

# 实验 2-3: 迷宫求解（要求实现随机迷宫、最短路径的提取）

---

## 1. 问题描述

有一个 $m \times n$ 格的迷宫(表示有 $m$ 行、 $n$ 列)，其中有可走的点也有不可走的点，我们用1表示可以走，0表示不可以走。现在要你编程找出最短的道路，要求所走的路中没有重复的点，走时只能沿着上下左右四个方向。如果没有道路，则输出-1。

**输入要求:** 第一行是两个数 $m, n$  ( $1 < m, n < 100$ ) 接下来是 $m$ 行 $n$ 列由1和0组成的数据 最后两行是起始点和结束点的坐标，输入坐标时先行后列。 **输出要求:** 一条最短的路径，描述一个点时用 $(x, y)$ 的形式，除开始点外，其他的都要用“->”表示方向。如果没有一条可行的路则输出-1。

## 2. 算法的描述

### 2.1 数据结构的描述

本程序使用了 双向广搜, 即从头和尾同时进行搜索，从而提高了搜索的效率。程序中的 `aflag` 和 `bflag` 分别标记了头和尾经过的路线，当头经过的点出现在了尾的路径中，或尾经过的点出现在头的路径中，则停止，并输出路径。实现上述操作的代码为：

```
if (bflag[(*a).x][(a).y] != NULL) {
    answer1(tmp);
    answerb1((bflag[(a).x][(a).y]).dad);
    isfound = 1;
    break;
}
```

### 2.2 程序结构的描述

程序中的 `main()` 函数接受外界输入的迷宫，并通过双向广搜得到结果，分别通过 `answer1()` 和 `answerb1()` 输出从头到相同点、从相同点到尾的路径。

## 3. 算法的时间分析

算法的最坏时间复杂度为  $O(M \times N)$ ，算法的空间复杂度为  $O(M \times N)$