目是在说反话 其实是如何避免写出无法维护的代码
- [zjh](#): @TonyQ 诡异的是这个浏览器很好的支持 那个浏览器就变了支持的模式
- [limiteed](#): 除非整个团队都是非常专业的人，不然我才不管收么工程师文化呢， 全部996。
- [品酒的狐狸](#): 赞博主，从你这里学到好多！
- [天下彩](#): 深有同感

友情链接

- [陈皓的博客](#)
- [并发编程](#)

- [四火的唠叨](#)
- [HelloGcc Working Group](#)
- [吕毅的Blog](#)
- [Todd Wei的Blog](#)
- [C++爱好者博客](#)
- [HTML5研究小组](#)
- [朱文昊Albert Zhu](#)
- [C瓜哥的博客](#)
- [开源吧](#)
- [ACMer](#)
- [陈鹏个人博客](#)
- [OneCoder](#)
- [More Than Vimer](#)
- [运维派](#)
- [书巢](#)

功能

- [注册](#)
- [登录](#)
- [文章RSS](#)
- [评论RSS](#)
- [WordPress.org](#)



[回到顶部](#) [WordPress](#)

版权所有 © 2004-2016 酷壳 - CoolShell.cn

主题由 [NeoEase](#) 提供, 通过 [XHTML 1.1](#) 和 [CSS 3](#) 验证.

