

Warmup / Linux & Git

酒井科协暑培 2023

abmfy x icera x jkjkmxmx

清华大学计算机系学生科协

2023 年 7 月 9 日

① 课程大纲

课程概述

推荐路线

② 如何解决问题

③ Linux & Git

课程形式

- 所有课程均在 哔哩哔哩直播 进行
 - 请关注哔哩哔哩「清华大学计算机系科协」
- 除 Unity 以外的课程统一在 19:30 开始
- Unity 课程在 20:00 开始
- 可以通过哔哩哔哩观看课程回放

课程主页

- 今年上线了全新的课程主页 summer23.net9.org
- 在首页可以看到近期课程预告
- 为了方便快速索引，对不同 track 课程作了分类
- 在每个课程的页面可以看到课前准备、课程讲义及课后作业

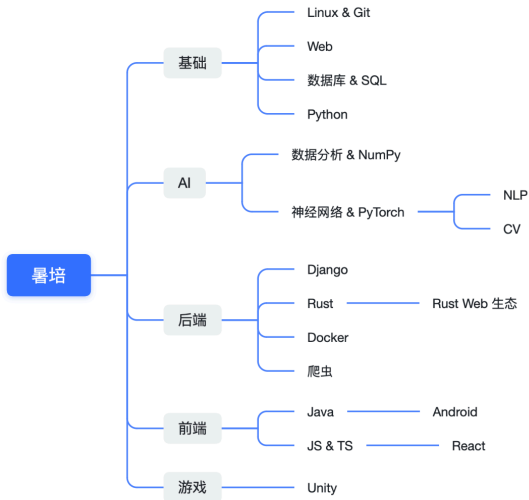
课程大纲

分为 5 个 track:

- 基础
- AI
- 后端
- 前端
- 游戏

原则：适度学习，拒绝过载

课程大纲



① 课程大纲

课程概述

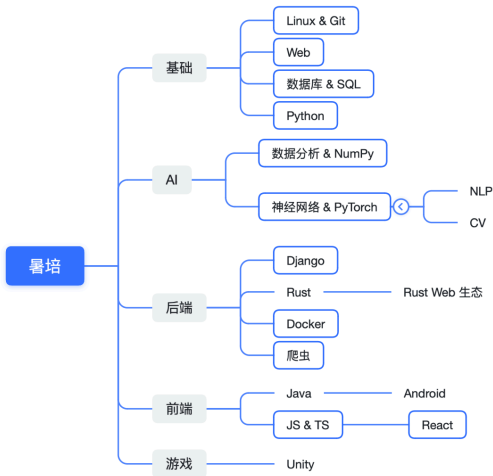
推荐路线

② 如何解决问题

③ Linux & Git

推荐路线：贵系漫游

面向计算机系同学，覆盖后续课程需要用到的绝大多数技术栈

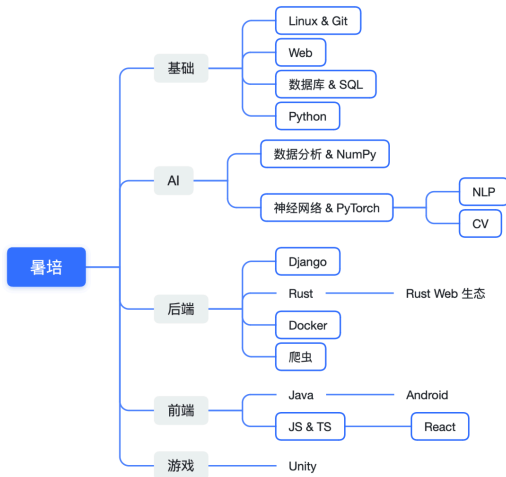


推荐路线：贵系漫游 ++

- 如果你对某一方面的技术特别感兴趣：
- 在贵系漫游路线的基础上，针对较为感兴趣的方向额外增加 2 课时课程
 - Unity 方向为 4 课时

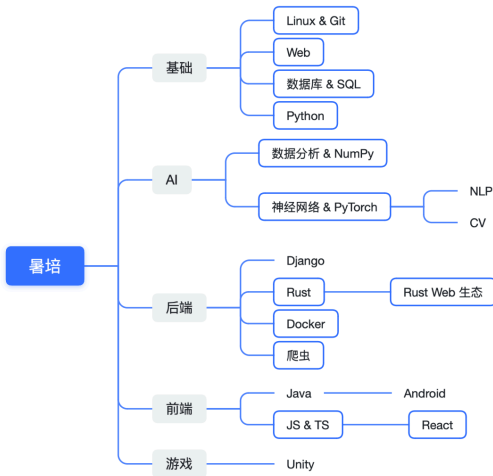
推荐路线：贵系漫游 ++ - 炼丹师

深入学习 AI 的两大具体方向



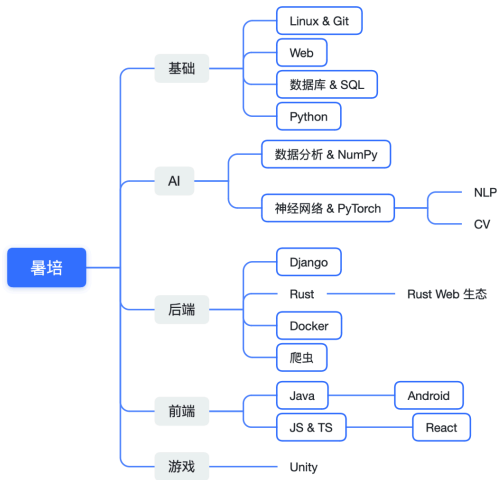
推荐路线：贵系漫游 ++ - 不如 Rust

你在学一门很新的编程语言……



推荐路线：贵系漫游 ++ - 多端协同

Write Once, Debug Everywhere



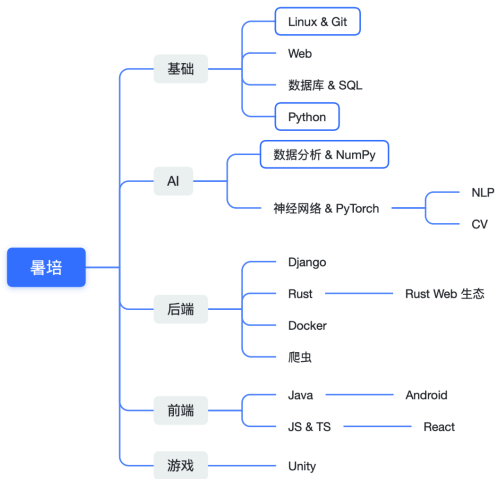
推荐路线：贵系漫游 ++ - 游戏大师

Made with Unity



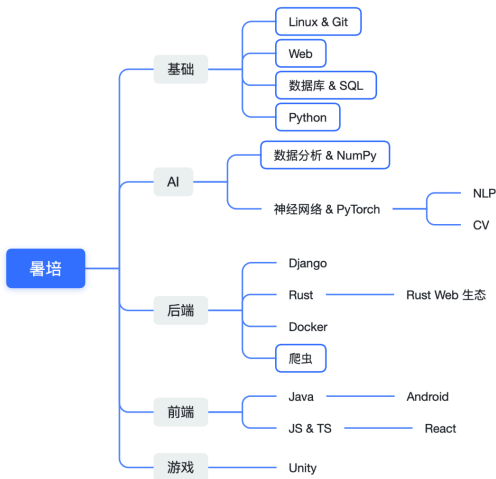
推荐路线：人生苦短，我用 Python

面向非计算机系同学：从 Python 快速入门编程



推荐路线：来点「蜘蛛侠」

面向非计算机系同学：利用爬虫快速获取信息



① 课程大纲

② 如何解决问题

搜索引擎

常用网站

官方文档

如何提问

③ Linux & Git

说明

摘取改编自 <https://docs.net9.org/basic/meet-problems/>

引入

科协水群里闲聊时，有人“暴言”：“在贵系有技术问题时，摇人比查文档更方便。”遇到问题时多和其他人交流、避免闷头苦干当然是支持的，但这并不意味着问题前不需要付出努力（例如在搜索引擎上查找）、问问题时对方有义务替你解答.....

——摘自“酒井科协”公众号 2022年3月25日文章

① 课程大纲

② 如何解决问题

搜索引擎

常用网站

官方文档

如何提问

③ Linux & Git

选用搜索引擎

由于一些商业性因素的限制，百度常常会返回一些与搜索结果不是最强相关的内容。我们在这里推荐两款搜索引擎：

- 如果你因为中国大陆的限制无法访问互联网上的一部分内容，我们推荐你使用“**Bing** 国际版”
- 如果你可以访问到世界互联网上的内容，我们则推荐你使用“**Google**”。

互联网上技术相关的资料信息的语言分布并不与全地球上掌握该门语言的人数成正比，推荐大家尝试使用英文进行搜索

使用搜索引擎

- 删去与你本地配置有关的信息，例如文件路径、项目名称。这些信息不仅会对搜索结果造成干扰，还可能会泄漏你的个人隐私。
- 使用传统搜索引擎的时候，使用关键词而非完整的语句。例如说，不要用“为什么我的 Mingw C++ 无法编译并且报错 code 987”，改用“Mingw 无法编译 code 987”

使用搜索引擎

有一些较为进阶的搜索方法：

- 指定网站搜索 `site:web.site`
- 精确匹配 "Exact Match"
- 必须包含 + 与必须不包含 -
- 任意匹配 *
- 指定时间范围，例如只获取过去 1 年内的搜索结果，这样可以保证搜索结果的时效性

新选择：ChatGPT

ChatGPT 可以快速帮我们解决许多的问题，其有以下优势：

- 往往可以给出较为靠谱的解答
- 无需输入关键词，直接输入问题即可
- 在强大的模型支撑下，即使使用中文询问也能得到特别不错的结果

如果你有能力访问 ChatGPT，建议你尝试使用其协助你进行搜索；如果你在清华校内，也可以使用 ChatGLM。

① 课程大纲

② 如何解决问题

搜索引擎

常用网站

官方文档

如何提问

③ Linux & Git

选择常用网站

要想搜索不同的内容，我们一般会选择不同的网站

这里我们针对一些常见的问题，列出一些常用的网站

Debug 类

Runtime Error / Compilation Error: StackOverflow

- 程序运行时遇到了我不了解的报错，即使查看了函数调用栈也不理解最终的报错信息；
- 程序编译不过，其中用了我不理解的编译特性...

StackOverflow 系列网站是这些问题的好帮手，可以替代 CSDN 作为你的搜索首选。

事实上，CSDN 的部分内容是直接从 StackOverflow 上搬运而来，而这些搬运而来的内容在国内网站上又相互爬取、相互展示...

出处类

What is the original paper of XXX...: scholar.google.com

我们想知道某篇学术论文的出处，或者是其引用格式。这时 scholar.google.com 就可以起到很好的作用。如果你知道这篇论文的全名，你可以直接进行搜索，然后便捷地获取引用格式与论文原文的 PDF 版本。

代码类

Whether or not someone has implemented this...: GitHub

想找过往的轮子？好办，在 GitHub 中进行搜索，你可以看到规模庞大的开源代码... 但是，在使用时请务必遵续其对应的 LICENSE。

可以尝试使用 <https://cs.github.com/>，能够更加精确、全面的搜索公开仓库的代码。

教程类

此外，在学习对应的内容时，我们推荐到对应的官方网站、教程、文档或 Wiki 上进行学习。

在学习一些公开课程可能涉及到的知识时（如 CS231n），我们推荐先搜索课程主页，进而在 YouTube 上搜索对应的讲解视频，然后在 B 站与知乎等国内平台上进行检索。

① 课程大纲

② 如何解决问题

搜索引擎

常用网站

官方文档

如何提问

③ Linux & Git

查阅官方文档

大部分你使用的工具或者库都会有自己的官网网站 / GitHub Repo / Wiki。了解一个工具或者函数库最全面、也是最便利的方法，可能是读它对应部分的说明文档。

查阅官方文档

大部分你使用的工具或者库都会有自己的官网网站 / GitHub Repo / Wiki。了解一个工具或者函数库最全面、也是最便利的方法，可能是读它对应部分的说明文档。

例如说，我们想了解怎么在 PyTorch 中实现叉乘：

- 或许我们可以在百度中搜索 `pytorch` 叉乘，然后点开一个 CSDN 教程
- 我们更可以 Google 搜索 `pytorch cross product`，然后打开 `pytorch.org` 的官方文档进行阅读来的全面而透彻。

查阅官方教程

- 不同工具的文档会有不同的特点，部分工具的文档对于初学者而言晦涩难懂。如果遇到这种情况，可以尝试寻找是否有针对初学者的实例代码，涵盖了期望的内容。
- 另外，在尝试引入一个新的包的时候，文档的质量也应该是着重考虑因素之一。

查阅官方教程

- 当你在使用一个造好的轮子时，发现自己完全无从下手，连如何编译运行也不知道
- 你找到了解决问题的一个回答，却发现里面充满自己不认识的名词，需要学习的知识呈现二叉树一般...

查阅官方教程

- 当你在使用一个造好的轮子时，发现自己完全无从下手，连如何编译运行也不知道
- 你找到了解决问题的一个回答，却发现里面充满自己不认识的名词，需要学习的知识呈现二叉树一般...

你可能需要换一种思路，将从现象出发改为从底层出发。这时候，官方文档中的教程部分是你的好帮手。

查阅官方教程

- 当你在使用一个造好的轮子时，发现自己完全无从下手，连如何编译运行也不知道
- 你找到了解决问题的一个回答，却发现里面充满自己不认识的名词，需要学习的知识呈现二叉树一般...

你可能需要换一种思路，将从现象出发改为从底层出发。这时候，官方文档中的教程部分是你的好帮手。

例如，我想快速上手一个名叫"Carla"的模拟仿真工具，最合适的方式，应该是搜索"carla doc"，然后打开 [官方文档](<https://carla.readthedocs.io/en/latest/>) 阅读其中"Getting Started"部分，这样能对这个软件有最基本的了解。之后再根据自己的需要阅读其他部分的内容。

查阅官方教程

这看上去不能立刻解决你的问题，但长远来看，这将为你节省更多的时间。

通过例子来学习也是十分重要的方法。

这些示例文档也可能没有中文版本。Getting Started, Tutorial, Learn 是常见的关键词。

① 课程大纲

② 如何解决问题

搜索引擎

常用网站

官方文档

如何提问

③ Linux & Git

有效提问

在你尝试了如上内容无果之后，自己闷头苦干或许已经不能解决你的问题了。是时候进行提问了——

打开微信列表，翻出一个人的账号，然后敲下一句“在吗”，等待回复——不，这并不是正确的提问方式。

有效提问

在你尝试了如上内容无果之后，自己闷头苦干或许已经不能解决你的问题了。是时候进行提问了——

打开微信列表，翻出一个人的账号，然后敲下一句“在吗”，等待回复——不，这并不是正确的提问方式。

要想有效地进行提问，我们强烈推荐你阅读一遍参考资料。当然，我们在这里列出几点重点：

提问之前

- 请确保你通过搜索引擎检索不到你想问的问题
- 请确保你查阅了官方文档，其中并没有提及可以直接解决你的问题的答案
- 请确保你自己做了充分的预实验——比如"这段程序的运行结果是什么？"

提问内容



- 请提供充分的信息给回答者——比如，请不要发一个"这段代码有 Bug [捂脸]"的语句给对方，然后附上一个不完全的代码段截图。
- 不要以文本方式发送在微信聊天框发送大段代码。请使用在线的 Pastebin 或者发送文件，注意不要将生成的二进制文件也一同发出。
- 请提供充分的说明：哪段代码？出了什么问题，报错信息是什么？这段代码的目标功能是什么？
- 如果有可能的话，请给出足够多的注释（不管是代码还是你的思考内容）
- 请说出你目前做了哪些努力

提问之后

即使提问，也不代表问题的主动权放在了对方手里。问题毕竟是你自己的问题，对方没有义务帮助你必须解决掉这个问题，所以继续自行寻找答案也许是需要

① 课程大纲

② 如何解决问题

③ Linux & Git

课程简介

容器发放

- ① 课程大纲
- ② 如何解决问题
- ③ Linux & Git
 - 课程简介
 - 容器发放

课程简介

- 通过文档的引导，自行学习 Linux & Git 的相关知识
- 在我们提供的 Docker 容器中动手实践，探索 Linux 世界
- Capture-The-Flag: 你需要根据指引找到隐藏的一系列 flag 来完成作业

这不是 Fly, *****! 吗?

你先别急，但是：

- 自主学习能力是在计算机系生存不可或缺的，同时我们提供了充足的引导让你明白该到何处寻求帮助；
- 这部分内容是后续所有学习和开发的基础，如果不上手实践而只是听讲很难真正掌握；
- 经过去年的验证，这种学习模式效果很好；
- 如果在完成作业的过程中碰上任何困难，可以随时在课程主页评论区 / 课程群求助；
- 最重要的：CTF 真的很好玩！

① 课程大纲

② 如何解决问题

③ Linux & Git

课程简介

容器发放

容器发放

如果你是清华大学的学生：

- 我们为所有 7 月 8 日 23:59 前填写报名问卷的同学在科协的 Zeus 服务器上提供了完成作业的容器环境；
- 请关注清华邮箱通知，查收我们发放的容器。

否则：

- 很抱歉由于服务器资源所限，我们无法在服务器上为你提供容器；
- 我们在课程文档中提供了如何在本地运行作业容器的指引。

Thanks!

Questions?