

搜索

袁浩天

IIIS, Tsinghua

2023.8.2

目录

① 搜索题目选讲

② 总结

CF 912E

给定 n 个质数 p_1, p_2, \dots, p_n , 求出第 k 小的质因数全在给定质数内的数, 保证这个数 $\leq 10^{18}$
 $n \leq 16, p_i \leq 100$

Solution

一句话题解: meet-in-middle + 二分

Solution

一句话题解: meet-in-middle + 二分

考虑将质因数分成两个集合 A, B , 分别预处理集合内的质因数能构成的数集合, 二分最后的答案, 枚举其中一个集合里的因数
two-pointer 计算另外一个集合内的因数上限计算方案数。

Solution

一句话题解: meet-in-middle + 二分

考虑将质因数分成两个集合 A, B , 分别预处理集合内的质因数能构成的数集合, 二分最后的答案, 枚举其中一个集合里的因数
two-pointer 计算另外一个集合内的因数上限计算方案数。

如果我们是按照小一半大一半来划分集合的话会造成其中一个集合过大, 所以我们可以把排序后按照奇偶位置划分。

CF 478E

如果一个数在十进制表示下每个数码要么比左右都小要么比左右都大，那么我们就称这个数是 wavy 的。

求第 k 小的是 n 倍数的 wavy 数，如果这个数大于 10^{14} 你只需要输出 -1 。

$n, k \leq 10^{14}$ 。

Solution

观察数据范围, 我们没法直接暴力枚举, 考虑 meet in middle.
预处理左一半和右一半满足 wavy 条件的数同时记录模 n 的余数。我们在合并的时候不仅需要保证左右的 wavy 性质还需要保证中间合并点的性质, 那么需要我们在预处理分类的时候讲端点值以及端点的大小关系也记进去。

复杂度 $O(\sqrt{n})$

Code

阅读 478Edreamoon.cpp

Luogu P3230

题目描述

[复制Markdown](#) [展开](#)

沫沫非常喜欢看足球赛，但因为沉迷于射箭游戏，错过了最近的一次足球联赛。此次联赛共 N 支球队参加，比赛规则如下：

1. 每两支球队之间踢一场比赛；
2. 若平局，两支球队各得 1 分；
3. 否则胜利的球队得 3 分，败者不得分。尽管非常遗憾没有观赏到精彩的比赛，但沫沫通过新闻知道了每只球队的最后总得分，然后聪明的她想计算出有多少种可能的比赛过程。

譬如有 3 支球队，每支球队最后均积 3 分，那么有两种可能的情况：

可能性 1 and 可能性 2

球队	A	B	C	得分	球队	A	B	C	得分
A	-	3	0	3	A	-	0	3	3
B	0	-	3	3	B	3	-	0	3
C	3	0	-	3	C	0	3	-	3

但沫沫发现当球队较多时，计算工作量将非常大，所以这个任务就交给你了。请你计算出可能的比赛过程的数目，由于答案可能很大，你只需要输出答案对 $10^9 + 7$ 取模的结果。

输入格式

第一行是一个正整数 N ，表示一共有 N 支球队。接下来一行 N 个非负整数，依次表示各队的最后总得分。输入保证 20% 的数据满足 $N \leq 4$ ，40% 的数据满足 $N \leq 6$ ，60% 的数据满足 $N \leq 8$ ，100% 的数据满足 $3 \leq N \leq 10$ 且至少存在一组解。

Solution

搜索, 剪枝:

- 每个人每个时刻的当前得分不能超过总分
- 如果某个时刻赢下剩下所有的比赛都不够总分就剪枝
- 由于总分和总场数是确定的, 同时胜场贡献 3 分, 平场贡献 2 分可以确定胜、平的总场数。
- 最后剩余人数很少的时候可以记忆化
- 可能的对称性剪枝。

Code

阅读 3230.cpp

Gym 100503B

如下图你需要完成一个 kakuro 拼图, 每个白色的格子需要填入一个 1 到 9 的数字, 每个 run 里面每个数字最多使用一次, 黑色格子上写着数字和的一些限制。 $(n, m \leq 6)$

		28	17	28	
	22				10
34					
14			13		
22			16		
	16				

Picture of the first sample input

		28	17	28	
	22	5	9	8	10
34	9	7	8	4	6
14	8	6	13	9	4
22	5	1	9	7	
	16	9	7		

Picture of the first sample output

Solution

搜索

- 每个位置最多属于两个 run, 在搜索的过程中记录每个 run 使用的数字集合以及当前的和。
- 如果某个时刻一个 run 剩余没有填的元素都按照最小的去填仍然超过, 或者都按照最大的去填仍然到不了就剪枝。
- 随机打乱格子顺序 (?)

Code

阅读 100503Baid.cpp

爬山算法

考虑所有的状态集合 S , 那么搜索算法可以抽象成以某种方式遍历整个状态集合同时统计是否合法/合法个数/最优解。

爬山算法

考虑所有的状态集合 S , 那么搜索算法可以抽象成以某种方式遍历整个状态集合同时统计是否合法/合法个数/最优解。

在搜索过程中我们可以去定义两个状态的”相邻”, 比如: 两个二进制串 (或任意字符串) 差正好一位的串, 棋盘上再下一步, ...

爬山算法

考虑所有的状态集合 S , 那么搜索算法可以抽象成以某种方式遍历整个状态集合同时统计是否合法/合法个数/最优解。

在搜索过程中我们可以去定义两个状态的”相邻”, 比如: 两个二进制串 (或任意字符串) 差正好一位的串, 棋盘上再下一步, ...

那么同时我们每个状态会有一个属性, 在最优化问题中这个属性就是最优化的值, 在合法性问题中这个属性需要我们去给定, 可以是对当前局面的一个评分, 也可以是当前局面能到达的最好局面 (估算) 的评分。

爬山算法

那么爬山算法可以描述为从某个初始状态开始, 每次选择最好的走到一个相邻状态直到当前状态的任意相邻状态都会变差停止。

爬山算法

那么爬山算法可以描述为从某个初始状态开始, 每次选择最好的走到一个相邻状态直到当前状态的任意相邻状态都会变差停止。我们可以发现这种算法会在一个局部极值停止不一定会是全局的最值, 那么一种优化方法就是执行算法若干次每次随机一个不同的起点, 这样增加了停在全局最值的概率。

爬山算法

那么爬山算法可以描述为从某个初始状态开始, 每次选择最好的走到一个相邻状态直到当前状态的任意相邻状态都会变差停止。我们可以发现这种算法会在一个局部极值停止不一定会是全局的最值, 那么一种优化方法就是执行算法若干次每次随机一个不同的起点, 这样增加了停在全局最值的概率。

在凸函数上爬山算法是有正确性保证的, 但在一般函数上正确性无法保证。爬山算法一般应用于已知的 NP-Complete 问题或者常规算法较难求解的问题, 多数情况下能取得比较不错的解。

模拟退火算法

模拟退火算法是基于爬山算法的一种优化, 同样也是使用某种方式对状态集合进行搜索。

模拟退火算法

模拟退火算法是基于爬山算法的一种优化, 同样也是使用某种方式对状态集合进行搜索。

简单的说, 爬山算法只能接受更优 (非更劣) 的状, 而模拟退火算法可以以某种概率走向更劣的解, 当然损失的权值越大概率越小, 同时随着迭代次数 (时间) 的推移概率越小。当 $t \rightarrow \infty$ 的时候模拟退火算法就是爬山算法。

模拟退火算法

算法流程:

- 1. 初始随机状态 $s = s_0$, 温度 $t = t_0$
- 2. 随机一个相邻状态 s' , 计算目标函数差值
 $\Delta C = C(s') - C(s)$
- 3. 以 $\min(1.0, \exp(\frac{\Delta C}{t}))$ 的概率 $s \leftarrow s'$
- 4. $t \leftarrow \gamma t$, 其中 γ 为温度衰减系数, 一般为 $1 - [10^{-7}, 10^{-3}]$
- 5. goto 2

JOISC2020 ㊦説の㊦子職人

您面前有一个 $R \times C$ 的网格，每一个格子里有一个团子，您可以横向，竖向，斜向地将三个连续的团子按顺序串起来，按顺序指可以串上中下，下中上之类的，但是不能串中下上，上下中之类的。

如果一串团子的颜色为绿，白，粉或者粉，白，绿，那么称这串团子叫 AK IOI 串。

求串最多 AK IOI 串的方法（我坚信做了几个 AK IOI 串就会 AK 几次 IOI）。

输入格式

第一行两个整数 R, C 代表网格大小。

接下来 R 行每行 C 个字符代表网格：

- P 代表粉色团子
- W 代表白色团子
- G 代表绿色团子

Solution

考虑将每一个合法的串子建一个点，如果两串共用了团子就连一条边，那么这个问题就抽象成了一般图最大独立集问题，这显然是一个目前无法在多项式时间内解决的 NP-complete 问题。

Solution

考虑将每一个合法的串子建一个点，如果两串共用了团子就连一条边，那么这个问题就抽象成了一般图最大独立集问题，这显然是一个目前无法在多项式时间内解决的 NP-complete 问题。

考虑贪心的做法，按照某个顺序加点，直到没有点能加为止。那么状态的相邻可以描述为：选择一个有冲突的串子加进去然后删掉有矛盾的串子，目标函数就是串子的个数。使用爬山或者模拟退火解决。

Code

阅读 JOISC2020.cpp

① 搜索题目选讲

② 总结

总结

搜索剪枝：可行性剪枝，最优性剪枝，对称性剪枝。

总结

搜索剪枝：可行性剪枝，最优性剪枝，对称性剪枝。
记忆化搜索

总结

搜索剪枝：可行性剪枝，最优性剪枝，对称性剪枝。

记忆化搜索

其他技巧：折半搜索， A^* ，模拟退火等

作业

- 模拟赛补题
- CF 912E,478E,100503B
- JOISC 2020 D4T2
- luogu P3230

Thanks!