

# 问题解决指南

---

## 仔细阅读本节

当你遇到问题的时候，本节的内容会很有帮助，如果你在实验过程中遇到了难以解决的问题，请回来看看。

## 面对问题的态度

---

当你在进行本实验或是现实生活中其他项目时，你可能会遇到很多问题：

- 代码出现报错
- 不知道某个功能的代码如何实现
- 不知道某样工具怎么使用
- 其他各种各样的问题.....

那么，面对一系列问题的时候，你会怎么做？首先，我们希望你们不要害怕问题，遇到问题是正常的，或者说，没遇到问题才是不正常的。然后，我们希望你们可以端正态度，不要抱着应付的心态，也不要抱着“抱紧大佬的大腿就能解决问题”的心态，希望你们可以先独立地思考问题，独立地多做尝试，等到实在是解决不了，实在需要帮助的时候，请继续往下看。

## 如何获得帮助

---

### 面向搜索引擎编程

一般情况下，你遇到的问题大部分人都已经遇到过，所以通过搜索引擎，你可以很快找到问题的解决方案。但是，不要使用百度、搜狗、360等搜索引擎，而且尽量不要使用中文关键词，因为你遇到的大部分问题可能都是一串英文提示。

一般来说，在CS方向推荐使用Google搜索，再不济使用bing搜索。至于如何使用Google，不属于本实验讨论的范围，但这是一个计算机专业学生的必备技能。实在需要帮助的，可以寻找身边同学。

如果你确实不知道怎么访问，我们在[这里](#)提供了一个镜像。

### 面向问答网站编程

当你遇到问题时，去逛逛互联网的问答网站提问也是一个很好的选择，毕竟很多大佬在网站上给出了自己的问题解决方案。

我猜你们很多人过去都是使用CSDN作为问答网站的（我猜的没错吧），但CSDN上的问答实在是良莠不齐，我们强烈建议使用[StackOverflow](#)，里面有非常多的技术问题和学术性问题，全世界杰出的工程师和研究者都针对一些问题提出了自己建设性的想法，相信我，你会在其中收获良多的。

## 面向AI编程

此外，最近ChatGPT大行其道，或许你们很多人以及感受过它的强大之处了。当然，这也是一个很好的提问渠道，是的，本实验不仅不反对大家使用ChatGPT，还非常鼓励大家使用ChatGPT，毕竟，我们也要跟上时代的潮流。ChatGPT确实会提高你们获取答案的效率，也许手册中一个难以寻找的点，它只需要几秒钟就告诉你们了，前提是你懂得如何提问。

## 面向人类编程

当然，你也可以寻找身边同学的帮助，可以寻找助教帮助。

这个时候就会有同学问了：

助教，我的内核panic了，怎么办？

你知道么，你这样助教也很panic啊。拜托，不要再这样问问题了好吗？如果你们不知道如何正确提问，强烈建议阅读[别向弱智一样提问](#)和[提问的智慧](#)这两篇文章。

最后想说的是，你们可以向别人寻求帮助，但我们并不希望你们直接抄别人的代码，这是违反学术诚信的行为，也会让你们失去很多本来应该获得的能力。

## 还有一些很有帮助的参考资料

### xv6

- [xv6 book](#)

### RISC-V

- The RISC-V Instruction Set Manual ([Volume I](#), [Volume II](#))
- [ABI for RISC-V](#)

### 工具类

- [Harley Hahn's Guide to Unix and Linux](#)
- [GDB User Manual](#)
- [GDB, The GNU Debugger, By Example](#)
- [Git User Manual](#)
- [Pro Git](#)（该书有中文版）
- [The Missing Semester of Your CS Education](#)（该教程有中文版）

## 教材

- The C programming language (second edition)
- [Operating Systems: Three Easy Pieces](#)
- [操作系统：原理与实现（银杏书）](#)