

**Note:** 均有可能存在更优或更优美的算法, 欢迎各位联系菜鸡出题人, QQ: 352912432

## 宝藏

将所有宝藏按其价值排好序。

考虑枚举从小到大开采的宝藏个数, 答案必定不增, 可以用双指针扫。

考虑如何单次判断某个宝藏的价值能否成为答案, 则必定是从价值高于它的和价值低于它的各选 $x$ 个消耗时间最小的, 这个可以用主席树来维护。

时间复杂度 $O(n \log \max(t_i))$

## 寻找道路

可以先把从1号点能只走0边就能到达的点都缩成一个点。

接下来我们要让路径经过的边数尽量少, 在此前提下最小化字典序。

可以考虑直接bfs, 每次从队列前部取出所有距离相同的点并先遍历0边再遍历1边更新答案, 具体可以见std。

时间复杂度 $O(n + m)$ 。

## 猪国杀

最优策略肯定是取所有牌中点数最小的几张

考虑固定选了哪些牌, 并求出有多少个方案使得选的牌中前几个恰好是这些, 那么有

$$ans \times A^n = \sum_{i=0}^n \sum_{j=1}^A \sum_{k=1}^{n-i} g_{i,j,k} \times \binom{n}{i} \sum_{t \geq k} \binom{n-i}{t} \times (A-j)^{n-i-t}$$

其中 $g_{i,j,k}$ 表示有多少个长度为 $i$ 的正整数序列满足每一个数字不大于 $j$ 且所有数字总和不超过 $k$

大概就是枚举选的牌中的最大值 $j$ , 最大值个数 $k$ , 以及选了 $i$ 个小于 $j$ 的牌。

可以用背包计算, 也可以枚举有多少个数字大于 $j$ 容斥计算, 那么有

$$ans \times A^n = \sum_{i=0}^n \sum_{j=1}^A \sum_{k=1}^{n-i} \left( \sum_{t=0}^i (-1)^t \binom{i}{t} \binom{m-k \times j - t \times (j-1)}{i} \right) \times \binom{n}{i} \sum_{t \geq k} \binom{n-i}{t} \times (A-j)^{n-i-t}$$

组合数为0的时候能直接跳过, 那么直接按照上式计算大概是 $O(n^2 m \log m)$ 的。

## 数树

可以考虑求出 $T1$ 的每个连通块有多少个双射 $f$ 是合法的之和, 然后除去 $T2$ 的自同构方案数即可。

我们只要固定一个 $T1$ 的根即可, 但是需要枚举 $T2$ 的根并每次进行dp。

令 $dp_{u,S}$ 表示 $u$ 的儿子已经向 $T2$ 的 $S$ 中的点建立双射的方案数, 每次枚举当前儿子与哪个点建立双射进行转移。

然后将 $T1$ 每个点与 $T2$ 的根配对的方案数加起来。

时间复杂度 $O(nm^22^m)$ 。