

深度优先搜索

基本模型：给定起点和终点，寻找可通行路线，如下图所示@代表起点，*代表终点，求一共需要多少步才能从起点走到终点。

	1	2	3	4	5	6
1	#	.	.	.	#	#
2	#	.	@	.	.	#
3	.	#
4	.	#	.	.	#	.
5	.	.	#	.	*	#

数据准备：

- 1、建立Node类型结构体，用来存储每一个节点的信息，用于解答题目
- 2、建立char类型地图数据表map[行][列]，记录整个地图信息
- 3、建立bool类型访问记录表vis[行][列]，记录某个节点是否已经走过

遍历逻辑：

- 1、将起点记录已访问，并按上左下右的方向搜寻其他的邻格。

地图表							访问记录表						
	1	2	3	4	5	6		1	2	3	4	5	6
1	#	.	.	.	#	#	1						
2	#	.	@	.	.	#	2						
3	.	#	3						
4	.	#	.	.	#	.	4						
5	.	.	#	.	*	#	5						

- 2、首先是起点上方的格子(1,3)点，它可以通行，且不是终点，我们需要将其标记为已访问。

地图表							访问记录表						
	1	2	3	4	5	6		1	2	3	4	5	6
1	#	.		.	#	#	1						
2	#	.	@	.	.	#	2						
3	.	#	3						
4	.	#	.	.	#	.	4						
5	.	.	#	.	*	#	5						

- 3、接下来，我们以(1,3)点作为出发点，继续搜索它上方的(1,3)点，很明显，这个点超出范围了。

地图表							访问记录表						
	1	2	3	4	5	6		1	2	3	4	5	6
1	#	.		.	#	#	1						
2	#	.	@	.	.	#	2						
3	.	#	3						
4	.	#	.	.	#	.	4						
5	.	.	#	.	*	#	5						

- 4、现在，我们搜索(1,3)点左侧的点，它可以通行！同样的，将其加入访问记录表。

	1	2	3	4	5	6
1	#	■	■	.	#	#
2	#	.	@	.	.	#
3	.	#
4	.	#	.	.	#	.
5	.	.	#	.	*	#

	1	2	3	4	5	6
1		■	■			
2			■			
3						
4						
5						

5、继续向下搜索，以(1,2)点作为起始，可以发现它的上左都没有办法通行，唯一可走的就是下边(2,2)点。

	1	2	3	4	5	6
1	#	■	■	.	#	#
2	#	■	@	.	.	#
3	.	#
4	.	#	.	.	#	.
5	.	.	#	.	*	#

	1	2	3	4	5	6
1		■	■			
2		■	■			
3						
4						
5						

6、以(2,2)点为起始时，我们已经无路可走了。

	1	2	3	4	5	6
1	#	■	■	.	#	#
2	#	■	@	.	.	#
3	.	#
4	.	#	.	.	#	.
5	.	.	#	.	*	#

7、此时，我们只能退回到(1,2)点。

	1	2	3	4	5	6
1	#	■	■	.	#	#
2	#	■	@	.	.	#
3	.	#
4	.	#	.	.	#	.
5	.	.	#	.	*	#

8、很明显(1,2)点也不存在可行走路径，我们只能继续向后退至(1,3)点。此时，以(1,3)点作为起点时，右侧的(1,4)点可以通行！我们将其加入访问列表中。

	1	2	3	4	5	6
1	#	■	■	■	#	#
2	#	■	@	.	.	#
3	.	#
4	.	#	.	.	#	.
5	.	.	#	.	*	#

	1	2	3	4	5	6
1		■	■	■		
2		■	■			
3						
4						
5						

9、现在，我们以(1,4)点作为起点，很明显它的上左邻格都无法通行，唯一可以通行的就是下方的(2,4)点。

	1	2	3	4	5	6
1	#	■	■	■	#	#
2	#	■	@	.	.	#
3	.	#
4	.	#	.	.	#	.
5	.	.	#	.	*	#

	1	2	3	4	5	6
1		■	■	■		
2		■	■	■		
3						
4						
5						

1	#	.	.	■	#	#
2	#	.	@	■	.	#
3	.	#
4	.	#	.	.	#	.
5	.	.	#	.	*	#

1		■	■	■		
2		■	■	■	■	
3						
4						
5						

10、按照这样的方法，我们不断的遍历每一个节点的上左下右，并将已经走过的进行标记，直到最后走到终点。

地图表

	1	2	3	4	5	6
1	#	.	.	.	#	#
2	#	.	@	■	.	#
3	.	#	.	■	.	.
4	.	#	.	.	#	.
5	.	.	#	.	*	#

访问记录表

	1	2	3	4	5	6
1		■	■	■		
2		■	■	■		
3				■		
4						
5						

地图表

	1	2	3	4	5	6
1	#	.	.	.	#	#
2	#	.	@	■	.	#
3	.	#	■	■	.	.
4	.	#	.	.	#	.
5	.	.	#	.	*	#

访问记录表

	1	2	3	4	5	6
1		■	■	■		
2		■	■	■		
3			■	■		
4						
5						

地图表

	1	2	3	4	5	6
1	#	.	.	.	#	#
2	#	.	@	■	.	#
3	.	#	■	■	.	.
4	.	#	■	.	#	.
5	.	.	#	.	*	#

访问记录表

	1	2	3	4	5	6
1		■	■	■		
2		■	■	■		
3			■	■		
4			■			
5						

地图表

	1	2	3	4	5	6
1	#	.	.	.	#	#
2	#	.	@	■	.	#
3	.	#	■	■	.	.
4	.	#	■	■	#	.
5	.	.	#	.	*	#

访问记录表

	1	2	3	4	5	6
1		■	■	■		
2		■	■	■		
3			■	■		
4			■	■		
5						

地图表

	1	2	3	4	5	6
1	#	.	.	.	#	#
2	#	.	@	■	.	#
3	.	#	■	■	.	.
4	.	#	■	■	#	.
5	.	.	#	■	*	#

访问记录表

	1	2	3	4	5	6
1		■	■	■		
2		■	■	■		
3			■	■		
4			■	■		
5				■		

地图表

	1	2	3	4	5	6
1	#	.	.	.	#	#
2	#	.	@	■	.	#
3	.	#	■	■	.	.
4	.	#	■	■	#	.
5	.	.	#	■	*	#

访问记录表

	1	2	3	4	5	6
1		■	■	■		
2		■	■	■		
3			■	■		
4			■	■		
5				■	■	

使用深度优先搜索的关键词一般包含：“求总方案数”、“列出所有方案”等，这种就要求我们穷尽所有情况。

逻辑航线培优教育，信息学奥赛培训专家。

扫码添加作者获取更多内容。

