

## 一. 编译运行环境

编译环境 Microsoft Visual Studio Community 2017 版本 15.8.4

Windows 10 家庭中文版 版本号 1803 操作系统版本 17134.472 64 位

## 二. 关键函数说明

整个程序由 BigWork, BmpOp, IdaStar, GenerSolve 四个 cpp 文件组成。

1.BigWork 为主程序, 实现了拼图的大部分主要功能。

Printinfo 打印游戏介绍。IniPuzzle 用于生成拼图, PrintPuzzle 用于打印拼图。CheckState 检测游戏是否结束。

Solvable 判断生成的拼图的可解性。计算 puzzle 的逆序数, 在  $n$  为奇数或偶数时分别用对应的数学方法, 与目标状态间进行判断。需要注意的是  $m$  或  $n$  为 1 时空白块只能左右移动, 此时需要单独进行判断。

LoadFile 和 ReadFile 函数分别用于存档与读档。读档时将拼图的初状态一并更新, 并重新读入图片数据防止重新切割后颜色错位。

CheckInput 函数整合了各个模块, 并集成了重新初始化的功能, 用于检测用户输入, 调用相关模块。

2.BmpOp 实现了 bmp 图片的读取, 切割和保存。

包含四个函数。Readbmp 根据用户输入的文件名打开相应的 bmp 图片。

Savebmp 函数将读取的图片根据当前的 puzzle 状态切割, 涂空白后保存。

Freebmp 用于在每一轮游戏结束时释放储存 bmp 的数组的内存, 防止内存泄漏。

PrintEgg 是程序的一个彩蛋, 将在附加功能中加以说明。

3.IdaStar 使用启发式算法+DFS, 用于在拼图的规模较小时求取最优解。

启发式函数  $f = \text{搜索深度} + \text{曼哈顿距离}$

Manha\_dis 计算当前状态与目标状态间的曼哈顿距离。

IdaStar 使用递归实现深度优先查找, 结合判断启发式函数的变化减少状态空间的数量。

Init\_IdaStar 为主函数调用搜索求解模块提供接口。

aStar 算法在  $m*n < 16$  时表现良好, 且能够确保得到拼图的最优解, 但是在  $m*n$  较大时耗时太长, 影响游戏性, 所以在只用来求解  $m*n < 16$  的情况。

```
已生成拼图:
7      0      -1
2      6      3
5      4      1
*****
该拼图可解, 请开始游戏
F
使用启发式算法, 确保得到最优解, 进行自动求解中:
S A A W D S D S A A W D D W A S D S A A W D D S
求解结束, 共用了24步! 用时3.00 ms!
```

4. GenerSolve 模拟人求解拼图的思路，用于在拼图规模较大时给出一个可行解。

算法大体实现思路如下：

将整个 puzzle 分成三个部分分步进行还原。

先还原前  $n-2$  行，思路是将每一行的数字从左到右一个一个交换到对应的位置上，每行最后两个数字的还原方法略有不同，调用 SpecCase4 函数进行直接操作。

再还原最后两行的前  $n-2$  列，思路与第一步类似，将每一列的两个数字在他们所属行的右侧连到一起之后，调用 SpecCase5 函数将其归位。

最终还原 puzzle 右下角的三个数字，由于主程序已经保证了 puzzle 的可解性，在此只要将三个数字中的一个数字归位，再将空白块移到右下角即可完成拼图的还原。

TryMov 函数用于判断空白块移动的合法性，并根据曼哈顿距离找出距离目标位置较近的移动方向，MoveBlank 通过调用 TryMov 函数将空白块移动到拼图的某一位置。MoveTarget 通过调用前述函数，用于将特定拼图块归位。

MoveRow, MoveCol, MoveFin 三个函数分别对应还原拼图的三个步骤。Ini\_GenerSolve 为主程序提供接口调用求解模块。

在具体还原的过程中有可能遇到一些特殊情况，此时调用 SpecCase 函数手动解决。

在  $m \times n > 16$  时本程序自动求解模块将调用生成算法，能在 0.1s 时间内给出  $100 \times 100$  拼图的一个可行解。为了防止程序崩溃，也考虑到拼图的可玩性， $m \times n > 100$  时将不能调用自动求解模块（实际上还是可以求解，但是步数往往已经过万，失去了游戏本身的价值）。

```
已生成拼图:
70      30      32      64      25      77      8       58      60
19      44      28      62      33      59      2       71      3
16      13      69      14      78      7       41      68      43
39      4       0       49      17      9       67      15      23
26      65      48      21      31      76      40      66      74
1       47      18      61      38      -1      36      55      56
24      37      63      6       22      35      54      20      57
51      29      11      45      53      12      42      50      27
73      46      75      5       10      79      34      72      52
*****
该拼图可解, 请开始游戏
F
使用生成算法进行自动求解中:
W W A A W A A S D W A A S D W A S D W W A S D W W A S D S S S S A W D
W A A S D W A A S D W A S A S S A W D D S A W D D S A W D S D W W A
S S S A W D D S A W D D S A W D S D W W A S D W W A S D W W A S D W W
A S D W A A S D W A A S D W A S D W W A S D W W A S D W W A S D W W
A S D W W A S D W W A S D S S S S S D W A S D W W A S D W W A S D W
D W W A S D W W A S D S S A W D S D W W A S D W W A S A A S S A W D
S D W W A S D W W A S D W W A S D W D D S A A W D D D S S S S D W A
S A W D D S A W D S D W W A S D W W A S D W W A S D W D S A A A A S
A A A S S S A W D D S A W D D S A W D D S A W D D S A W D D S A W D
S D W A A S D W A S D W W A S D W W A S D W W A S D S D W W A S D S
S D W A S D W W A S D W W A S A S D D W A S D D W W A S D W D D S A A
A W A S A A A S A A W D S A A W D S A S D D W W A S D D W D S S S A
S A W D D S A W D D S A W D D S A W D D S A W D S A W W D S A W A W D
W D D S A W D D S A W D S D W W A S D D S A W D S D W W A S A A S S
D D S A W D D S A W D D S A W D D S A W D D S A W D D S A W D S A W A
W A A S D W A S D D D D D D S D W A A S D W A A S D W A A S D W A A
S D W D D S A A W D S S D W A S D W W A S A A A A A A S D D W A S D D
W D D S A W A S A A A A A S D W A A S D D D W A A S D W A A S D D D
S A W A S D D D D W A A S D W A A S D W A S D W D D D S A A W D S A
D D W A A S D D W A A S D D W A A S D W A S D D W A S D W A A S D W
求解共计1832步, 用时3.00 ms!
```

### 三、附加功能说明

在 BmpOp 中包含了一个彩蛋函数 PrintEgg, 如果检测到用户还原拼图的时间短于五秒, 主程序将调用此彩蛋函数, 打印一个用 Ascii 码拼成的图案, 对用户可能的作弊行为的表示滑稽。

[illegible]