

2020牛客暑期多校训练营（第九场）

Results

Summary

- Solved 7 out of 12 problems
- Rank 32/1041 in official records
- Solved 8 out of 12 afterwards

#	Who	=	Penalty	A	B	C	D	E	F	G	H	I	J	K	L	Dirt
3	大吉大利，今晚吃mian();	7	928	+	+8			+	+			+	+1	+2		61%
				00:09	04:55			00:49	00:49			00:14	03:13	01:39		11/18

Member Distribution

Solved	A	B	C	D	E	F	G	H	I	J	K	L
Pantw	√				√		O			√		
Withinlover												
Gary		√				√			√		√	

(√ for solved, O for unsolved, - for tried but not solved)

Solutions

A

```
print(eval(input().replace('(', '**(')))
```

B

考虑类似树上DP的过程，记录每个节点完成其子树所需的最小初始HP以及可以获得的HP。最小初始HP可以通过二分来求解，可以增加HP的子树直接选取，减小HP的子树贪心选取。

C

D

E

考虑 x, y 的质因数分解 $x = p_1^{u_1} p_2^{u_2} \dots p_n^{u_n}$, $y = p_1^{v_1} p_2^{v_2} \dots p_n^{v_n}$

再回过头看答案的式子,

$$\begin{aligned} & \prod_{i=a}^b \prod_{j=c}^d \gcd(x^i, y^j) \\ &= \prod_{i=a}^b \prod_{j=c}^d \gcd((p_1^{u_1} p_2^{u_2} \dots p_n^{u_n})^i, (p_1^{v_1} p_2^{v_2} \dots p_n^{v_n})^j) \\ &= \prod_{i=a}^b \prod_{j=c}^d \prod_{k=1}^n p_k^{\min(i u_k, j v_k)} \\ &= \prod_{k=1}^n p_k^{\sum_{i=a}^b \sum_{j=c}^d \min(i u_k, j v_k)} \end{aligned}$$

考虑对每个质因数 p_k 求 $\sum_{i=a}^b \sum_{j=c}^d \min(i u_k, j v_k)$

容易发现 i 固定后 $\sum_{j=c}^d \min(i u_k, j v_k)$ 至多由一段等差数列和一段常数组成。

分界点是 $\lfloor \frac{i u_k}{v_k} \rfloor$

那么我们枚举 p_k 再枚举 i 即可。

F

将所有衣服排序, 滑动窗口遍历排序后的序列, 保证窗口内有 m 间不同时间的衣服, 扫一遍对所有满足条件的状态求最小值

G

这个题很有趣, 提示我们不要乱写 Simpson 不是什么时候收敛都好的。

题意是给一个凸多边形, 可以绕原点转。给一条直线, 问随机转之后直线截凸多边形弦长大于 L 的概率。

容易发现我们可以把直线当成动的, 凸包当成静的。重要的参数只有初始时原点到直线的距离, 这个是决定直线轨迹的参数。

容易发现每次只要确定两端的边就可以确定这个弦变动的范围。

那么我们随机选个端点作为起点, 然后扫一遍找到这个端点的弦对应的另一条边。把所有点对应的两个角度按凸包顺序分别装进两个数组里, 扫的时候每次判断碰见的下一个端点属于哪一条边, 然后就直接把边编号记一下, 另开一个函数算这段角度区间对应的贡献。

至于这个贡献怎么计算，赛场上第一个想到的是辛普森，然后放弃了单峰性质，事实证明这么干挺蠢的（这个直接如同题解所说，把它峰值找出来，再二分找到两个分界点，就直接可以求出贡献为 1 的区间长度了。

H

I

推算一下会发现，选取最小的一位数和剩余数组成的最小数字是最优的解

J

这个题直接把所有 0 替换成 -1，然后对每条底边做一个左侧区域的前缀和，这个可以滚动数组维护。

然后就可以直接枚举两行作为上下底，用 bitset 直接维护中间可以作为墙的位置，然后连续段直接并起来处理。

具体处理方法：同一个块内，扫描并统计所有前缀和出现的次数。后面位置的前缀和如果是 $S[i]$ 那么 $S[i]-1$, $S[i]$, $S[i]+1$ 都可以与这个位置形成一个合法子矩阵，直接加进答案里就可以了。

K

先求出开始追击的节点，只有对追的人遍历所有节点记录他到每个位置的时间，再对逃跑的人遍历，记录所有可以比追击的人先到的节点，对所有可行的时间去最大值

Comments

ptw:

- 我决起而飞

Gary:

- 大概把会的都做了
- 读题要仔细，最好两人读题

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

Permanent link:
https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:mian:nowcoder_training:2020_multi-university_training_contest_9 

Last update: **2020/08/14 11:20**