subtask1~4

直接暴力记忆化

subtask5~7

大力猜结论：每次只会操作[i,i+1]

~~尽管这个结论不全~~，但是仍可以设f[i,j]表示a'[i]=j的答案来做到O(nQa)

subtask8~12

考虑确定贪心策略，设k为最大的$2^k <= n$，则按照第k位的01可以分成两段

结论：每次选则两段分界处的两个直到选完，然后分成两个子问题

证明：

设选的区间为[x,x+1]，考虑反证，如果存在反例则有两种情况：

①某个区间包含[x,x+1]

则该区间最大可能的贡献为$2^{k+1}-1$，等于[x,x+1]的贡献，但其完全包含[x,x+1]，显然不如选[x,x+1]优

②存在[L,x]和[x+1,R]

则每个的值最大为2^k-1，和最大为$2^{k+1}-2$，小于$2^{k+1}-1$，所以换成[x,x+1]更优

综上，不存在跨过[x,x+1]的区间以及同时选[L,x]和[x+1,R]，所以一定是把[x,x+1]选到不能选为止

接着左右两边最高位相同，所以可以忽略最高位变成子问题[1,x],[x+1,n]

从上往下递归做，一次询问的时间为O(n)，总时间为O(nQ)

subtask13~20

考虑一次修改的具体变化，把操作树建出来，显然每层操作的区间之间无交

比如对[1,7]询问，第一层操作[3,4]，第二层操作[1,2],[5,6]，第三层操作[2,3],[4,5],[6,7]

可以发现，把最后一层除掉后的所有操作区间之间无交，因此可以分成两层做，第一层操作[2n-1,2n]，第二层操作[2n,2n+1]，这样得到了一个好写的O(nQ)做法

维护第一层操作后的a'，修改只会影响到两个a'，算答案只需要四个a'，这样就可以O(n)预处理O(1)询问了