

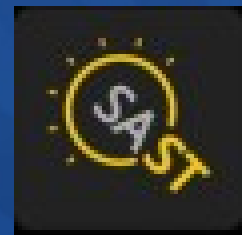
# 诡异的文本处理

(CleverTick's Editing eXecutor)

梁泽宇

Mato\_No1

Unknown OJ 用户群



# 命名来源

- 大家可以注意到，这篇 CTEX 的题解并不是用 `ctex` 写的囧……
- 原因：某日，我的 `ctex` 坏了，用了很多方式（包括重装）都木有修复……
- 于是我一怒之下想到了一种可怕的方案……
- 出题黑 `ctex`……
- 于是就有了 `C`lever`T`ick's `E`ditting `eX`ecutor…

# 题目大意

- 以有限状态自动机为框架……
- 给出一个有限状态自动机的结点和边，边权（状态转移）需要你来确定……
- 某些结点上可能有输出装置，可以在到达此结点时在输入串末尾写上一个字符……输出装置也需要你来确定……
- 给定若干个输入串及其输出结果（限制条件）……
- 同时给出每条边各种转移类型、每个结点各种输出装置类型的概率权值……
- 需要找到一种方案满足所有限制条件且权值之和尽可能小…  
…
- 提 (xi) 交 (wen) 答 (le) 案 (jian) 型试题……

# 数据特点

- 对于提 (xi) 交 (wen) 答 (le) 案 (jian) 型试题，读题之后首先应该观察数据的特点……
- 本题数据特点很容易发现……最典型的特点是所有的数据都有  $s=2$ ，即字符集为  $\{0, 1\}$ 。
- 还有别的特点么囧……

# 数据特点

- 首先数据可以分为两大类：
- Case #1~5：所有非终止结点上均不存在输出装置，终止结点均存在输出装置且类型确定。对于所有输入串，输出结果均为一个数，即到达终止结点时原有内容刚好读完，同时写上该终止结点的输出装置类型……
- 标准的有限状态自动机模型……
- 可以认为写有 1 的终止结点表示接受，写有 0 的表示拒绝……等价于给出所有状态转移使得所有限制条件（对于给定的输入串，接受或拒绝）全部满足，且权值之和尽可能小……

# 数据特点

- Case #6~10 : 所有结点上都有可能存在输出装置且类型不定, 对于所有  $1 \leq i \leq 5$  , Case # $i$  与 Case # $(i+5)$  的自动机结点数、边数和构造特点相同……
- 图灵机模型, 但可用自动机的思想求解……

# 数据特点

- 各个测试点的自动机构造也各有其特点……
- Case #1 && #6 : 深度为 10 的满二叉树 ;
- Case #2 && #7 : 深度为 10 的 2- 树 ;
- Case #3 && #8 : 深度为 10 的普通二叉树 (由题意, 自动机中度为 1 的结点起到的是阻碍某个字符通过的作用) ;
- Case #4 && #9 : 在深度为 10 的普通二叉树基础上补上度为 1 的结点所缺的边, 均为前向边 (指向后代的边) 且树上连接该边两端点的路径上的点 (不含两端点) 均不是度为 1 的结点 ;
- Case #5 && #10 : 深度 (层数) 为 51 的分层图, 每层不超过 5 个结点, 最上层为初始结点, 最下层为终止结点, 且每个非初始结点有至少一条入边, 每个非终止结点都有两条出边。

# 讨论时间

- 关于本题，大家有什么好的想法或做法……
- 希望能够上台讨论……



# Case #1~2 的动态规划解法

- 对于前两个测试点，注意到将所有输入串按字典序递增排序后，所有最终到达的结点落在同一子树内的输入串刚好形成一个区间……
- 因此我们可以使用动态规划解决这两个点……
- 设  $F[i][j][k]$  表示最终结点落在以  $i$  为根的子树内的输入串区间为  $[j..k]$ （排序后），最小的权值和；
- 枚举从  $i$  出发的转移哪个是 0 哪个是 1，取权值和较小的那个即可。
- 时间复杂度  $O(LZ \log Z + NZ^2)$  ……

# NPC 问题??

- 本题的后 8 个点，目前没有好的多项式解法，可能是 NPC 的……
- 我们只有尝试用非多项式算法解决这些测试点……
- 搜索???
- 问题是怎么搜……

# 局部搜索算法

- 本题前 4 个点可行解的构造较为容易……
- 由于自动机为树形或近似树形，可以使用每个输入串的最终结点的集合表示一组解……
- 使用局部搜索算法（模拟退火等）可以获得很优的解……

# 记忆化搜索 + 局部搜索估价

- 注意到动态规划也可以用记忆化搜索的方式实现……
- 从上往下搜索，状态为所有输入串在每层结点上的分布（Case #1~4）或所有输入串在每层结点上的分布及其纸带内容（Case #6~9）……
- 到达每个结点时，用之前所述的局部搜索算法在以该结点为根的子树内进行估价，可以估计出这里最优解的近似值……
- 在记忆化搜索的过程中再次进行局部搜索，比如每层保留前  $K$  优解等……

# 加入贪心思想的搜索

- 本题还有另一种思路：使用贪心思想在搜索过程中改变顺序。
- 我们可以对于每个输入串，启动自动机进行模拟状态转移。
- 对于未确定转移的结点，采用“先小后大”的搜索方案（即先尝试两种转移方案中较小的那个并填上，后尝试较大的那个填上），中间若发现输出结果不满足限制条件则剪枝；
- 随机输入串的排列顺序多次搜索，可得到相当优的解。
- 可以加入其它剪枝提高搜索效率……
- 对于第 10 个点，可行解数量很少（可能只有一组），所以只需找到一组可行解即可得到满分。

# 一些启示

- 没有最优，只有更优……
- 很多实际问题没有能保证得到最优解的解法，此时使用合适的近似算法得到尽可能优的解就显得尤为必要……
- 希望大家充分发挥人类智慧、探索、测试、改进解决方案的能力，去得到尽可能优的解，优化我们的世界。

谢谢大家！！