```python
def print_puzzle(arr):
    print(20*'-')
    for i in range(len(arr)):
        print(arr[i],end ="  ")
        if(i==2 or i==5 or i==8):
            print('')

def misplaced_elements(curr,goal):
    count = 0
    for i in range(len(goal)):
        if(goal[i]!=curr[i]):
            count +=1
    return count

def puzzle_sol(current,goal,function_value,Node_number):
    print_puzzle(current)
    if((current==goal)!=True):
        Node_number += 1
        index_of_empty = current.index('_')
    #---------------for Index 0---------------------
        if(index_of_empty == 0):
            arr1 = current.copy()
            arr1[1] = current[0]
            arr1[0] = current[1]
            c1 = Node_number + misplaced_elements(arr1,goal)
            if(c1<=function_value):
                function_value = c1
                current = arr1.copy()
            arr3 = current.copy()
            arr3[3] = current[0]
            arr3[0] = current[3]
            c3 = Node_number + misplaced_elements(arr3,goal)
            if(c3<=function_value):
                function_value = c3
                current = arr3.copy()

            puzzle_sol(current,goal,function_value,Node_number)
    #---------------for Index 1---------------------
        elif(index_of_empty == 1):
            arr2 = current.copy()
            arr2[2] = current[1]
            arr2[1] = current[2]
            c2 = Node_number + misplaced_elements(arr2,goal)
            if(c2<=function_value):
                function_value = c2
                current = arr2.copy()
            arr4 = current.copy()
            arr4[4] = current[1]
            arr4[1] = current[4]
            c4 = Node_number + misplaced_elements(arr4,goal)
            if(c4<=function_value):
                function_value = c4
                current = arr4.copy()
            arr0 = current.copy()
            arr0[0] = current[1]
            arr0[1] = current[0]
            c0 = Node_number + misplaced_elements(arr0,goal)
            if(c0<=function_value):
                function_value = c0
                current = arr0.copy()
```

```
            puzzle_sol(current,goal,function_value,Node_number)
    #---------------for Index 2---------------------
        elif(index_of_empty == 2):
            arr5 = current.copy()
            arr5[5] = current[2]
            arr5[2] = current[5]
            c5 = Node_number + misplaced_elements(arr5,goal)
            if(c5<=function_value):
                function_value = c5
                current = arr5.copy()
            arr1 = current.copy()
            arr1[1] = current[2]
            arr1[2] = current[1]
            c1 = Node_number + misplaced_elements(arr1,goal)
            if(c1<=function_value):
                function_value = c1
                current = arr1.copy()

            puzzle_sol(current,goal,function_value,Node_number)
    #---------------for Index 3---------------------
        elif(index_of_empty == 3):
            arr4 = current.copy()
            arr4[3] = current[4]
            arr4[4] = current[3]
            c4 = Node_number + misplaced_elements(arr4,goal)
            if(c4<=function_value):
                function_value = c4
                current = arr4.copy()
            arr6 = current.copy()
            arr6[3] = current[6]
            arr6[6] = current[3]
            c6 = Node_number + misplaced_elements(arr6,goal)
            if(c6<=function_value):
                function_value = c6
                current = arr6.copy()
            arr0 = current.copy()
            arr0[3] = current[0]
            arr0[0] = current[3]
            c0 = Node_number + misplaced_elements(arr0,goal)
            if(c0<=function_value):
                function_value = c0
                current = arr0.copy()
            puzzle_sol(current,goal,function_value,Node_number)
    #---------------for Index 4---------------------
        elif(index_of_empty == 4):
            arr5 = current.copy()
            arr5[4] = current[5]
            arr5[5] = current[4]
            c5 = Node_number + misplaced_elements(arr5,goal)
            if(c5<=function_value):
                function_value = c5
                current = arr5.copy()
            arr7 = current.copy()
            arr7[4] = current[7]
            arr7[7] = current[4]
            c7 = Node_number + misplaced_elements(arr7,goal)
            if(c7<=function_value):
                function_value = c7
                current = arr7.copy()
            arr3 = current.copy()
```

```
            arr3[3] = current[4]
            arr3[4] = current[3]
            c3 = Node_number + misplaced_elements(arr3,goal)
            if(c3<=function_value):
                function_value = c3
                current = arr3.copy()
            arr1 = current.copy()
            arr1[1] = current[4]
            arr1[4] = current[1]
            c1 = Node_number + misplaced_elements(arr1,goal)
            if(c1<=function_value):
                function_value = c1
                current = arr1.copy()
            puzzle_sol(current,goal,function_value,Node_number)
    #---------------for Index 5---------------------
        elif(index_of_empty == 5):
            arr8 = current.copy()
            arr8[5] = current[8]
            arr8[8] = current[5]
            c8 = Node_number + misplaced_elements(arr8,goal)
            if(c8<=function_value):
                function_value = c8
                current = arr8.copy()
            arr4 = current.copy()
            arr4[4] = current[5]
            arr4[5] = current[4]
            c4 = Node_number + misplaced_elements(arr4,goal)
            if(c4<=function_value):
                function_value = c4
                current = arr4.copy()
            arr2 = current.copy()
            arr2[5] = current[2]
            arr2[2] = current[5]
            c2 = Node_number + misplaced_elements(arr2,goal)
            if(c2<=function_value):
                function_value = c2
                current = arr2.copy()
            puzzle_sol(current,goal,function_value,Node_number)
    #---------------for Index 6---------------------
        elif(index_of_empty == 6):
            arr7 = current.copy()
            arr7[7] = current[6]
            arr7[6] = current[7]
            c7 = Node_number + misplaced_elements(arr7,goal)
            if(c7<=function_value):
                function_value = c7
                current = arr7.copy()
            arr3 = current.copy()
            arr3[3] = current[6]
            arr3[6] = current[3]
            c3 = Node_number + misplaced_elements(arr3,goal)
            if(c3<=function_value):
                function_value = c3
                current = arr3.copy()
            puzzle_sol(current,goal,function_value,Node_number)
    #---------------for Index 7---------------------
        elif(index_of_empty == 7):
            arr8 = current.copy()
            arr8[7] = current[8]
            arr8[8] = current[7]
```

```
            c8 = Node_number + misplaced_elements(arr8,goal)
            if(c8<=function_value):
                function_value = c8
                current = arr8.copy()
            arr4 = current.copy()
            arr4[4] = current[7]
            arr4[7] = current[4]
            c4 = Node_number + misplaced_elements(arr4,goal)
            if(c4<=function_value):
                function_value = c4
                current = arr4.copy()
            arr6 = current.copy()
            arr6[6] = current[7]
            arr6[7] = current[6]
            c6 = Node_number + misplaced_elements(arr6,goal)
            if(c6<=function_value):
                function_value = c6
                current = arr6.copy()
            puzzle_sol(current,goal,function_value,Node_number)
    #---------------for Index 8---------------------
        elif(index_of_empty == 8):
            arr5 = current.copy()
            arr5[5] = current[8]
            arr5[8] = current[5]
            c5 = Node_number + misplaced_elements(arr5,goal)
            if(c5<=function_value):
                function_value = c5
                current = arr5.copy()
            arr7 = current.copy()
            arr7[8] = current[7]
            arr7[7] = current[8]
            c7 = Node_number + misplaced_elements(arr7,goal)
            if(c7<=function_value):
                function_value = c7
                current = arr7.copy()
            puzzle_sol(current,goal,function_value,Node_number)


Node_number = 0
function_value = 0
goal = [1,2,3,4,5,6,7,8,'_']
current = [1,2,3,'_',4,6,7,5,8]
final_hx = misplaced_elements(current,goal)
function_value = Node_number + final_hx
puzzle_sol(current,goal,function_value,Node_number)
```