迷宫问题及搜索方法的研究

金加宝 10175101146

目录

**[一. mediumMaze路径搜索](#_Toc5497_WPSOffice_Level1)** **[2](#_Toc5497_WPSOffice_Level1)**

[1. 深度优先搜索](#_Toc967_WPSOffice_Level2) [2](#_Toc967_WPSOffice_Level2)

[2. 广度优先搜索](#_Toc11211_WPSOffice_Level2) [3](#_Toc11211_WPSOffice_Level2)

[3. 贪心最好优先搜索](#_Toc27510_WPSOffice_Level2) [5](#_Toc27510_WPSOffice_Level2)

[4. A\*搜索](#_Toc23233_WPSOffice_Level2) [5](#_Toc23233_WPSOffice_Level2)

**[二. openMaze路径搜索](#_Toc967_WPSOffice_Level1)** **[6](#_Toc967_WPSOffice_Level1)**

[1. 深度优先搜索](#_Toc16811_WPSOffice_Level2) [6](#_Toc16811_WPSOffice_Level2)

[2. 广度优先搜索](#_Toc28540_WPSOffice_Level2) [8](#_Toc28540_WPSOffice_Level2)

[3. 贪心最好优先搜索](#_Toc31631_WPSOffice_Level2) [8](#_Toc31631_WPSOffice_Level2)

[4. A\*搜索](#_Toc7985_WPSOffice_Level2) [9](#_Toc7985_WPSOffice_Level2)

**[三. 其他算法](#_Toc11211_WPSOffice_Level1)** **[10](#_Toc11211_WPSOffice_Level1)**

[1. close\_a方法](#_Toc876_WPSOffice_Level2) [10](#_Toc876_WPSOffice_Level2)

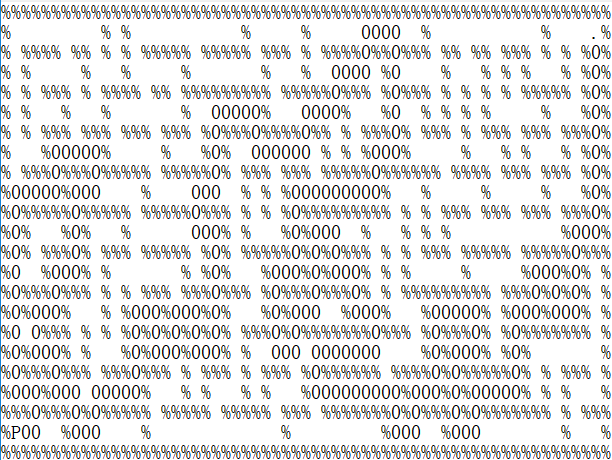
[2. A\*加权算法](#_Toc13637_WPSOffice_Level2) [10](#_Toc13637_WPSOffice_Level2)

[3. mul方法](#_Toc24653_WPSOffice_Level2) [10](#_Toc24653_WPSOffice_Level2)

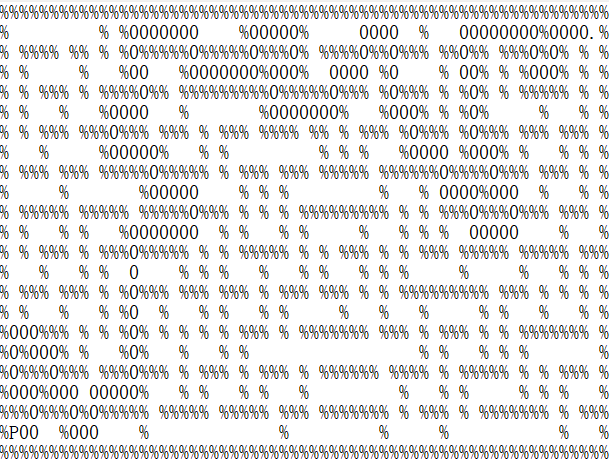
**[四. 总结](#_Toc27510_WPSOffice_Level1)** **[10](#_Toc27510_WPSOffice_Level1)**

1. mediumMaze路径搜索
2. 深度优先搜索
3. 启发函数：无
4. 从起始位置到终止位置的路径

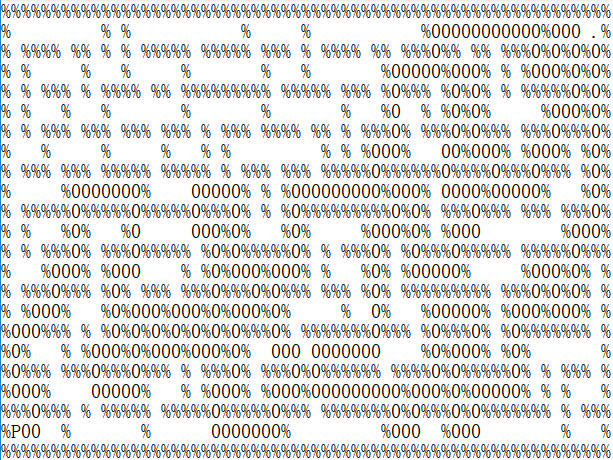
方案一：



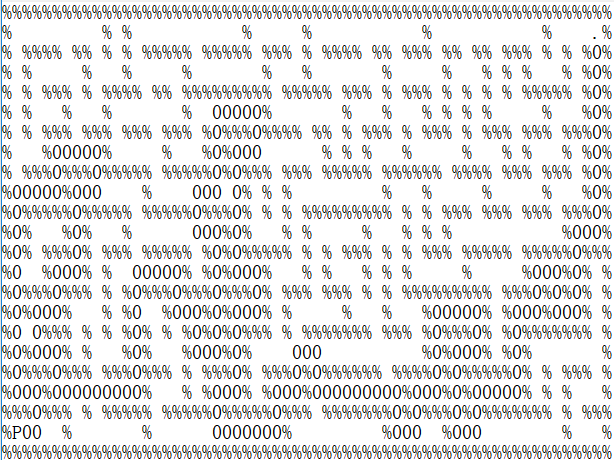
方案二：



方案三：



方案四：



1. 步骤数

方案一：246

方案二：160

方案三：300

方案四：202

1. 搜索结点数

方案一：72

方案二：65

方案三：96

方案四：51

1. 运行时间

方案一：0.0059833526611328125

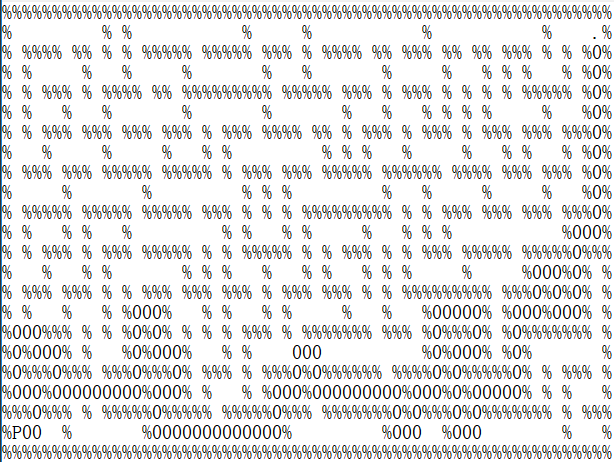
方案二：0.0029935836791992188

方案三：0.0169539451599121145

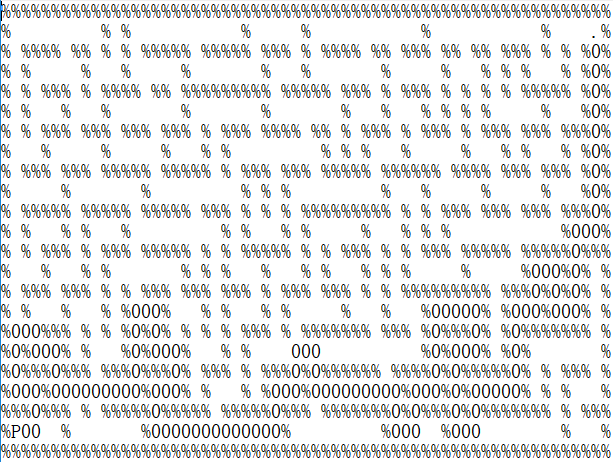
方案四：0.0026926300048828125

1. 广度优先搜索
2. 启发函数：无
3. 从起始位置到终止位置的路径

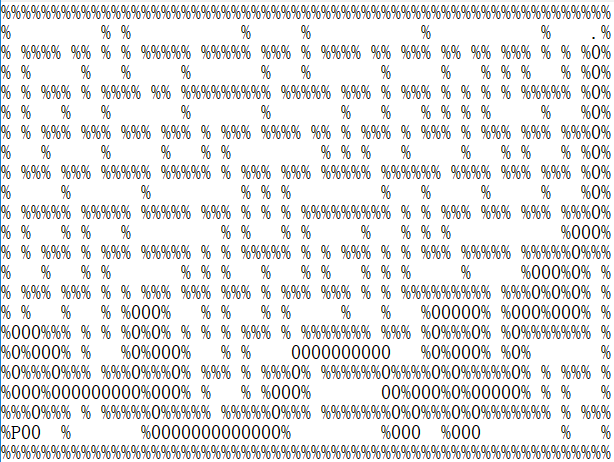
方案一：



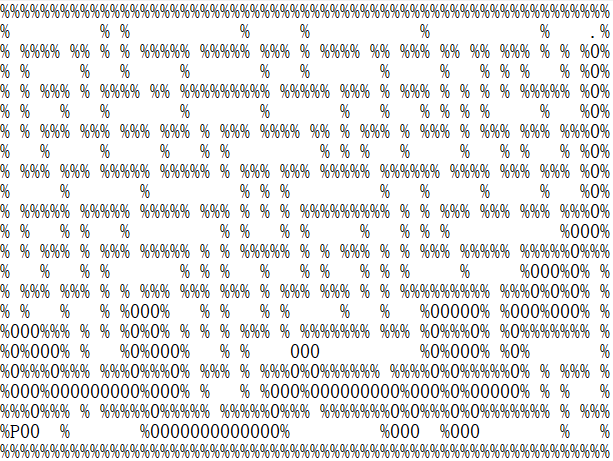
方案二：



方案三：



方案四：



1. 步骤数

方案一：132

方案二：132

方案三：132

方案四：132

1. 搜索结点数

方案一：133

方案二：132

方案三：121

方案四：126

1. 运行时间

方案一：0.003989219665527344

方案二：0.003987789154052734

方案三：0.008975744247436523

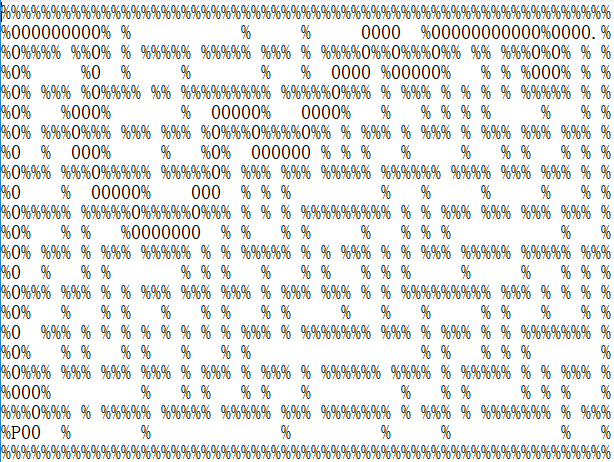
方案四：0.0026926300048828125

1. 贪心最好优先搜索
2. 启发函数：

H：当前位置到终点的曼哈顿距离

G：0

1. 从起始位置到终止位置的路径

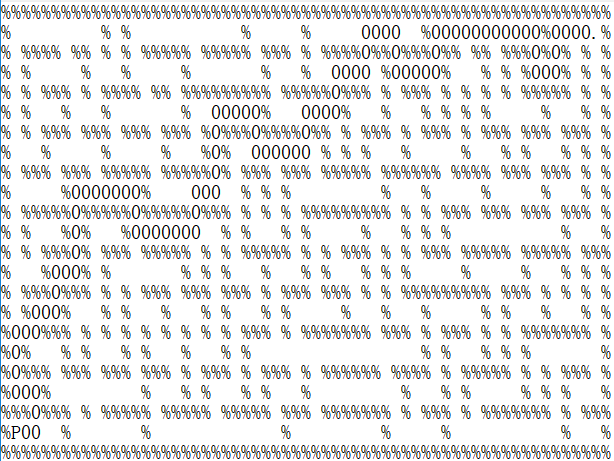


1. 步骤数：118
2. 搜索结点数：36
3. 运行时间：0.001994609832763672
4. A\*搜索
5. 启发函数：

H：当前位置到终点的曼哈顿距离

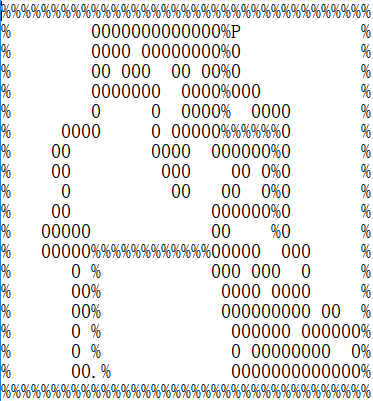
G：当前位置到起点的曼哈顿距离

1. 从起始位置到终止位置的路径

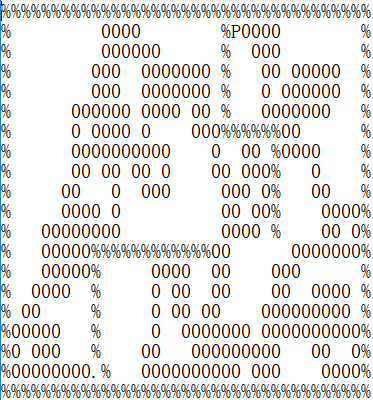


1. 步骤数：98
2. 搜索结点数：96
3. 运行时间：0.0029921531677246094
4. openMaze路径搜索
5. 深度优先搜索
6. 启发函数：无
7. 从起始位置到终止位置的路径

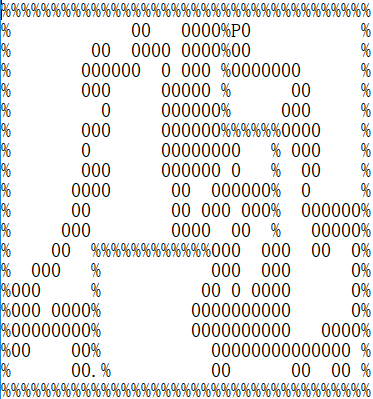
方案一：



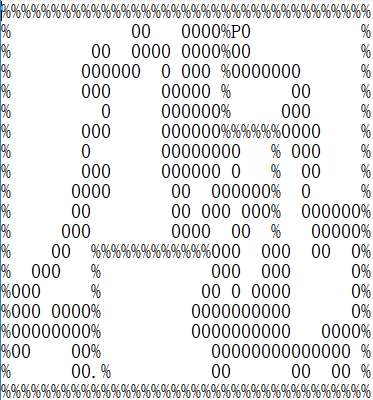
方案二：



方案三：



方案四：



1. 步骤数

方案一：201

方案二：277

方案三：235

方案四：211

1. 搜索结点数

方案一：428

方案二：420

方案三：356

方案四：287

1. 运行时间

方案一：0.022940397262573242

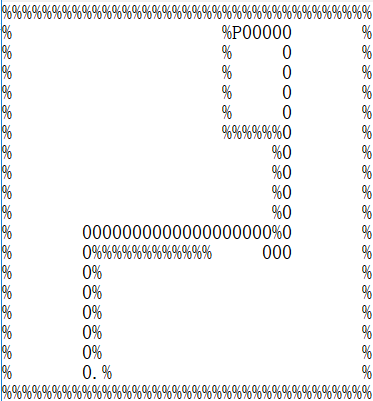
方案二：0.00698089599609375

方案三：0.005956888198852539

方案四：0.005983591079711914

1. 广度优先搜索
2. 启发函数：无
3. 从起始位置到终止位置的路径

方案一、二、三、四：



1. 步骤数

方案一、二、三、四：45

1. 搜索结点数

方案一：566

方案二：613

方案三：600

方案四：588

1. 运行时间

方案一：0.003989458084106445

方案二：0.003989458084106445

方案三：0.003991127014160156

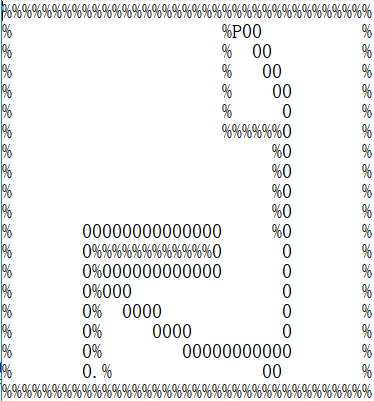
方案四：0.004995107650756836

1. 贪心最好优先搜索
2. 启发函数：

H：当前位置到终点的曼哈顿距离

G：0

1. 从起始位置到终止位置的路径

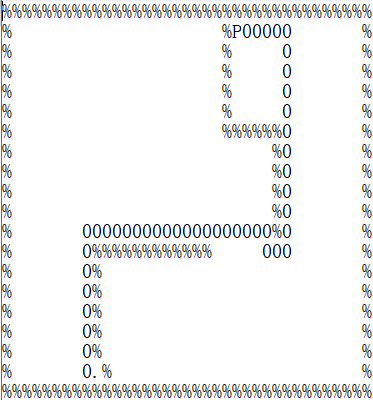


1. 步骤数：79
2. 搜索结点数：212
3. 运行时间：0.003988981246948242
4. A\*搜索
5. 启发函数：

H：当前位置到终点的曼哈顿距离

G：当前位置到起点的曼哈顿距离

1. 从起始位置到终止位置的路径



1. 步骤数：45
2. 搜索结点数：346
3. 运行时间：0.007974386215209961
4. 其他算法

另外，我还尝试了其他启发函数，都取得了不错的结果，具体见results文件夹。

1. close\_a方法

启发函数

H：当前位置到终点的曼哈顿距离

G：当前位置到起点的曼哈顿距离

F: 

1. A\*加权算法

启发函数

H：当前位置到终点的曼哈顿距离

G：当前位置到起点的曼哈顿距离

F：

1. mul方法

启发函数

H：当前位置到终点的曼哈顿距离

G：当前位置到起点的曼哈顿距离

F：

1. 总结

实验结果表明，无信息的搜索结果的表现往往没有带有信息的搜索的好。同时，对于带信息的A\*算法而言，启发函数的选择异常重要。一个好的启发函数能够很大程度上的加快搜索效率。

加权的A\*算法的表现往往好于经典的A\*算法，但是如何选取权值是一个较难解决的问题。而另一方面，经过试验，close\_a的结果出乎意料的好，不仅能够很大程度上超越A\*算法，而且能与最好的加权A\*方法想媲美，却不需要反反复复寻找最优权值。