绝顶之战

ix35

ltst、Kubic、水老师 (I’m your biggest fan)

绝顶之战

题目大意：

有个长度为 m 的空间和 n 个长度为 $a_1, \ldots, a_n$ 的物品，按顺序尝试将物品放入空间，物品必须占据一段连续空间，且如果能放则必须放。判定每个物品集合是否可能恰好是被放入的集合。

数据范围：

$n \leq 14,\ m \leq 10^{16}$

绝顶之战 - 算法一

暴搜。

15 分。

绝顶之战 - 算法二

聪明一些的暴搜。

? 分。

绝顶之战 - 算法三

很牛的搜。

枚举一个放进去的子集。枚举空间它们空间上的顺序。剩下的是个差分约束。

有一些相对位置是不重要的，实际上本质不同的情况数会少很多。

~100 分（根据 ltst）。

绝顶之战 - 算法四

每加入一个物品将会把一段连续空间划分成两段，所以整个过程可以用一棵二叉树来表示：叶结点是最终的连续空间，非叶结点是被加入的物品。那么对于子集外的物品，限制就变成了要求某棵子树的权值（长度）和不能超过某个值。

对子树进行状压 DP。令 $f(i,R,S,T)$ 表示：$\{i,\ldots,n\}$ 被分成三个子集 R,S,T，其中 R 是子集外的物品（限制），S 是子树内的物品，T 是子树外（但在子集内）的物品，这种情况下子树大小的最大值（最小值就是物品长度之和，并且最小值到最大值之间的值都能取到）。

绝顶之战 - 算法四 - 转移 1

当 $i \in R$ 时，是对整个子树的限制：

$$f(i,R,S,T) = \min\{a_i - \varepsilon, f(i+1, R\backslash\{i\}, S, T)\}$$

ε 是无穷小，实现时把它去掉，但是最后 f 是一个无法达到的上界。

绝顶之战 - 算法四 - 转移 2

当 $i \in S$ 时，i 是根，枚举左右子树：

$$f(i,R,S,T) = a_i + \max_{S_1+S_2+\{i\}=S}\{f(i+1,R,S_1,T \cup S_2), f(i+1,R,S_2,T \cup S_1)\}$$

绝顶之战 - 算法四 - 转移 3

当 $i \in T$ 时，不用特殊处理：

$$f(i, R, S, T) = f(i+1, R, S, T \setminus \{i\})$$

绝顶之战 - 算法四

最终，如果子集 S 满足 $\sum_{i \in S} a_i \le m \le f(i, [n] \backslash S, S, \emptyset)$，那就可以作为答案。

时间复杂度：$O(4^n)$，或者可能可以 $O(\text{poly} \times 3^n)$。

100 分。

绝顶之战 – 吐槽