A

题意：给出长度为n(≤1e5)的小写字母字符串s，一开始有三个空串t1,t2,t3，有q(≤1000)次操作，每次操作对这三个字符串中某一个末尾添上一个小写字母或者删掉末尾的字母，每次操作之后进行一次询问，能否将这三个字符串作为s中不相交的子序列。保证任何时候t1,t2,t3的长度都不超过250。

题解：设dp[i][j][k]表示对三个串的指针分别走到i,j,k时在s中对应的最短前缀的长度，如果不能匹配则为n+1。这样每次操作之后更新一次dp，只需要判断当前的dp值是否超过n即可。因为每次只更新一个串，所以每次操作只需要O(2502)，总复杂度则为O(q∗2502)，其中可以加入nxt数组来加速，nxt[i][j]表示在s中第i个字母之后的第一个字母j所在的位置，如果不存在则为n+1。

blog链接：<https://blog.csdn.net/qq_41289920/article/details/89739551>

B

题意：奶牛们有一个工厂用来生产奶酪，接下来的N周时间里，在第i周生产1 单元的奶酪需要花费ci，同时它们也有一个储存室，奶酪放在那永远不会坏，并且可以无限放，每一单元奶酪放在那的价格恒定为每周s。然后奶牛在第i周会交付顾客yi的奶酪，让你求最小花费。

题解：第i周的奶酪，可以在第i周生产，也可以在前几周生产，然后储存。通过把s转化为花费，跟原有花费去比较，取一个最小值，这样从头到尾，每一周都可以取得一个花费的最小值。贪心求解。

blog链接：<https://www.cnblogs.com/sdfzsyq/p/9812936.html>

C

题意：你初始有D，A都为0，游戏有n轮，每轮有三个选项你可以选择其一：对敌人造成A+ai的伤害、使自己的A增加ci、使自己的D增加bi。每回合开始时你的A会增加D，保证ai, bi, ci为正整数。

题解：dp[i][j][k]表示到第i个位置，攻击了j次，j次操作的位置下标之和是k，的最大值。倒着dp。

blog链接：<https://blog.csdn.net/qq_45523675/article/details/99670953>

D

01背包。

E

题意：消除一个字符串，有两种消除方式：1、消除一个单独的字母，代价为a；2、s[j]到s[k]是s[1]到s[j - 1]的子串，那么s[j]到s[k]可以消除，代价为b。问最小的代价。

题解：令dp[i]表示考虑到第i个位置时的最小代价。处理出f[i][j]表示以i位置结尾的和以j位置结尾的相同子串的最大长度。转移：dp[i] = min( dp[i-1] + a , dp[ i - min( i-j , f[i][j] ) ] + b )。（枚举j）

F

题意：有n个数，对于每个数i，可以自己用3个来形成一个答案，或者是用连续的3个数形成一个答案。输出最多的答案数。

题解：f[i][j][k]表示考虑前i个数，以i为开头的三元组有j个，以i为中间的三元组有k个。

f[i][j][k] = max( f[i][j][k] , f[i-1][k][p] + p + (cnt[i]-j-k-p)/3 )。

blog链接：<https://blog.csdn.net/weixin_44316314/article/details/104804627>

G

完全背包。

H

题意：对于一个数组，求和为p的倍数的不相交连续子段的最大数目。

题解：f[i]表示考虑到i位置的最大数目，f[i]=max(f[i-1],f[last[cur]]+1)，其中cur表示当前位置的前缀和对p取模，last[cur]表示前缀和取模后同为cur的上一个位置。

blog链接：<https://blog.csdn.net/qq_42840665/article/details/107676598>

I

数字三角形。