

Shiki 题解

小写英文字母可以被映射到0-26之间的整数。所以我们定义矩阵 $a[i][j]$ 表示 i, j 字母形成的二元组产生的收益。

此外定义 $dp[i][j]$ 表示 $[i, j]$ 之间的子串产生的最大收益值。

如果第 i 个字母为 j , 那么 $dp[i][j] = \max_{0 \leq k \leq 26} \{dp[i-1][k] + a[k][j]\}$

反之, $dp[i][j] = dp[i-1][j]$

pokemon 题解

容易发现, 从任意一点出发, 总是靠着右手边墙壁走总能绕地图走一圈回到起点。这是因为, 两点间只存在一条简单路径这一性质保证了图中可以走的路径形成了一棵树。而走的过程就是在一棵规定了儿子顺序的树上进行欧拉序遍历的过程。

因此我们直接沿任意点出发, 求出上面的这个环, 并记录下经过每个点在环上的位置, 对询问的两个点的位置作差即可 $O(1)$ 地回答询问。因此这个题的复杂度为最差 $O(nm)$ 。

需要注意的细节:

- 如果你用 map 存储三元组来获得每个点的位置, 会有较大常数, 可能会被卡。标程使用了将一个三元组 (x, y, dir) 映射到一个正整数的做法, 从而可以开一个一维数组存储。你也可以只将坐标映射到整数, 用二维数组存储;
- 起点和终点可能相同, 这个时候需要特判一下输出 0;
- 如果直接把地图的两个维度拉到最大值 10^5 会爆空间, 所以可以用 STL 的动态容器, 或者把二维坐标映射到一个数字, 开一位数组存储地图。

另外, 直接把可以走的路径建成树再遍历也是一种写法。

matrix 题解

考虑公比为整数的等比数列, 长度最多为 $\log w$ 级别, 因此我们每次清空周围 $2\log w * 2\log w$ 的区域搜索一下即可。无解的情况为有连着的相同数字(来回走)

实现时需要注意一些细节

multiplication 题解

对 n 进行奇偶分治。我们将 n 以内奇数提出来求自然溢出意义下的积, 对于偶数我们全部除以2, 将除了多少2记录下来, 然后我们发现我们需要计算1至 $\lfloor n/2 \rfloor$ 的乘积, 递归做即可, 最后我们把除的2乘回去, 因为要去掉末尾0, 因此要将2的个数模4再乘回去就得到了答案

然后我们考虑1至 n 的奇数之积怎么算, 即计算 $\prod_{i=1}^{(n-1)/2} (2i+1)$

我们可以把括号拆开, 因为是模自然溢出意义下的, 因此对于每个括号中, $2i$ 至多选63次, 否则一定无贡献。

于是问题就变成了: 从1至 n 的数中选择 m 个数相乘, 对于所有选择方案求和

容易发现这就是第一类斯特林数 $\left[\begin{matrix} n+1 \\ n-m+1 \end{matrix} \right]$

根据第一类斯特林数的定义，我们需要选出 $n - m + 1$ 个环排列，但我们发现环长大于1的环排列个数非常少，于是我们设 $dp[i][j]$ 表示 i 个点排除 j 个环长大于2的环排列的方案数。最后对于每一个 $dp[i][j]$ 我们从 n 个点中选出 i 个数放入这些环排列，组合数算一下就行

复杂度看实现，因为 m 很小，较高的复杂度也是可以接受的