**实验题目:**

**八数码问题**

**import** numpy **as** np  
  
*#返回两个数组对应位置相同值的个数***def** calc(state1):  
 b = np.array([[1, 2, 3], [8, 0, 4], [7, 6, 5]])  
 postion = np.where(state1 == b)  
 **return** len(state1[postion])  
*#打印八数码***def** showInfo(a):  
 **for** i **in** range(3):  
 **for** j **in** range(3):  
 print(a[i, j], end=**' '**)  
 print(**"\n"**)  
 print(**'->'**)  
directions = [**'up'**, **'down'**, **'left'**, **'right'**]  
  
**def** SubStates(state):  
 subStates = []  
 row, col = np.where(state==0)  
 **for** direction **in** directions:  
 **if 'left'** == direction **and** col > 0:  
 s = state.copy()  
 s[row, col],s[row, col - 1] = s[row, col - 1],s[row, col]  
 subStates.append(s)  
 **if 'up'** == direction **and** row > 0:  
 s = state.copy()  
 s[row, col],s[row - 1, col] = s[row - 1, col],s[row, col]  
 subStates.append(s)  
 **if 'down'** == direction **and** row < 2:  
 s = state.copy()  
 s[row, col],s[row + 1, col] = s[row + 1, col],s[row, col]  
 subStates.append(s)  
 **if 'right'** == direction **and** col < 2:  
 s = state.copy()  
 s[row, col],s[row, col + 1] = s[row, col + 1],s[row, col]  
 subStates.append(s)  
 **return** subStates  
**def** DFS(first):  
 stack = []  
 stack.append(first)  
 count = -1  
 **while** stack:  
 count += 1  
 node = stack.pop()  
 showInfo(node)  
 **if** calc(node) == 9:  
 **return True**,count  
 s = SubStates(node)  
 res = sorted(s, key=calc)  
 **for** x **in** res:  
 stack.append(x)  
  
  
*#主函数***def** main():  
 start = np.array([[0, 1, 3], [8, 2, 4], [7, 6, 5]])  
 *#start = np.array([[2, 8, 3], [1, 0, 4], [7, 6, 5]]) 无限循环* res,count = DFS(start)  
 **if** res:  
 print(**'经过%d次变换结束'** %count)  
**if** \_\_name\_\_ == **'\_\_main\_\_'**:  
 main()

**实验结果：**

