

2022-11-14总结

T1

字符串题，没有什么思路，考场上写了一个全排列，枚举每一种排序方式，求出长度最小的并且字典序最小的字符串。

竟然拿了50分！！

总结一下，最近几天把字符串的算法再看一遍，复习一下。

T2

这道题看起来很简单，但是当我看到数据范围的时候就被吓到了，这道题的关键是每次询问的 k 不相同，我考试的时候特意去翻了一下大样例的输入，发现里面相同的 k 平均只会出现1~2次，这就不好搞了啊，反正最后到交卷我也只写了一个大暴力。最后只拿到了20分。

T3

这道题一看就很 dp ，我想了半天，发现二维 dp 根本做不出来，然后又去想三维的 dp ，然后又发现有点难实现，彻底放弃。

正解

好吧，我承认这道题确实有点难想。这道题并不是一个很 dp 的 dp 题，这个 dp 超级奇怪，先放上代码

```
sort(node+1, node+1+n, cmp);
for(int i=1; i<=n; i++){
    dp[i][0] = dp[i][1] = 1;
    for(int j=i-1; j>=1; j--){
        if(node[j].y < node[i].y){
            dp[i][0] = (dp[i][0] + dp[j][1])%mod;
        }
        else{
            dp[j][1] = (dp[j][1] + dp[i][0])%mod;
        }
    }
}
for(int i=1; i<=n; i++){
    ans = (ans + dp[i][0] + dp[i][1])%mod;
}
write((ans - n + mod)%mod);
```

这里是核心代码。首先，考场上我想到的是将所有的点按照纵坐标来排序，但是正解却是按照横坐标来排序的，看起来很不能理解。其实正解判断的是两种情况。即每个点都存在两种情况，一种是从这个点向右下连，另一种是向左下连。

首先从横坐标最小的向右遍历，每次遍历到一个点，找他前面的所有点，如果前面的这个点的纵坐标比当前点更高，那么只会从这个前面的点向右下连到当前枚举的点。另一种情况是当前点向左下连到向前找的点。

所以说在整个过程中，每个点的答案都会有可能变化，所以只能最后再统计答案。最后输出的时候因为开始定义的每个点都多了1，所以最后要减去1。

T4

这是一道结论题，考场上没有推出来，然后当时没有太多时间了，就直接输出了3，以为可以骗点分，然而只有6分，其实这道题可以骗很多分的，有各种各样的骗分思路，比如说我们直接统计有多少个黑色块，然后输出，这就有40多分。总结教训就是能骗分的不要放过。

总结

考下来接近90分，但看到很多同学都有 100^+ ，有点低，以后考试的时候，如果没有稳妥的办法，能骗分的一定不要放过，然后最近要把字符串的算法好好复习一下（或者是重新学一遍，感觉什么都不会）。