

本文 GitHub <https://github.com/rd2coding/Road2Coding> 已经收录，里面有我整理的**6大编程方向(岗位)的自学路线+知识点大梳理、面试考点、我的简历、几本硬核pdf笔记**，以及**我的程序员人生**。

今天应大家要求，再来继续推荐几个我收藏的，关于C语言和C++领域的适合初学者学习和练手的开源项目，供大家参考。

学完编程语言感觉还只是玩具，其实也挺常见，主要是因为没有足够的实战和练手，花点时间好好研读这些开源项目大有裨益。大家可以认真**吸收**这些项目并真正**转化**为自己的技能点，这样以后不管是**复试、写简历**亦或是**求职找工作**，也能更加从容一点！

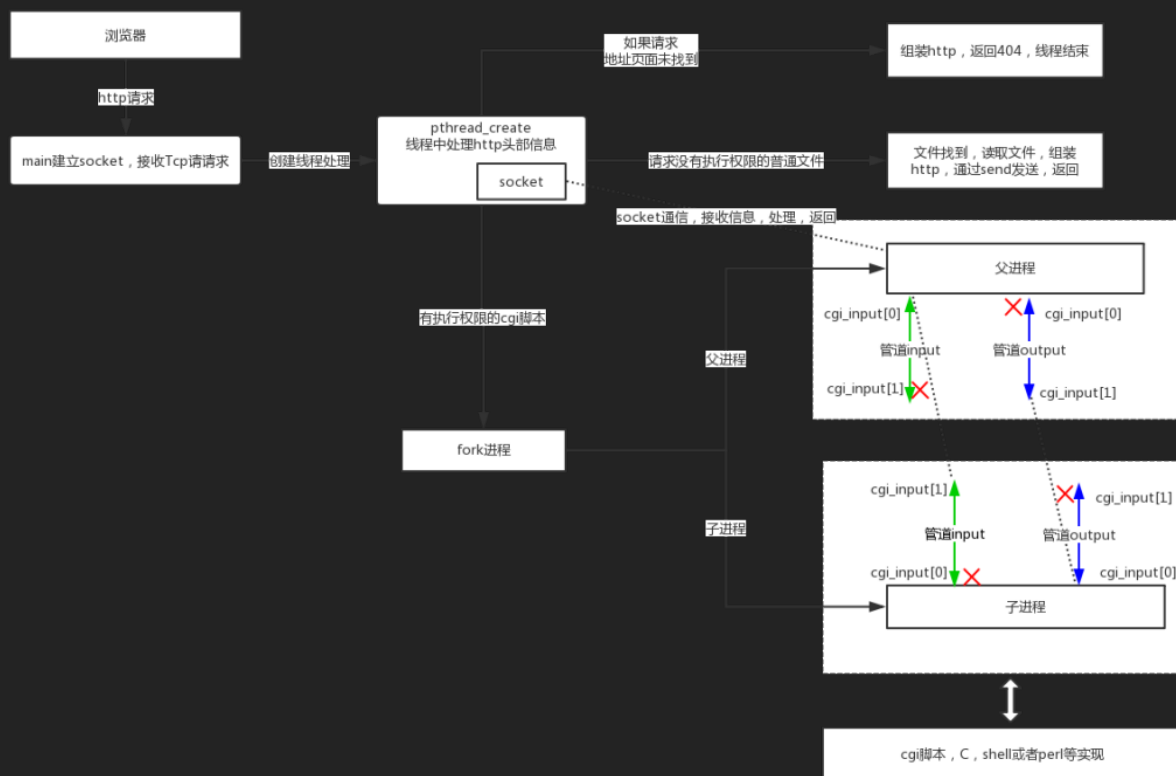


必须要说的是，不像Python，Java等开源项目一般都具备美观的可视化界面、网页、甚至手机App；而C语言和C++写的项目一般非常朴素，大多运行于命令行那个简单的黑乎乎界面，甚至完全运行于后台。

TINYHTTPD

项目名称： Tinyhttpd

项目简介： Tinyhttpd 是J. David Blackstone在1999年写的一个不到 500 行的超轻量型 Http Server，用来学习非常不错，可以帮助我们真正理解服务器程序的本质。建议源码阅读顺序为：main ->startup ->accept_request ->execute_cgi，通晓主要工作流程后再仔细把每个函数的源码看一看。这500行代码吃透了，C语言的功底就会大幅提升。



【注：图片来源于：www.cnblogs.com/nengm1988/p/7816618.html】

项目源码：<https://github.com/EZLippi/Tinyhttpd>

TMUX

项目名称：tmux

项目简介：tmux一个炫酷的终端复用软件，它提供了一个非常易于使用的命令行界面，可横向和纵向分割窗口，窗格可以自由移动和调整大小，而且还可以通过交互式菜单来选择窗口、会话及客户端。类似的终端复用器还有 GNU Screen。Tmux 与它功能相似，但是更易用，也更强大。大名鼎鼎的阮一峰老师还写过tmux的使用教程，大家也可以看一看

```

File Edit View Search Terminal Help
COMPUPDATE ON;
end transaction;
DEBUG: Upload for hour 22 complete
DEBUG: Upload for hour 21 complete
DEBUG: Upload for hour 20 complete
DEBUG: Upload for hour 18 complete
DEBUG: Upload for hour 19 complete
DEBUG: Upload for hour 17 complete
DEBUG: Upload for hour 03 complete
DEBUG: Upload for hour 14 complete
DEBUG: Upload for hour 13 complete
DEBUG: Upload for hour 12 complete
DEBUG: Upload for hour 11 complete
DEBUG: Upload for hour 10 complete
DEBUG: Upload for hour 09 complete
DEBUG: Upload for hour 04 complete
DEBUG: Upload for hour 08 complete
DEBUG: Upload for hour 05 complete
DEBUG: Upload for hour 07 complete

DEBUG: Upload for hour 01 complete
DEBUG: Upload for hour 23 complete
DEBUG: Upload for hour 06 complete
DEBUG: Upload for hour 21 complete
DEBUG: Upload for hour 20 complete
DEBUG: Upload for hour 19 complete
DEBUG: Upload for hour 02 complete
DEBUG: Upload for hour 18 complete
DEBUG: Upload for hour 17 complete
DEBUG: Upload for hour 12 complete
DEBUG: Upload for hour 15 complete
DEBUG: Upload for hour 11 complete
DEBUG: Upload for hour 10 complete
DEBUG: Upload for hour 14 complete
DEBUG: Upload for hour 09 complete
DEBUG: Upload for hour 06 complete
DEBUG: Upload for hour 08 complete
DEBUG: Upload for hour 05 complete
DEBUG: Upload for hour 03 complete

from 's3://redshift.
ens_06.csv'
credentials 'aws_access_key_id=
s_key=
ACCEPTINVCHARS
format as CSV
IGNOREBLANKLINES DELIMITER ','
MAXERROR 0
DATEFORMAT 'auto'
TIMEFORMAT 'auto'
TRUNCATECOLUMNS
COMPUPDATE ON;
end transaction;
DEBUG: uploading file to S3: /mnt/tmp/load_table_redshift.0h_Mfa/opens_0
0.csv => s3://redshift.
ns_06.csv
DEBUG: uploading file to S3: /mnt/tmp/load_table_redshift.ChhBRL/opens_0
7.csv => s3://redshift.
ns_07.csv

1 100.0% 9 100.0% 17 100.0% 25 100.0%
2 100.0% 10 98.7% 18 97.4% 26 98.1%
3 100.0% 11 100.0% 19 100.0% 27 100.0%
4 100.0% 12 100.0% 20 100.0% 28 100.0%
5 100.0% 13 100.0% 21 100.0% 29 100.0%
6 100.0% 14 100.0% 22 100.0% 30 100.0%
7 100.0% 15 100.0% 23 100.0% 31 100.0%
8 100.0% 16 100.0% 24 100.0% 32 100.0%
Mem[|||||]56518/60140MB Tasks: 328, 17 thr; 209 running
Swp[|||||]0/6MB Load average: 137.52 44.35 16.99
Uptime: 00:51:19

PID USER PRI NI VIRT RES SHR S CPU% MEM% TIME+ Command
7854 dfarrell 20 0 116M 757M 5744 R 99.0 1.3 1:14.00 perl /var
7859 dfarrell 20 0 1102M 748M 5744 R 99.0 1.2 1:14.62 perl /var
7861 dfarrell 20 0 1093M 739M 5744 R 99.0 1.2 1:14.98 perl /var
7858 dfarrell 20 0 1102M 749M 5744 R 99.0 1.2 1:12.87 perl /var
7856 dfarrell 20 0 1121M 768M 5744 R 94.0 1.3 1:14.68 perl /var

[0] <s 1:dfarrell@host-2:~/Projects/perltricks 2:dfarrell@host-2:~/Projects/ 3:dfarrell@host-2:~/Projects/* 4:dfarrell@batch2: 5: 17:39 11-Feb-16

```

项目源码: <https://github.com/tmux/tmux>

MUSIKCUBE

项目名称: musikcube

项目简介： musikcube是一个使用C ++编写的跨平台，运行于终端上的音乐播放器。musikcube可以在Windows，macos和linux上轻松编译和运行。它也可以在带有raspbian的树莓派上很好地运行，并且可以设置为流音频服务器。炫酷得一腿。



项目源码: <https://github.com/clangen/musikcube>

MYTINYSTL

项目名称: MyTinySTL

项目简介: 很多人表示学完C++不知道用来干什么, 我觉得学完C++的第一个练手的好机会那就是自己试着实现一个小型的STL库。MyTinySTL的作者它就用 C++11 重新复写了一个小型 STL (容器库+算法库)。代码结构清晰规范、包含中文文档与注释, 并且自带一个简单的测试框架, 非常适合新手学习与参考!

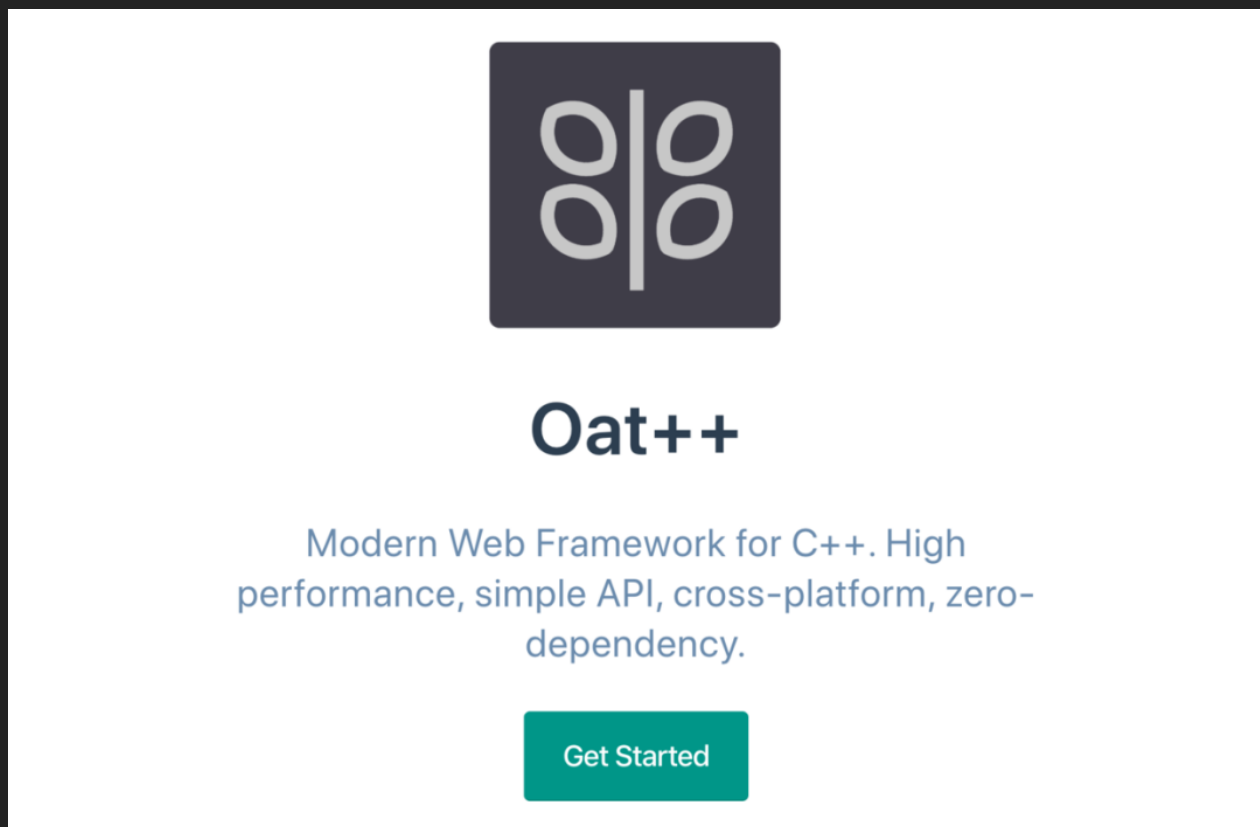
目录

- 配置器
 - allocator
 - allocate
 - deallocate
 - constructor
 - construct
 - destroy
- 迭代器
 - iterator
- 仿函数
 - functional
 - hash_functional
- 容器
 - vector
 - list
 - deque
 - map
 - set
 - unordered_map
 - unordered_set
 - basic_string
- 算法
 - 基本算法
 - 数值算法
 - set 算法
 - heap 算法
 - 其他算法
- 配接器
 - container adapters
 - stack
 - queue
 - priority_queue
 - iterator adapters
 - reverse_iterator

项目源码: <https://github.com/Alinshans/MyTinySTL>

项目名称: oatpp

项目简介: 我们知道Java领域的Web框架非常繁荣, 最知名的当属Spring全家桶, 而C语言和C++阵营则几乎没有。那oatpp则是一个轻量、跨平台、高性能、完全零依赖, 用纯 C++ 实现的 Web 框架, 实在是难得, 小伙伴们可以学习学习。



项目源码: <https://github.com/oatpp/oatpp>

MUDUO

项目名称: muduo

项目简介: muduo是一个基于Boost库实现的现代C++高并发网络库, 由陈硕大神编写。它是一个高质量的事件驱动型的网络库, 其核心代码不超过4500行, 使用 `non-blocking IO(IO multiplexing)` + `one loop per thread` 模型, 适合开发 Linux 下的多线程服务端应用程序, 通过阅读源码还可学习到 C++ 语言、Linux 网络编程等后端知识。

README

Muduo is a multithreaded C++ network library based on the reactor pattern.

<http://github.com/chenshuo/muduo>

Copyright (c) 2010, Shuo Chen. All rights reserved.

Use of this source code is governed by a BSD-style license that can be found in the License file.

Requires:

Linux kernel version \geq 2.6.28.
GCC \geq 4.7 or Clang \geq 3.5
Boost (for boost::any only.)

Tested on:

Debian 7 and above
Ubuntu 14.04 and above
CentOS 7 and above

To build, run:

./build.sh

See <https://github.com/chenshuo/muduo-tutorial> for how to use muduo in your project.



项目源码: <https://github.com/chenshuo/muduo>

CPPNET

项目名称: CppNet

项目简介: CppNet一个封装在 TCP 协议上的 Proactor 模式 multi-thread 网络库。包含 OS 接口调用、回调处理、定时器、缓存管理等，这里有从操作系统到应用层的所有网络细节，便于初学者学习和实践。

- 简单: 只导出了最少量的接口，其声明都类似系统 socket API。对客户端而言，只新增了一个 buffer 类型
- 快速: 采用性能最优的 epoll 和 IOCP 做事件驱动。每个连接都独享一个内存池，从内存池中申请的内存都由智能指针管理
- 清晰: 结构上分为事件驱动，会话管理，接口三层，通过回调向上通知。模块之间职责分工明确，最大的类不超过 500 行代码

项目源码: <https://github.com/caozhiyi/CppNet>

联系我，直接微信扫码，给我私信即可 ↓



本文 GitHub <https://github.com/rd2coding/Road2Coding> 已经收录，里面有我整理的**6大编程方向**的自学路线+知识点大梳理、我的简历、面试考点、几本硬核pdf笔记，以及我的程序员人生，欢迎star。

每天进步一点点！Peace！