

Lorem Ipsum Dolor

DAY4 题目选讲

XDU_School of
Software
glory_g



Codeforces 839C

给定一棵有 n 个节点的树，某个人骑着一匹马等概率走向任意相连的未经过节点的马从1号点开始旅行，当到某个点的无法移动旅行结束。求期望的旅行长度(每条边的长度为1)

solution : 先用dfs 将无向图转化为以1为根有根树然后在树上做一遍dfs 即可得到答案

CodeForces 839D

给你 n 个数，当其中的一些数的最大公因数大于1时，就将它们的最大公因数再乘以这些数的个数加到答案上，问你最后答案是多少？

solution：注意到 a_i 的范围，我们便可以用一个数组记录每个数出现的次数，然后从大到小扫描每种可能的最大公因数的情况，使用容斥原理推公式即可得到答案。

POJ - 2349 Arctic Network

有 S 颗卫星和 P 个哨所，有卫星的两个哨所之间可以任意通信；否则，一个哨所只能和距离它小于等于 D 的哨所通信。给出卫星的数量和 P 个哨所的坐标，求 D 的最小值

solution : 注意到克鲁斯卡尔求最小生成树的本质，贪心的选择可能边权最短的边加入集合中。一开始我们可以认为有 n 个连通块，每加入一条边相当于连通块数目减去1，直到连通块的数目为 s 输出加入的最大边即可

D - Count Color POJ - 2777

给你一个长度为 L 的区间，给你 n 中颜色和 m 中操作，每一次操作可以把一个区间染成一种颜色，或者是询问一个区间内有多少种颜色，输出询问答案

solution：区间修改，区间询问，线段树是符合这个题目要求的数据结构，注意到最多有30种颜色，我们可以用一个整数的二进制位进行状态压缩，认为二进制位是1的是有颜色的，0的没有颜色

CodeForces - 812C

给你 n 件物品和你拥有的钱 S , 接下来给出这 n 件物品的价格, 这些物品的价值不是固定不变的, 价格的变化公式是 $a[i] + k * i$ (i 代表第 i 件物品, k 代表你选择买的物品数量, $a[i]$ 为物品的底价), 现问你最多能够买多少件物品和所买物品总和, 输出时应该使得所买物品总和尽量小

solution : 不难发现答案 k 满足二分性质 (具有单调性), 所以我们可以二分答案 ans 再用贪心的策略判断答案是否可行就可以了。

POJ - 3279

有一个 $n*m$ 的格子，每个格子都有黑白两面（0表示白色，1表示黑色）。我们需要把所有的格子都反转成黑色，每反转一个格子，它上下左右的格子都会跟着反转，求一种翻转方案使得全部翻转成黑色。

solution：不难发现每一个格子最多被翻转一次 注意到 m 、 n 的范围，我们便可以暴力枚举第一行每一个是否进行翻转，第一行确定以后剩下的每一行也就都确定了，最后判断最后一行是否全被翻转成黑色即可

UVA - 11624

有个人被困于火场中。火焰每分钟蔓延1格，人每分钟移动一格，求出人是否会被困于火场，或者逃脱，如果逃脱，求出最短的时间。

solution : bfs

CodeForces 799C Fountains

你有两种不同的货币;
余额分别为 c 和 d
然后有 n 种商品;
每种商品只能由两种货币中的某一种购买;
(且每件商品都有它的美丽值和对对应货币的花费)
让你用这两种货币的余额严格买两件商品;
要求让美丽值最大;
输出最大的美丽值;

solution : 先进行预处理, 处理出所有价格为 i 买一件的最大美丽程度, 然后分类讨论一下, 取最大值即可

Codeforces 821E

我们现在位于 $(0,0)$ 处，目标是走到 $(K, 0)$ 处。
每一次我们都可以从 (x, y) 走到 $(x+1, y-1)$ 或者 $(x+1, y)$ 或者 $(x+1, y+1)$ 三个位置之一。
现在一共有 N 段线段，每条线段都是平行于 X 轴的。
我们如果此时 x 是在这段线段之内的话，我们此时走到的点 (x, y) 需要满足 $0 \leq y \leq C_i$ 。
现在保证一段线段的终点，一定是下一段线段的起点。
问我们从起点走到终点的行走方案数。

PS：这个题本来
是想用来防ak的
结果..... orz Bruce

xuan

对于这个题目我们很容易看出

$$Dp[i][j] += Dp[i-1][j] + Dp[i-1][j-1] + Dp[i-1][j+1]$$

看到这么大的数据范围，按照这种方式dp 时间复杂度和空间复杂度都受不了，然后看到y的范围只有15，注意到加上这个方程是一个一阶线性递推，不难想到用矩阵快速幂去优化这个dp

空间复杂度怎么降？ ？ ？ ？ ？

这个矩阵该怎么写？ ？ ？ ？ ？

空间可以用滚动数组直接解决

