北京邮电大学第五届程序设计竞赛

题目讲解

比赛题目通过情况

- A Graph 27 通过 174提交 15.52%通过率
- B Earthquake 8通过37提交21.62%通过率
- C Water Tanks 58通过157提交36.94%通过率
- D Longest Subsequence 73通过201提交36.32%通过率
- E Fruit Ninja 0通过6 提交0通过率
- F Scheduling Problem 53通过157提交33.76%通过率
- G Tetris 3通过20提交15.00%通过率
- HBG 6通过106提交5.66%通过率
- IT-Residual Set 22通过178提交12.36%通过率
- J Shortest String 4通过15提交26.67%通过率

比赛队伍做题情况

多样性,中等题目里出的题目不相同. 62.5%的队伍做出3道题以上 由于题目难度稍大,没有队伍能做到 难题

Problem A Graph

- ●由于题目中的数据范围不大
- ●可以直接保存整个地图
- ●在查询的时候,使用dijk求最短路即可
- ●注意数据中会存在重边,所以保存地图的时候需要注意下细节
- ●数据如果是50个联通块,每个联通块最多50个点时如何处理?

Problem B Earthquake

- ●树与树之间是独立的
- ●dp[v][1][1]表示以节点v为根的子树,都被覆盖且v被选中
- ●dp[v][0][1]表示以节点v为根的子树,都被覆盖且v没被选中
- ●dp[v][0][0]表示以节点v为根的子树,除了v以外
- •dp[v][1][1]=w[v]+sum(min(dp[v.son_i][1][1],dp[v.son_i][0][1]
 ,dp[v.son_i][0][0])) 0<i<v.son_size

 dp[v][0][0]-cv.co(dp[v.son_i][0][1]) 0<i<v.son_size</pre>

dp[v][0][0]=sum(dp[v.son_i][0][1]) 0<i<v.son_size 对于

dp[v][0][1],首先计算

temp=sum(min(dp[v.son_i][0][1],dp[v.son_i][1][1])),如果全部从dp[v.son_i][0][1]转移过来,需要在其中找到一个子节点,使得dp[v.son_i][1][1]-dp[v.son_i][0][1]最小,temp加该最小值即为dp[v][0][1]每个树的最优值为min(dp[p][0][1],dp[p][1][1])最后累加所有棵树的最优值

Problem C Water Tanks

- ●中等偏下的题
- ●求出给定体积水的时候,问可以覆盖水箱的 高度为多少
- ●由于水的体积和覆盖水箱的高度为单调关系.
- ●所以可以二分高度后计算水的体积.

Problem D Longest Subsequence

- ●简单题
- ●由于数据范围比较小,可以使用n^2枚举的方法.
- ●每次处理一个单独的值.
- ●注意处理n=0的情况

Problem E Fruit Ninja

- ●线段树,难题---无人通过
- ●Trick:count=0时方案数是inf.
- ●0边加上后的最大值区间是当前X内的垂直区间
- ●1边减去后的最大值区间是当前X右侧的水平 开区间

Problem F Scheduling Problem

- ●简单模拟,题意一开始描述有些误解.
- ●由于题目比较长,大家都没有选择去做.
- ●读清楚题意后直接按照题意模拟即可

Problem G Tetris

- ●难题
- ●需要对状态进行设计
- ●用[x][y][z]3维表示对应列的高度,第一维表示当前dp到第几行.
- ●状态转移显然
- ●由于状态转移时需要一定的编码量,所以通过的队伍比较少

Problem H BG

- ●中等题
- ●直接暴力做的复杂度是2^38次方.
- ●所以需要一个技巧,将集合分为两个子集.
- ●对两个子集做状态dp处理出所有可能的和.
- ●然后对和进行二分即可

Problem I T-Residual Set

- ●中等偏下题.
- ●打表找规律.
- ●可以发现公式是(p-1)/gcd(t,p-1)
- ●由于具体证明比较长,将会发布在论坛

Problem J Shortest String

- ●中等偏上题
- ●使用KMP预处理;
- ●然后再做一次求汉密尔顿路的状态dp即可