包

30 分解法：2^n 爆搜即可。

50 分解法：30 分做法加剪枝优化一下。

70 分解法：因为 $n \leq 6$，可以从 $1-n$ 枚举选取物品个数，然后爆搜，因为总的选取方案数为组合数 C_k^6，这个数很小，所以跑过第三部分的小数据毫无压力。

100 分解法：

$Meet-in-the-Middle$ 思路，这是搜索题目中一种重要的想法！

先预处理前一半物品，枚举这部分物品的所有可能选取集合，并用 set 存下来 $S[i]$ 里面存放选取物品数量不超过 i 的所有可能重量集合。同样，枚举后一半物品的所有选取集合，并到对应的 set 里面二分查找满足要求的最大权值即可。（当然这里也可以不用 set，改用数组和排序在没有 O2 优化的情况下速度更快）

在实现时，用**搜索**来枚举选取的集合是最常见的做法。但由于枚举集合的特殊性，我们也可以使用**状态压缩**枚举子集。

时间复杂度 $\mathcal{O}(tlogt)$ 其中 t 是 $2^{n/2}$

数列

小数据暴力即可，做法很多。

首先观察数列本身，注意到这个数列增长非常快，要不了几十项就超过 10^9 了。

再进一步观察，mex 那一堆不过是增加了差值集合中未出现的最小的正整数，那么这也就很容易证明得到题目中所说的那一个性质。

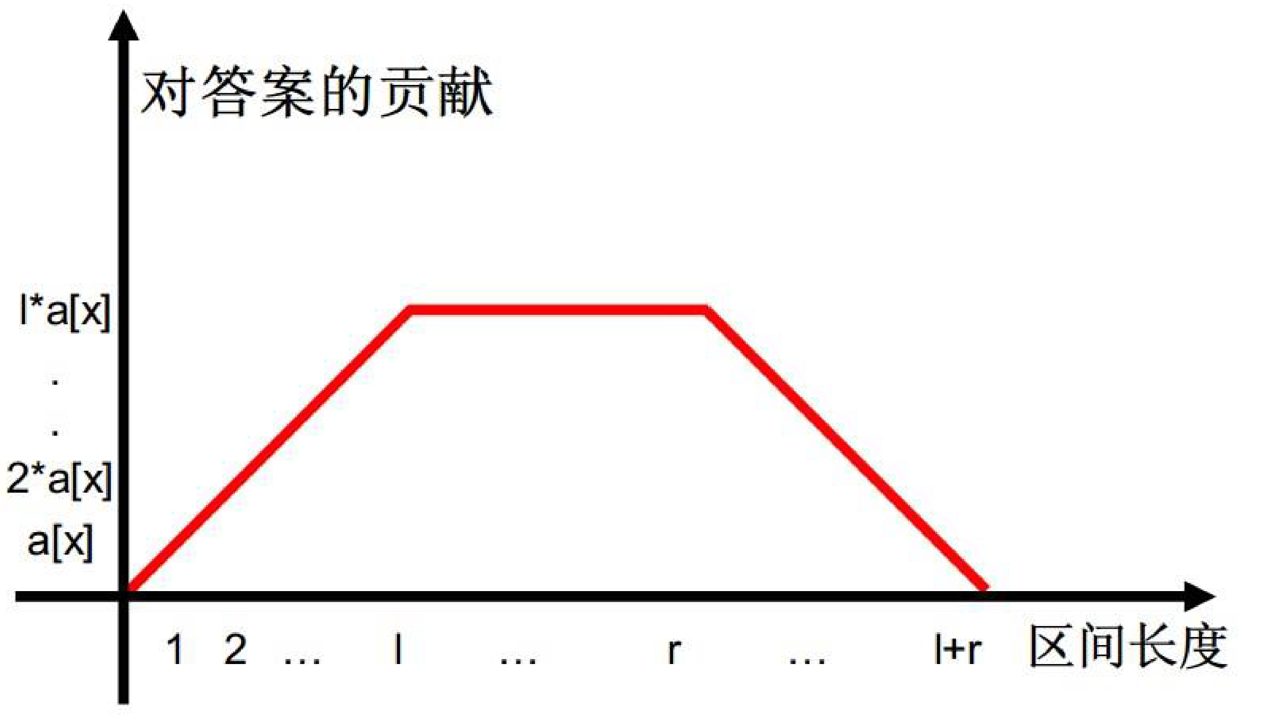
询问的值不大于 10^9 ，那么我们预处理出当 $a_i \leq 10^9$ 时所有可能出现的差值，那么未出现的只有可能是后面的某连续两项之差了。

经观察会发现，每增加两项只会增加一个最小的未出现的差值 ，那么只需要二分求出小于它的在预处理中出现过的差值个数即可。

护甲

30 分解法：暴力。

50 分解法：分析每一个元素对于不同询问区间长度的贡献，l 为它到左边第一个比他小元素的距离，r 为它到右边第一个比他小元素的距离，那么可以发现，贡献呈这种趋势：



于是可以预处理每个元素的 l, r，$\mathcal{O}(1)$ 即可得到每个元素对所询问区间长度的贡献，于是 $\mathcal{O}(n^2)$解决。

100 分解法：注意到，所询问就是如上图中多个梯形，在某一个 x 处的权值和，而上述梯形又显然可分成三段，两段公差为 1 的等差数列和一段常数数列，于是三段分开用**二阶差分**前缀后缀 $\mathcal{O}(n)$ 预处理一下，每个询问 $\mathcal{O}(1)$，于是最终复杂度 $\mathcal{O}(n)$。

头盔

20 分做法：任意暴力。

60 分做法：没有 3 类型操作，这就意味着权值是恒定的，离线即可。输入时处理出每一个询问时相应的根，再将询问按照询问的权值排序，dfs 序（以 1 为根） + 权值树状数组解决即可（如果新根在以 1 为根时询问点的子树外，对应 dfs 序区间无影响，若在子树内，则新子树对应 dfs 序区间分为 $[1 \cdots p], [q \cdots n]$前后两个区间）。另外需要注意对点的权值进行离散化。

100 分做法：dfs 序后分块，并维护块内有序，这样询问时，到对应的块中二分查找一下即可，多出来的没有在整块内的部分暴力枚举处理即可，时间复杂度 $\mathcal{O}(ntlogt)$ 其中 t 可以为 $\sqrt{n}$。