|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 试题编号 | 名称 | 题目大意 | 算法讨论 | 时空复杂度 |
| 2003E | Covering Whole Holes | 判一个直角多边形（盖子）是否可以通过平移完全覆盖另一个直角多边形（洞）。 | 盖子覆盖住洞后，通过微调一定会有两条边重合。枚举他们各自一条水平的线段假设他们重合，再维护盖子相对于洞水平移动哪些距离是可行的，这可以用一些区间的集合来表示。然后从上到下扫下来，进行一些区间的操作即可。 | 时间：  空间： |
| 2008D | The Hare and the Hounds | 猎狗和兔子在进行公路拉力赛。参赛者（猎狗）必须找到组织者（兔子）选择的一条或多条路。他们在路上走的时候会遇到各种各样的十字路口。进入任何十字路口时（除了一些特殊的，之后会讲到），他们都应该使用主要的道路规则。主要的道路规则总是“尽量直”，意思是转最小的角度（可能不需要转弯）。如果两条路转的角度相同（例如 'Y'字路口），这时主要的道路规则要求选择右边那条。  在一些路口，兔子可能违反的主要道路规则（但不会去来的那条路），随机选一条路走，这些十字路口被兔子标为选择点（通常在十字路口的路面上放一个彩色标记）。猎狗遇到选择点后，必须尝试每一条道路，直到到达一个确认标记（通常是兔子甩了一些面粉在路上）。下列都表示一个不正确的路线选择：  1.猎狗走了maxdis后还没有碰到确认点。  2.猎狗在碰到确认点前遇到了死胡同（只有一条路的十字路口）  3.猎狗在碰到确认点前遇到了其他的选择点（兔子总是会在到达下一个选择点的路上放确认点，且兔子不会再回到之前那个选择点）  在遇到一个不正确的路线后，猎狗必须沿原路返回，然后选其他路，即使中间他到达了终点，他也会忽略。  当猎狗在选择点时，他使用主要道路规则的方式略有不同，第一道路的选择是与原来一样的。但是如果猎狗必须返回这个选择点时（例如遇到一个不正确的路线），猎狗会从回来的方向使用主要道路规则选路（忽略所有他已经尝试过的路与一开始来这个点时走的路），这个过程会不断的循环知道他找到了正确的路。这种不同的主要道路规则的方式只会用在选择点上，对于本问题，猎狗不会记得它走向一条路的结果，即使他在探索选择点的时候不断的经过这条路。  对于这个问题，你会得到一个路线图(一个配置的道路和路口)，选择点的列表，确认点的位置，maxdis的值，起始点与终点（不是选择点），离开起始点时的方向。有了这些信息，你就可以模拟猎狗追兔子时的路线，你可以认为猎狗使用这些策略总可以找到兔子的路线。 | 纯模拟，注意起点=终点时的情况。 | 时间：  空间： |
| 2013H | Matryoshka | 有一行n个单层套娃，大小为ai，每个套娃可以直接放入更大的单层套娃中。每次操作可以合并相邻两个套娃，合并时打开并关闭1个套娃的代价为1，套娃合并后不可拆分。问把这些单层套娃合并成若干个完整套娃（大小为从1开始的连续正整数，层数若干）需要的最少代价。 | 两个套娃合并，设两套中分别的最小值中较大的值为x，则代价为总数量-另一套中大小<x的套娃数量。这个通过预处理一段中的最小值，以及前i个套娃中大小<=j的数量后可O(1)算出。用f[i][j]表示[i,j]段套娃合并所需的最小代价，转移为f[i][j]=min{f[i][k]+f[k+1][j]+cost(i,k,j)}。最后把完整的套娃再DP一遍即可。 | 时间：  空间： |