1. **Design**
2. Thoughts

Some key accesses to output a disordered matrix are as follows.

1. *Create a one-dimensional sequential list matrix[ ].* All elements are arranged in numerical order. In the list, the blank space is temporarily replaced by “0”. The length of the sequential list is decided by the input of user through mode=input (" Please enter the mode you want (8/15):"). If the mode is 8, then the standard matrix is *matrix=[0,1,2,3,4,5,6,7,8]*. Similarly, if the mode is 15, the standard matrix is *matrix=[1,2,3,...... ,16]*.
2. *Initialize.* First, use function random.shuffle(matrix) to disorder the list *matrix[ ]*. Then, use function print\_map(matrix) to print the list in 3×3 or 4×4 arrangement.
3. *Moving the adjacent tile.* The blank space and the number on its adjacent place will exchange the value.
4. *Limit the tile withing the square location.* Create functions find\_x and find\_y to find the blank space’s horizontal and vertical axis. As long as one of the axis exceeds the size of the matrix, the move will not come into effect.

The breakdown of the logic includes: If the input in mode is other than 8 or 15, the mode will default to 15.

1. Python Object
2. Global variables

mode: The mode selected by the player, 8 or 15.

matrix: The one-dimensional list that contains number 0 to 8 or 0 to 15.

d, u, r, l: Four user-defined keys, respectively represent moving the adjacent tile up, down, left and right.

movekey\_list: A list that consists d,u,t,l, that is, all the keys.

notice1: The instruction to prompt the user to enter one of the self-defined keys to move the tiles.

t: The index of number 0 in the one-dimensional list.

count: The number of already-taken steps.

H: The relative horizontal axis of the blank space, left, middle or right.

V: The relative vertical axis of the blank space, up middle or down.

1. Local variables

num: element in list matrix[ ].

h: The horizontal axis of the blank space.

v: The vertical axis of the blank space.

temp: A transfer variable to storage the number that is going to replace the blank space.

1. Structure

The program’s flow will follow the global function, and will come to loops in each functions.

Beginning of while(state)--mode 8 or 15--entering(d,u,r,l)--print\_map(matrix)-- access appointed mode--find\_x( )--find\_y( )--action8( ) or action15( )--count-- Bool:matrix==standard matrix?if False, go back to find\_x( )and find\_y( ); if True-- Bool(final),if final==1 or 2,go back to the beginning of while(state); if final==q, break.

1. Logic of Generating the matrix

Go through the whole list *matrix* and check each element. First check the position. If the position is on the right end of the matrix, then print a “\n” after this number is printed. Then check the value of the element. If the element is 0, print a double whitespace “ ”. If the element is other than 0, then print the element itself. “%2d” is used to have the output occupy two spaces, so as to regulate the alignment of the matrix which may contain both unit-digits and double-digits.

To avoid the case of unsolvable circunstance, a procedure is produced in function print\_map( ) to count the number of inversition. For 3×3 mode, if the number of inversition is odd, then the matrix is supposed to be disordered again, until it is solvable. For 4×4 mode, only when the number of inversition plus the row of blank space is odd is the block solvable.

1. Functions
2. print\_map(matrix). A loop which will print a blank space if the number is 0, and change the line after every 3 or 4 outputted number. It introduces the list *matrix*.
3. *find\_x8(t)*: A function that identifies the horizontal axis of the blank space in mode 8. It introduces parameter t which is the index of number 0 in the one-dimensional list. It returns the value of the horizontal axis of the blank space as H.
4. *find\_x15(t)*:A function that identifies the horizontal axis of the blank space in mode 15. It introduces parameter t which is the index of number 0 in the one-dimensional list. It returns the value of the horizontal axis of the blank space as H.
5. *find\_y8(t)*: A function that identifies the vertical axis of the blank space in mode 8. It introduces parameter t which is the index of number 0 in the one-dimensional list. It returns the value of the horizontal axis of the blank space as V.
6. *find\_y15(t)*: A function that identifies the vertical axis of the blank space in mode 15. It introduces parameter t which is the index of number 0 in the one-dimensional list. It returns the value of the horizontal axis of the blank space as H.
7. *action8(V,H,matrix,u,d,l,r,t,notice)* and *action15(V,H,matrix,u,d,l,r,t,notice)*: First, print a notice to instruct the user enter the self-defined key to move the tile. The notice, denoted as *notice*, will automatically include corresponding available direction at different positions. Second, by exchanging the value, the blank space will be replaced by the adjacent number. After the movement, the for loop for num in matrix will update the arrangement of the matrix.
8. Sample Outputs



