您好，欢迎使用本拼图游戏！

实验报告

本次大作业包含两部分

作业总体思路

比较模式化的将空白格移动到目标格的上左右三个固定的方向，然后分类讨论。

写基本函数，每个基本函数包含一定固定的操作数，每个基本函数操作后不影响已排好的方格的顺序然后遍历

编译环境

TDM-GCC 4.9.2 64-bit Release

本次大作业包含两小部分，第一部分是最开始写的目标是解决大规模的数组，但是后来发现容易产生越界的问题，在实验中实验测试数据，但后来发现在测试的过程中有时会出现空格越界，造成将空白格定位的函数无法正常运行。后来对小规模的，进行了改进，使用广度优先线搜索。

各个功能的实现

可行性分析通过逆序数分析

函数

Bool Solvable（）

移动

分为两个步骤，第一步是将空白格定位，第二步是将空白格与目标方向的小方块交换

关键函数

Int BlankX()

Int BlankY()

用于定位空格的横纵坐标

空格定向移动函数

blanket，blanket，blanket，left

Int TargetX(int id)

Int TargetY(int id)

用于遍历0-m\*n-2时，将各个待移动的方格(编号ID)定位

输入目标高ID的名称然后这个程序构造一些基本的结构，比如说我可以把它迁移到目标格的右边，然后之后呢我有三个基本函数，move，up，move，left，move right含有五个基本操作，比如说move up的基本操作就是上左下右下上，左下右上，这样的话就可以实现将目标格向上移动一个，然后呢，他也不会影响其他的方块结构

显示状态函数Print

自动打印状态

Void left()

Void right()

Void up()

Void down()

重新游戏指令R通过一个primary数组，将游戏开始时的状态存储输入后，primary覆盖现有数组，并且输出。

存档功能的实现

通过一个三维数组record其中两位用于保存棋盘的布局，第3维为用于存档记录名称。

考虑到实际情况，一般的使用者不会连续存档超过50次，本程序将档案容量设置为50。

但要读取档案，输入进度名称。子函数遍历存档数组record，并且返回档案编号；之后再将档案覆盖当前数组，实现存档，和读档的功能。

若要游戏结束，输入“P”

以while循环实现

结束后增加了附加功能，输出“再见！“