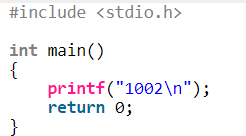
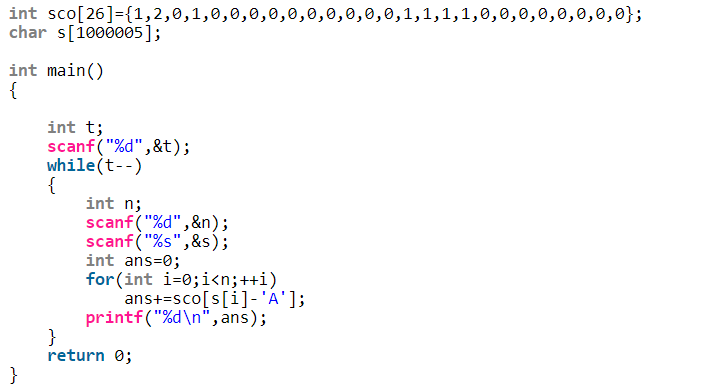
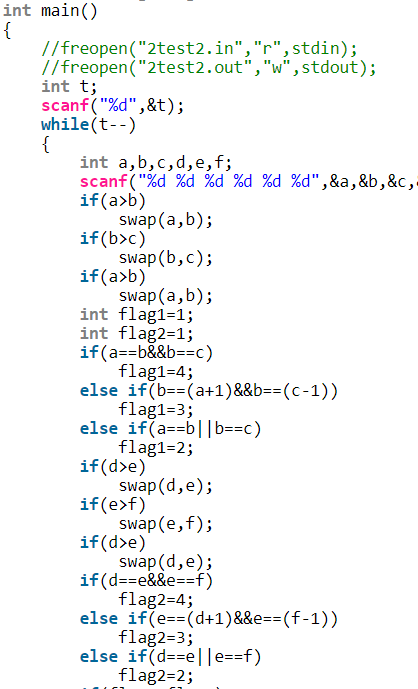
A：输出门牌号即可。。甚至B题上面有答案。

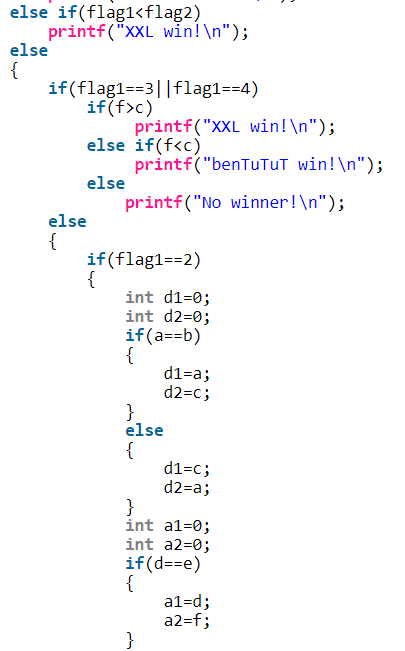


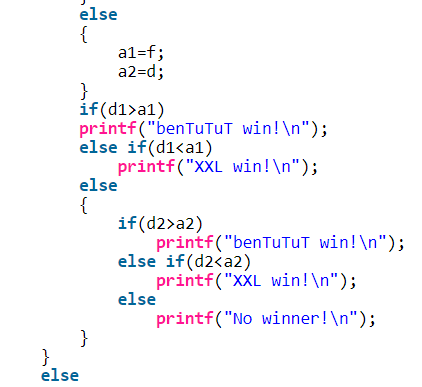
B：明确哪些字母有圈圈，记录一下每个字母对应的圈圈数，然后求和即可

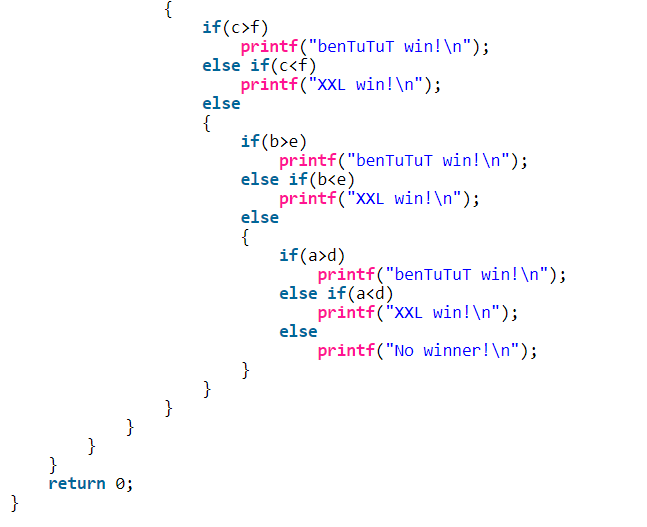


C：题目主要难度是细心，将每一种情况考虑完整即可

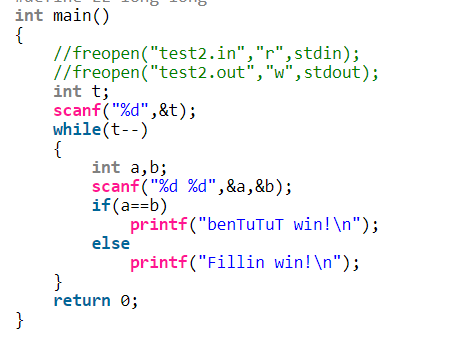








D：当两堆牌的牌数不同时，先手只需要让两堆牌牌数相等，然后无论对方怎么拿，只需要拿另一堆牌的相同数量，一定可以取得胜利，反之当两堆牌数量相同时，先手一定失败

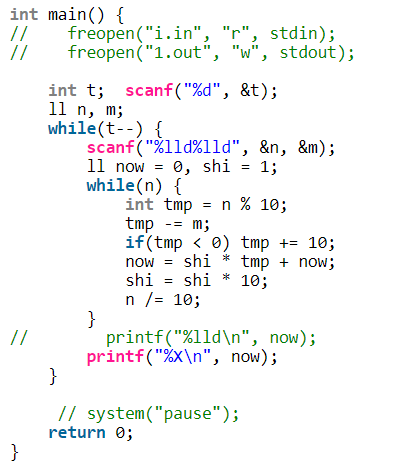


E：首先将U形的竖边和横边的长度算出来，然后根据长度将字符串存入二维数组，然后输出即可

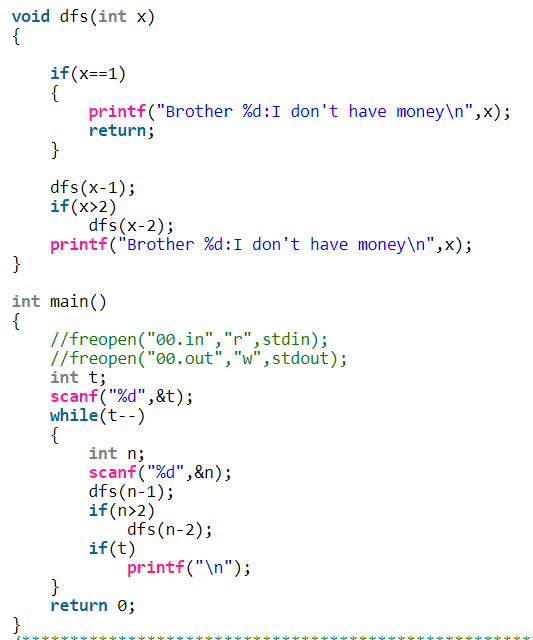
PS：一些同学可能是格式错误，原因是在最后一组样例后面多输出了一个空行，处理方法参照下面的程序，G题同



F：首先根据要求将数字转化成要求的数，之后用%X输出即可（大写X 16进制字母会用大写表示，小写x 16进制字母会用小写表示）

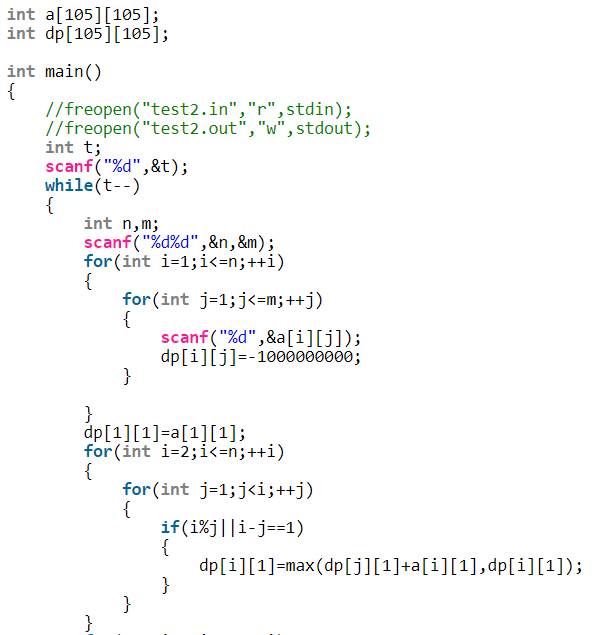


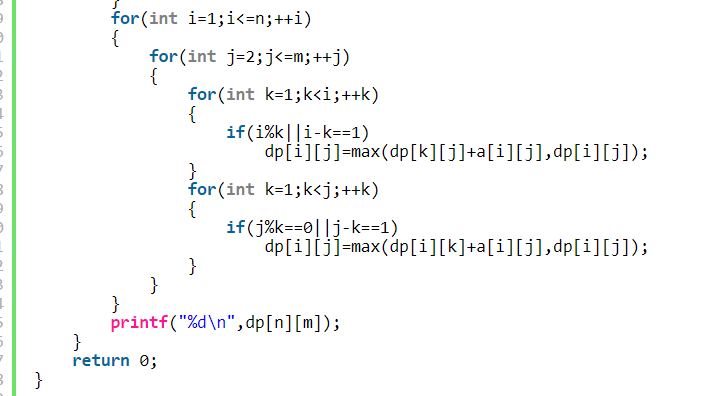
G：难点在于理解题目。。将题目看懂了就是一个简单的递归程序，注意下边界判定



H：一道难题，题目使用到了动态规划的知识。

使用DP[i][j]来表示到第i行第j列时的最大收益，因为只能向下走和向右走，所以可以保证在计算DP[i][j]时，它的所有左上部分的DP数组值确定，且是走到第X行第Y列的最大收益，所以可以不断推广到n行m列得出最后答案，具体请理解代码：





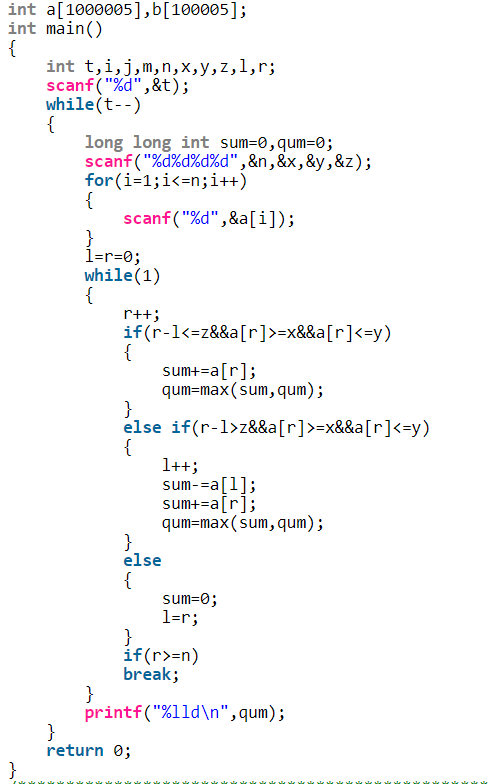
I：一道难题，运用了尺取的知识。

使用两个下标l和r，代表目前表示的区间，首先由于所有数都>0，所以当l确定时，一定有r越大数组值越大。

所以当l-r区间满足条件时，不断移动r，并且用sum加上进入区间的数的值

当l-r区间不满足条件时，不断移动l，并且用sum减去出区间的数的值

这样最多l和r分别从1-n跑一次，比枚举所有区间看是否满足条件需要跑n\*（n-1）/2次时间上优化很多。



J：一道很有趣的思维题，从大到小不好变，那么我们其实可以考虑葱小到大变化，

比如初始3边是4，那么最优的变化方案一定如下：

4 4 4

4 4 7

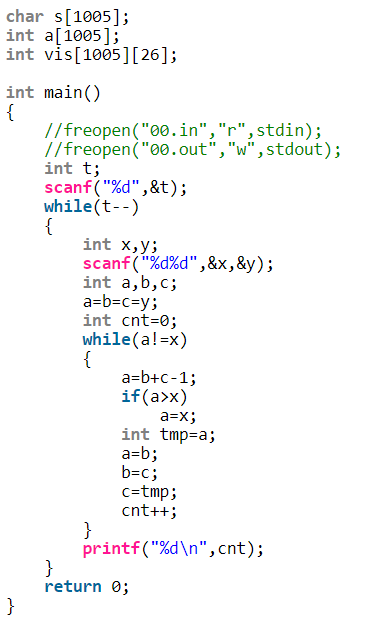
4 7 10

7 10 16

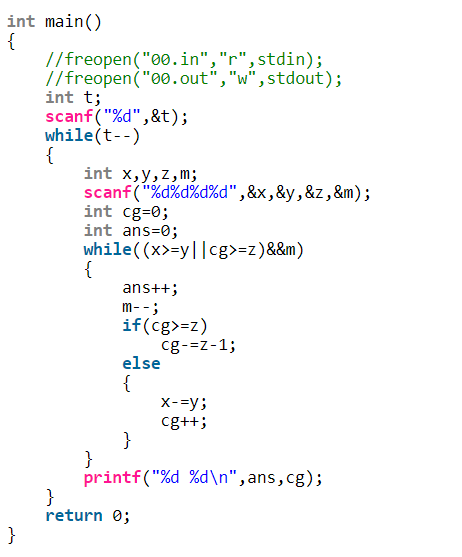
10 16 25

……

找到这个规律后就很简单了



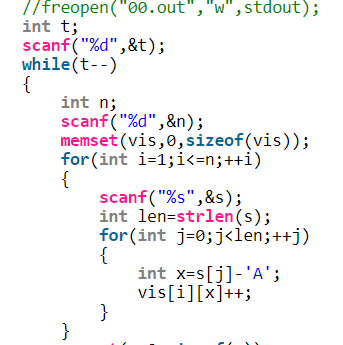
K题：按照题目给定的兑换规则兑换即可，如果瓶盖够就用瓶盖换，否则就用钱换，直到瓶盖和钱都换不了或者已经没有水为止



L：一道难题

首先使用一个数组记录每个字符串中每个字母出现的次数，然后分组即可

这是记录每个字母出现次数：



这是分组：



M：按照公式计算即可，需要注意的是直接计算会超时，仔细观察可以发现bi%bl%(bl+bl+1)……=bi%bl,所以只需要计算1次求模运算即可

