shop

显然有一个贪心算法：按深度从大到小考虑每个结点，对该结点上的每个顾客选择子树中未被选择的价格最大的商品。

可以得到一个等价的贪心算法：按价格从大到小考虑每个商品，选择祖先中未被选择的深度最大的顾客。

可以并查集维护。

时间复杂度 $O(n\alpha(n)+m)$ 。

xor

两个线性空间的交还是线性空间，所以两个线性基代表的线性空间的交仍然可以用一组线性基来描述。

考虑怎么求两个线性基的交。

引理：若 V_1, V_2 是线性空间，B_1, B_2 分别是他们的一组基，令 $W = B_2 \cap V_1$ ，若 $B_1 \cup (B2 \setminus W)$ 线性无关，则 W 是 $V_1 \cap V_2$ 的一组基。

但是 $B_1 \cup (B_2 \setminus W)$ 有可能线性相关，这时我们只需要换一组基即可。

考虑依次加入 B_2 的每个元素，维护 B_2' ，使得 B_2' 和 B_2 等价且 $B_1 \cup (B_2' \setminus W)$ 线性无关。

假如当前加入的元素为 x ，若 x 不能被 $B_1 \cup B_2'$ 表示，那么直接在 B_2' 中加入 x 即可；否则 x 一定能被 $B_1 \cup (B_2' \setminus W)$ 的恰好一个子集表示，设 $x = \mathrm{xor(S)} \hat{} \mathrm{xor(T)}$ ，其中 $S \subseteq B_1, T \subseteq B_2' \setminus W$ ，在 B_2' 中加入 $\mathrm{xor(S)}$ 即可。

综上，可以在 $O(log^2U)$ 的时间内求两个线性基的交。

线段树，每个结点维护对应区间线性基的交。

对于每个询问，如果每次合并 $O(\log n)$ 个线性基，复杂度为 $O(m\log^3)$ ，如果在这 $O(\log n)$ 个线性基上分别查询能否得到 x ，时间复杂度为 $O(m\log^2)$ ，可以通过本题。

modulo

设原来的序列为a，定义差分序列b，b_i=a_i-a_{i-1} 则对a[l..r]+1相当于b[l]+1,b[r+1]-1(若r=n则只有b[l]+1)； a变成0就相当于b变成0。 定义解的序列c，c[i]表示b[i]被加了几，就是说i每作为一次左端点c[i]就+1，每作为一次右端点+1c[i]就-1。 注意，b[i]是模k意义下的，但c[i]不是。 只对[l,n]操作可以得到一个初始解c0，c0[i]=(k-b[i])%k。 注意到，对于答案的解序列，有c[i]=c0[i]或c[i]=c0[i]-k。 一个c合法的条件就是所有前缀和非负，一个c对应的操作次数就是c中正数的和。 考虑从c=c0开始，每次贪心将一个c[i]从c0[i]变成c0[i]-k，来得到答案的解序列。 那么显然一次改变会将答案减少c0[i]，而对前缀和的影响为将[i..n]减少k。 因此我们按c0[i]从大到小贪心即可。