阅读计划（二）

学号：20201050331

姓名：黄珀芝

专业：计算机科学与技术

学院：信息学院

如何做到自由地表达

——读《编程珠玑》《黑客与画家》有感

未来属于科技，希望寄予科技，而科技基于人才。

怎么成为一个编程人才，是我在阅读完这两本书后思考的最为深入的问题。不难发现，好的编程类书籍教予读者的往往不是算法，而是深入性的批判思维与开阔思想。成为一个会编程语言的人，并不难，一个正常智力的人每天学习七八个小时，在三个月后一定精通了一门编程语言；成为一个能够解决公司的计算机问题的程序员，也不难，这是一份在技校学习两年的人就能做的工作。但是成为一个有创造性与自己自由灵魂的“创造者”，并不简单。

只要你掌握了一种编程语言基本规则（比如语法结构，数据类型，表达方式等），你就可以成为一名程序员，而剩下的问题就是你在通过程序语言来表达你的想法的时候，能否做到准确完美严谨和富有创意，这就是一般的程序员与顶尖的程序员之间的差别，至于背景，那就比较复杂，有自学成才也有科班出身，而根本一点就是看你是否对你掌握的语言有超人的天赋或者感悟。例如，会画画的有很多，但并不是所有的会画画的人都可以称为画家的。

在我看来，所谓的专业的程序员，根据我自身的了解就是能够将自己的想法通过程序语言的方式表达出来，从这一点来说跟文学家用文字表达自己想法，画家用画表达是一样的道理。

也就是说，与其他的编程者不同的是——“自由地表达”。通俗点的讲法就是：编程是如何培养人工智能时代最重要的技能之一：创造力。

那么，空说无益，如何做到自由地表达呢？

创造力是一种技能，在这两本书中我明白了可以通过这三个方面来培养：

1、实验者心态：我们不难理解，创造性思维始于质疑心态。它通过鼓励实践者进行实验，让人们探索他们的想法，质疑他们的假设，甚至是从所犯错误中学习宝贵的经验。发明大王爱迪生，在发明第一个工作灯泡之前，他测试了数千种材料和工艺。“我已成功发现了1000种不制造灯泡的方法，” 他曾这么调侃自己的发明经历。编程也是如此。一开始，我们跟随的是老师的步伐，从类似于“Hallo World”这种简单代码开始解决。慢慢的，我们能够独立解决难度递增的问题，我们可以做到从头到尾完成一个实验。通过不断地实验和练习，我们可以熟练掌握编程技术和假设技能，从而能解决日益复杂的问题并最终完全靠自己构建程序。

2、强化全脑思维：研究表明，大脑的每一面都控制着思维和信息处理的不同部分。左半球通常与逻辑、技术和分析思维相关联，而右半球则与想象力，艺术性和直觉性思维相关联。我们倾向于将创造力视为右脑功能，但最有创造力的思想家和问题解决者可以有效地同时调动左右脑。这种将“艺术与科学”结合在一起的想法就是乔布斯建立苹果公司的想法。

比如我们都熟知的STEAM：这个集科学，技术，工程，艺术，数学多学科融合的综合教育体系，它打破了学科界限，有效提升了孩子的综合素养，编程正是其中的学科之一。

3、成为创造者（而不仅仅是消费者）的天生渴望：很多时候，并不是程序员不想创造，而是缺乏自己能够创造的信心。中国式的打压式教育在某种程度上会促进学习者的竞争力，但是同时也会让我们看不清自己的实力。“我做不到”“你看别人大佬都做不出来，我怎么可能”“做不出来还浪费时间去做”等等想法并不罕见。如果想让自己坚持做一件事，就要给自己暗示源源不断的动力，自信心就是最好的动力。编程最大的一个特点就是能让我们在很短的时间内编写出可执行的程序，而且是不受任何禁锢，肆意挥洒想象力的那种。

想起了我第一次经过无数次调试与修改bug后运行出来的程序的那一刻，就如同高中时期解开一道困难的数学题那样兴奋的感觉，这是我第一次产生了浓厚的兴趣。它是如此神奇，美妙，令我不觉沦陷其中。这对我增加自信心会有很大帮助，有了自信心，自然有学下去的欲望。

我们或许神化了代码，代码不过也是一种语言，一种工具，我们可以用汉语和英语创作出富有想象力的文字和作品，自然可以用代码编写出充满创造力的程序和作品。

成为一个编程人才，要有自由的灵魂与源源不断的创造力。幸运的是，只要我们努力朝着目标前进，这些都是可以做到的。