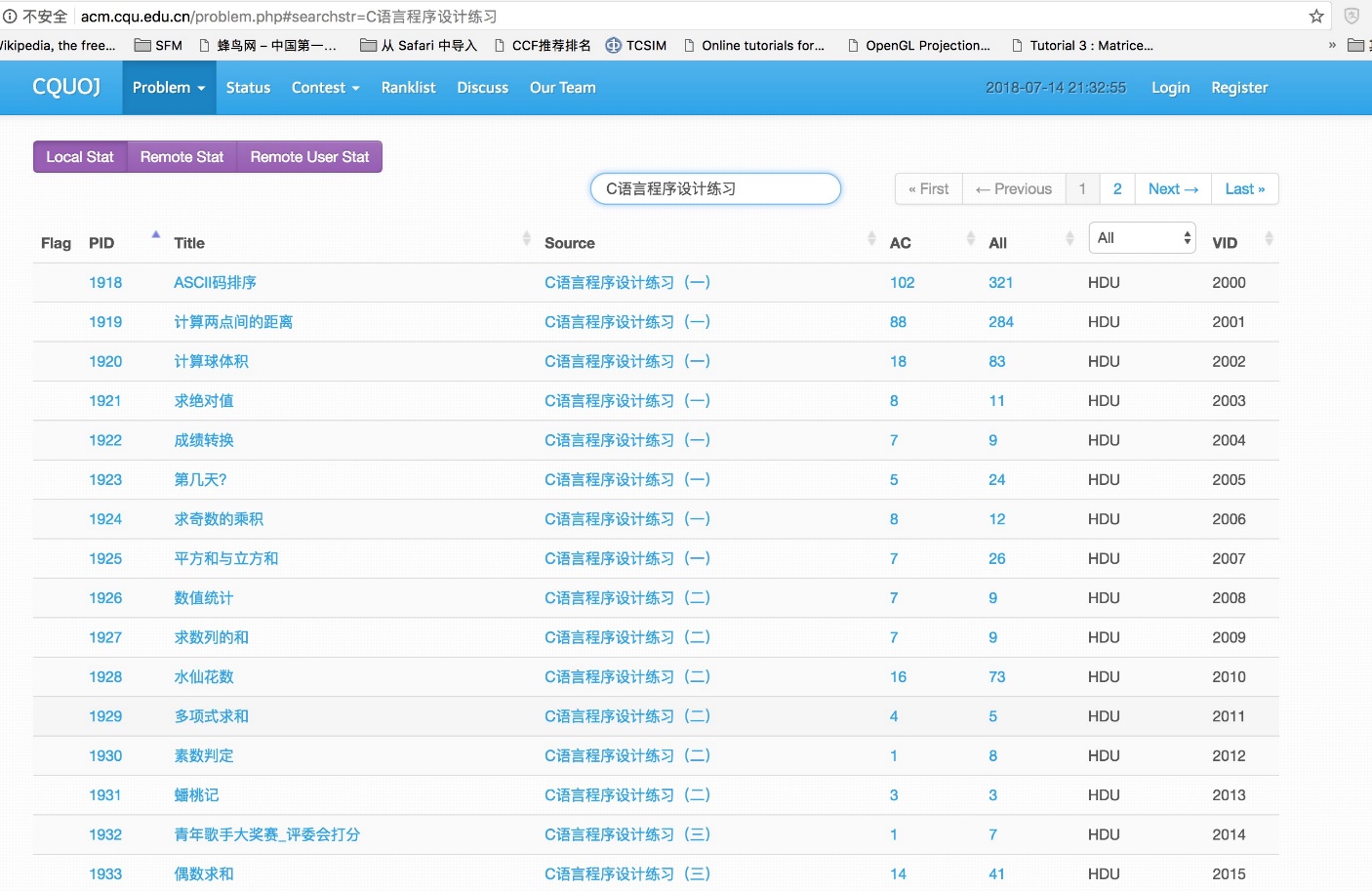
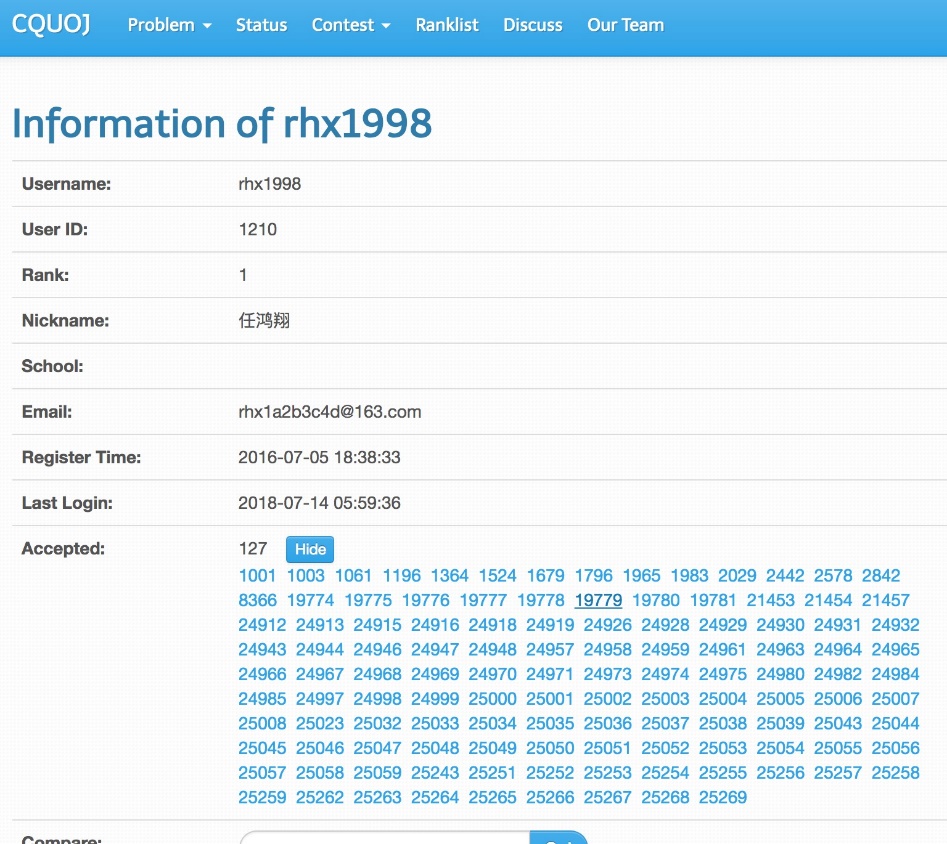
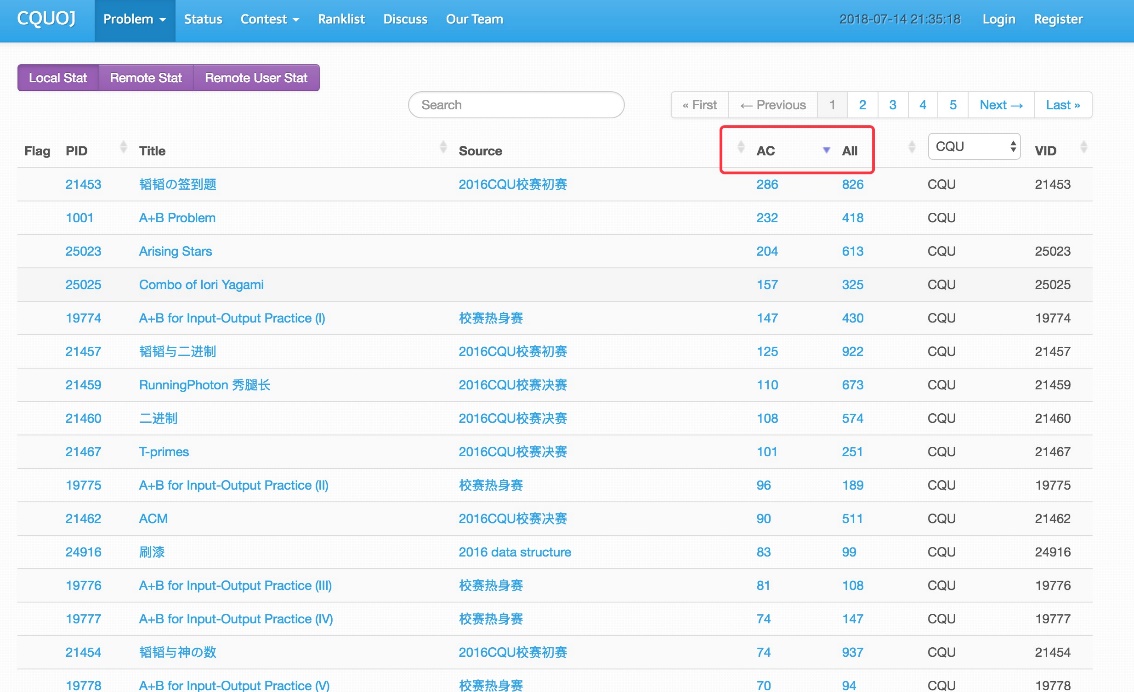
ACM学习教程

今天做了一套题，目的是为了对大家的基础进行摸底。有同学觉得基础比较差，担心很难通过这个课程。首先这种担心是没有必要的。这个培训的目的是提升各位同学的编程能力。成绩主要看大家在培训期间的表现、取得的进步。当然做题的成绩也是一部分。觉得基础比较差的同学，可以先做oj上面比较简单的题目，比如C语言和数据结构的习题。  
  
每位同学做了多少题，在系统中都是有记录的。只要你多做练习题，水平肯定能够提高，成绩也就不是问题。  


* 另一种方法是先做通过率高的题目，按照ac排序



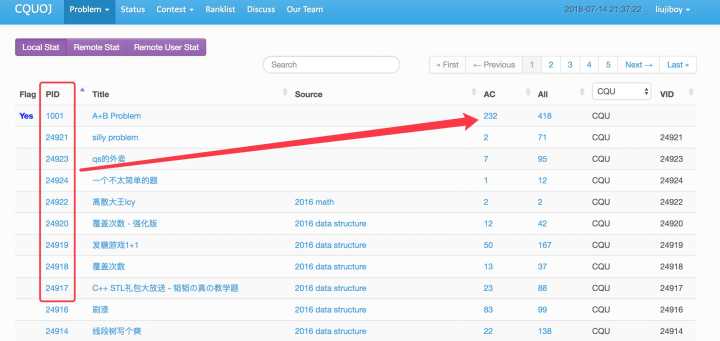
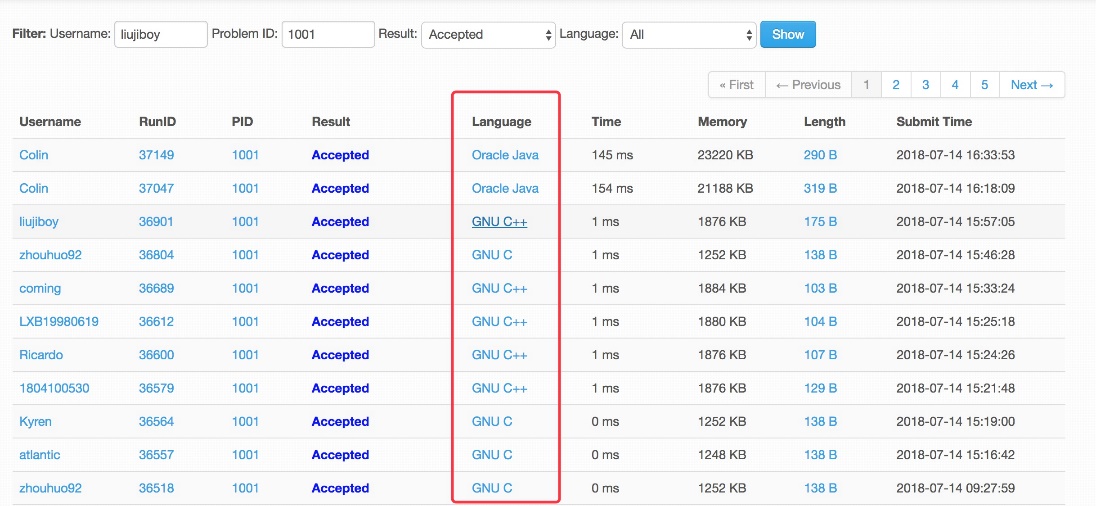
* AC越少的题目，难度越大
* 一道题如果实在搞不过，三种方法解决问题：

（1）看oj上别人accept的代码

（2）群里问

（3）上网搜

* oj上如何看别人的代码呢？

先找到题目的PID，然后找到AC  
  
  
  
点击AC，点击Language下面的链接  
  
  
  
  
把别人的代码copy一下，就可以过关，但这种作弊行为极不推荐。  
  
比赛过程中看不到其他人的代码，比赛之后是可以看到的。  
  
基础差的同学，先做AC高的题目（简单题目），水平上来之后，逐渐做难题。  
水平高的同学，直接去做难题，AC低的题目。

“刻意练习”是最近几年学习领域一个非常有用的方法。“刻意练习”告诉我们，要成为一个领域的专家，需要一定的练习时间（领域不同，必要时间不同）。但只是积累时间是不够的，还需要做到以下四点：

（1）练习的难度应该略高于自己当前的水平

（2）当觉得练习难度低于自己的水平时，应提升练习难度。

（3）绝对不要在远远高于自己水平的练习上浪费时间（比如5秒跑100米的练习）

（4）及时的反馈

实践证明，只要做到以上4点，再加上必要的练习时间，想不提高都难。

上面这个问题是因为编译器支持的语言标准不同

vs2017支持的语言标准比服务器上的低  
  
新的C语言已经支持变量作为数组大小

但处于性能的原因不建议这么做，对于有些题目可能会存在超时的风险。变量做数组大小说明数组的内存是动态分配，而不是静态分配。若数组较大，可能会占很多启动时间。  
  
常量就没有这个烦恼。处于性能的考虑，acm的题目总是倾向于采用静态的，而不是动态的结构。  
  
后面有些题目你们会遇到这个问题