程序设计实践报告

时间就像手中握着的细沙,慢慢的,从指缝中溜走,一不留神,程序设计小集训已接近尾声，用一个字形容这个小学期——累！每天8点起床，一声叹息，费脑的一天又要开始了，有时候为了能多完成一道题目，中午的休息时间也得缩水一大部分。从7月4号开始，已经经过了6天，这6天收获颇丰，改变很大，收获与反思并存。以下做一下总结。

实训期间，温习了学过的知识，为下学期学习数据结构做了些准备。对于没刷过题的我而言，许多问题都无从下手。对于复杂的问题，往往思考1个多小时，绞尽脑汁都不得其解。李杰老师的提示，给了我们很大的帮助，老师的解答如醍醐灌顶，而对我们，则甘之如饴。

每天的实训都有新的收获，我对C语言的理解更加深入。每天的rank榜，紧张而刺激。通过4个班的比较，才醒悟到，人外有人天外有天的道理。不高的排名，让我端正了心态，激励自己不断努力。就像最后一次的实训题目上所写的，学无止境。

对于几个经典的问题处理方法，我想做一下总结：

1. **二分法查找：**

二分法查找，也称为折半法，是一种在有序数组中查找特定元素的搜索算法。

二分法查找的思路如下：

1. 首先，从数组的中间元素开始搜索，如果该元素正好是目标元素，则搜索过程结束，否则执行下一步。
2. 如果目标元素大于/小于中间元素，则在数组大于/小于中间元素的那一半区域查找，然后重复步骤1的操作。
3. 如果某一步数组为空，则表示找不到目标元素。
4. **贪心算法**

贪心算法，是指，在对问题求解时，总是做出在当前看来是最好的选择。也就是说，不从整体最优上加以考虑，他所做出的是在某种意义上的局部最优解。贪心算法不是对所有问题都能得到整体最优解，关键是贪心策略的选择，选择的贪心策略必须具备无后效性，即某个状态以前的过程不会影响以后的状态，只与当前状态有关

贪心算法的实现：

1. 从问题的某个初始解出发。
2. 采用循环语句，当可以向求解目标前进一步时，就根据局部最优策略，得到一个部分解，缩小问题的范围或规模。
3. 将所有部分解综合起来，得到问题的最终解。
4. **筛法求素数**

当只需要求某个数是不是素数的时候，我们可以直接通过素数的定义来求，即如果可以被除1及素数本身的其他数整除，则这个数不是素数但是如果要求某个范围内的素数的个数的时候这个方法就不太合适了。虽然我们可以进行预处理，但是这种方法比较慢，一旦范围过大，预处理过程便会超时。因此，需要使用筛法求素数，这样可以在线性时间内求得范围内每个数是否为素数。

算法思想：去除要求范围内所有的合数，剩下的就是素数，而任何合数都可以表示为素数的乘积，因此，如果已知一个数为素数，则它的倍数都是合数。把2到n在内所有的数都列出来，然后从2开始，先划掉n内所有2的倍数，然后每次从下一个剩下的数开始，划掉其n内所有倍数。最有剩下的数就是素数。

**……**

小实训虽然结束了，但计算机学习的道路还很远。接下来的暑假，就要按照计划，继续提高自己的技术水平，期待一个“更高更妙”的自己。

**2019-7-9**