1. 碎碎念

为什么看过了很多编程的书，依然编不出一个程序？

为什么信心十足的开始学编程，看着满屏的代码，一脸懵逼后选择放弃？

因为，你没有来郭蜀黍的编程班。

以有趣的故事，将知识点穿针引线成一个任务。

秉承着最小难度原则，在不打击信心的情况下，力保每天学有所获。

实在不明白，怎么办？

还有一帮快乐编程的小伙伴帮你解答问题，毕竟教才是最好的学。

自己懂了，教会别人，收获了知识，收获了友谊，收获了快乐！

学完了一期之后，你能收获什么？

至少：

1. 掌握列表、字典、元组等基本的数据结构。（一定要认真学，写程序的时候就知道多有用了，在各种函数中来来回回用，不知道这，连错在哪都不知道额）
2. 两个编程的原则：单一职责原则和Don’t Repeat Yourself。（保证里能有条理的写出简洁程序）
3. debug技能。（这个要在实际中积累，蜀黍名言：反正电脑又不会爆炸！）
4. 命令行、Git、GitHub这些编程界看似高端的东东也统统轻松拿下来。
5. 知道了什么是面向对象的编程方法即类。

如果你乐于折腾，那更好了，python有各种各样的库，为你打开新世界的大门。

我会告诉你，我们班好多人玩乌龟都玩的好开心吗？

什么？这些还不够？

那就来编程班-section2吧！

1. 个人感受

首先学到了不少知识，相当于接触到了一个新的领域，也掌握了这个领域的一些思考方式，关于编程方面的就不展开了。

说一说关于学习的吧。三板斧：在练习中学习，通过教会同学来加深，通过复盘来总结反思。

其实，这个方法不止可以用于编程，也可以用于很多其他方面，比如学习英语或者其他的一项技能。

1. 课程建议

对于，基础班可以总结一下常见的问题汇总，这样可以避免重复才坑，反复回答同一个问题。

后续的班级，可以安排定点答疑，或者由小助手收集问题以后答疑。一般的问题可以在群里随口就答了，代表性的问题可能需要老师指点。

分组，并没有发挥出作用。加入了一个小组，都没人发言，到最后还被踢了，泪目。。。

最小难度原则没毛病，但是可以适当补充些以供学有余力的同学发挥下。

1. 反思

这次编程课做的不够好的地方是，好高骛远，在数据结构方面没有掌握扎实，导致后面大作业的时候需要反复的查找各种数据结构的特点和用法，做了很多重复性工作，当然通过这些工作我对知识的掌握更加扎实了，可这些如果在学习的时候多复习两边，其实原本就可以避免的。

同时，也深刻地体会到，只有用的时候才能激发出最大的潜力学东西。平时，把那些知识点掌握的再扎实，记得再熟悉，也不如用一次。