

2020.08.29-2020.09.04 周报

团队训练

wzx27

专题

暂无。

题目

一些topcoder的题

比赛

暂无。

Infinity37

专题

暂无。

题目

[codeforce 1392部分题解](#)

比赛

暂无。

Zars19

专题

无。

题目

topcoder若干。

比赛

Codeforces Round #666 (Div. 1)

本周推荐

Zars19

来源 [TopCoder - 16346](#)

tag : 计数, dp

概述 \$n\times n\$ 方格，其中 \$T\$ 个位置放置标记，每行每列都最多放两个，问方案数。

答案：可以用 \$dp[i][j][k]\$ (第一维可以滚动) 表示处理完第 \$i\$ 行，有 \$j\$ 列放 \$0\$ 个且 \$k\$ 列放 \$1\$ 个的方案数，显然 \$n-j-k\$ 列放两个。然后转移方程就很好想了。结果只需要统计放置完第 \$n\$ 行满足 \$k+2(n-j-k)=T\$ 的加和即可。

comments: 巧妙的状态设计。

_wzx27

来源 [topcoder 16399](#)

tag : dp, 优化

概述：给一个网格，要在格子里放东西，每个格子周围（包括自己）最多只有一个格子放了东西，这里的周围只共享边或者顶点。给定网格的大小，问最少放 \$B\$ 个东西的方案数 \$R \leq 190\$

答案

容易想到状压 \$dp[f][i][num][s1][s2]\$ 表示填了前 \$i\$ 列，放了 \$num\$ 个东西且最后两列的放置状态为 \$s1\$ 和 \$s2\$ 的方案数，但由于 \$R\$ 并不小，最大可以是 \$13\$，所以这样过不了。

但由于题目的限制 \$s1\$ 和 \$s2\$ 两个状态有很多是不必要的，因此可以预处理出放两列的合法状态，发现远比 \$2^{13} \times 2^{13}\$ 小，在此基础上在转移即可。

comments：根据题目要求优化 \$dp\$ 的范围，是比较不落俗套的题目。

Infinity37

来源 [codeforces 778E](#)

tag : 枚举/dp

概述

给定一个n位的数字，这个数字中可能有未确定的位数，再给出m个整数，我们按照如下方式计算答案。

对于数字0~9，我们给出其权值，得到答案就是m个整数都加上给定的数字后，各个位数的权值之和。

现在我们希望答案最大，要求你输出这个答案。

答案

我们考虑如果没有进位的话这其实是一道贪心，但是如果加上进位，我们可以轻易的想到用dp来解决这个问题，我们设置状态 $dp[i][j]$ 代表前*i*位选择完，第*i*位上没进位的状态为*j*的最大答案。

但是这里出现了问题，如果*j*用一个二进制表示的话，会非常大，我们思考如何优化这个状态。

考虑如果所有给出的*m*个整数，按照第*i*位数字有序，那么*j*这个状态就可以转换成前*j*个没进位，这样*j*的状态成功从 2^m 变成*m*个。

comments:从二进制枚举到枚举的转换，巧妙的dp思想。

From:
<https://wiki.cvbbacm.com/> - CVBB ACM Team

Permanent link:
https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:wangzai_milk:weekly18

Last update: 2020/09/04 15:15