# 勇闯迷宫游戏

## 一.项目背景：

迷宫只有两个门，一个门叫入口，另一个门叫出口。一个骑士骑马从入口进入迷宫，迷宫设置很多障碍，骑士需要在迷宫中寻找通路以到达出口。

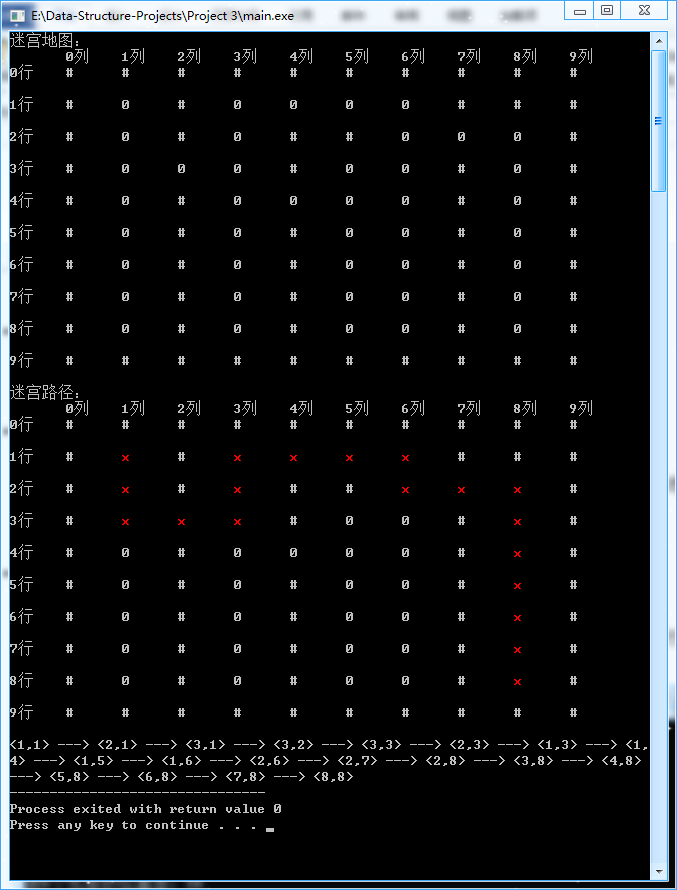
## 二.项目功能

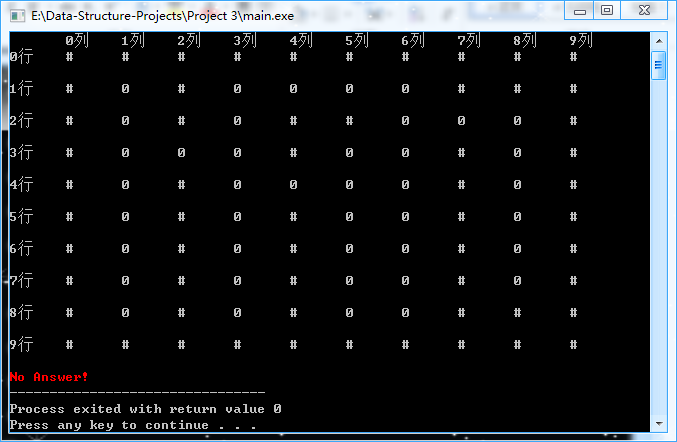
根据地图自动输出路径

说明：

起点默认为(1,1)，终点默认为(8,8)

迷宫地图按照老师要求是内置的，如需更改请进入源代码进行修改





## 三.程序描述

迷宫问题的求解过程可以采用回溯法即在一定的约束条件下试探地搜索前进，若前进中受阻，则及时回头纠正错误另择通路继续搜索的方法。从入口出发，按某一方向向前探索，若能走通，即某处可达，则到达新点，否则探索下一个方向；若所有的方向均没有通路，则沿原路返回前一点，换下一个方向再继续试探，直到所有可能的道路都探索到，或找到一条通路，或无路可走又返回入口点。在求解过程中，为了保证在达到某一个点后不能向前继续行走时，能正确返回前一个以便从下一个方向向前试探，则需要在试探过程中保存所能够达到的每个点的下标以及该点前进的方向，当找到出口时试探过程就结束了。

一般来说有BFS和DFS可以用来求解该问题，但是根据该项目的要求，并没有要求是最优值，故我选择了DFS进行搜索，只要找到路径即可，核心搜索代码如下：

bool dfs(int x,int y)

{

//如果找到了终点，就返回True

if ((x==goal.x)&&(y==goal.y))

{

map[x][y]='x';

return true;

}

for (int i=0;i<4;i++)

if (map[x+dir[i].x][y+dir[i].y]=='0')//如果下一个位置可以放

{

//在当前位置做个标记

map[x][y]='x';

//进入递归搜索

if (dfs(x+dir[i].x,y+dir[i].y))

return true;

//撤回标记

map[x][y]='0';

}

return false;

}

代码很短，很容易理解的