**Group Members: Abdul Moiz Asif (263802) + Muhammad Asim Khan (254089)**

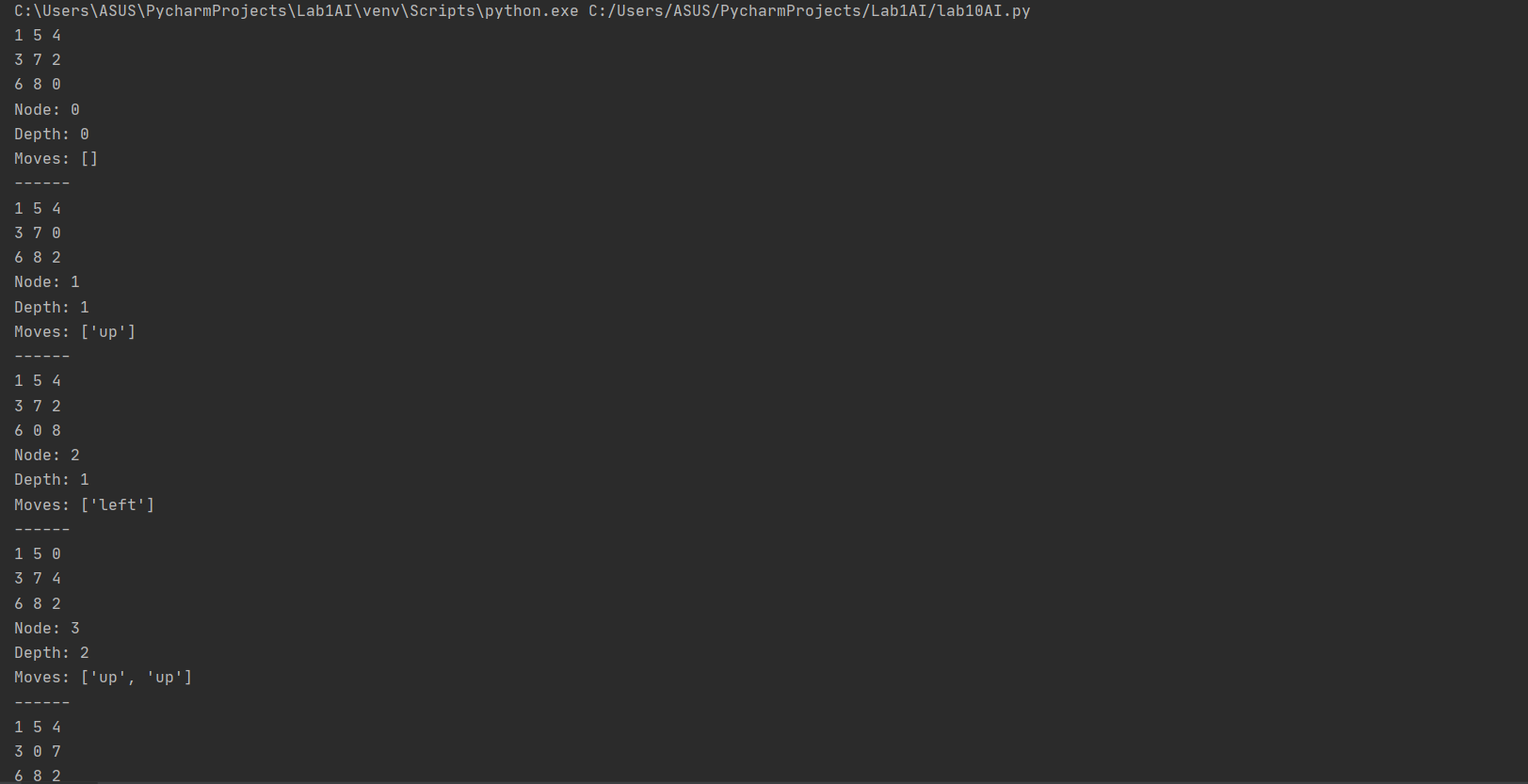
**LAB 10 AI**

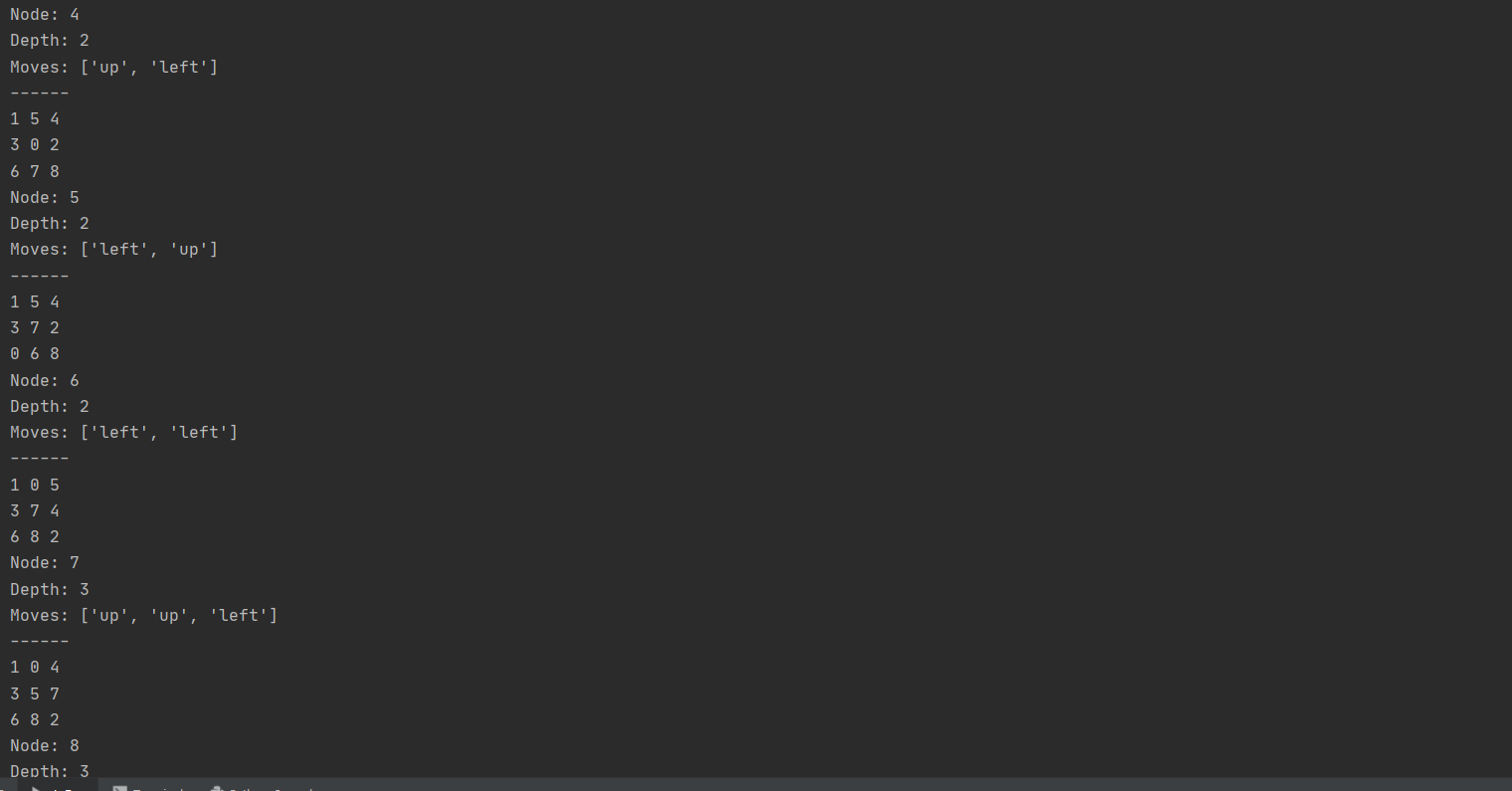
**Code for BFS:**

from queue import Queue  
import copy  
import time  
  
  
def printNode(node):  
 print(node[0], node[1], node[2])  
 print(node[3], node[4], node[5])  
 print(node[6], node[7], node[8])  
 global nodeNumber  
 print('Node:', nodeNumber)  
 print('Depth:', len(node[9:]))  
 print('Moves:', node[9:])  
 print('------')  
 nodeNumber += 1  
  
  
def checkFinal(node):  
 if node[:9] == finalNode:  
 printNode(node)  
 return True  
 if node[:9] not in visitedList:  
 printNode(node)  
 queue.put(node)  
 visitedList.append(node[:9])  
 return False  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 startNode = [1, 5, 4, 3, 7, 2, 6, 8, 0]  
 finalNode = [0, 1, 2, 3, 4, 5, 6, 7, 8]  
  
 found = False  
 nodeNumber = 0  
 visitedList = []  
 queue = Queue()  
 queue.put(startNode)  
 visitedList.append(startNode)  
 printNode(startNode)  
 t0 = time.time()  
  
 while (not found and not queue.empty()):  
 currentNode = queue.get()  
 blankIndex = currentNode.index(0)  
 if blankIndex != 0 and blankIndex != 1 and blankIndex != 2:  
 upNode = copy.deepcopy(currentNode)  
 upNode[blankIndex] = upNode[blankIndex - 3]  
 upNode[blankIndex - 3] = 0  
 upNode.append('up')  
 found = checkFinal(upNode)  
 if blankIndex != 0 and blankIndex != 3 and blankIndex != 6 and found == False:  
 leftNode = copy.deepcopy(currentNode)  
 leftNode[blankIndex] = leftNode[blankIndex - 1]  
 leftNode[blankIndex - 1] = 0  
 leftNode.append('left')  
 found = checkFinal(leftNode)  
 if blankIndex != 6 and blankIndex != 7 and blankIndex != 8 and found == False:  
 downNode = copy.deepcopy(currentNode)  
 downNode[blankIndex] = downNode[blankIndex + 3]  
 downNode[blankIndex + 3] = 0  
 downNode.append('down')  
 found = checkFinal(downNode)  
 if blankIndex != 2 and blankIndex != 5 and blankIndex != 8 and found == False:  
 rightNode = copy.deepcopy(currentNode)  
 rightNode[blankIndex] = rightNode[blankIndex + 1]  
 rightNode[blankIndex + 1] = 0  
 rightNode.append('right')  
 found = checkFinal(rightNode)  
  
 t1 = time.time()  
 print('Time:', t1 - t0)  
 print('------')

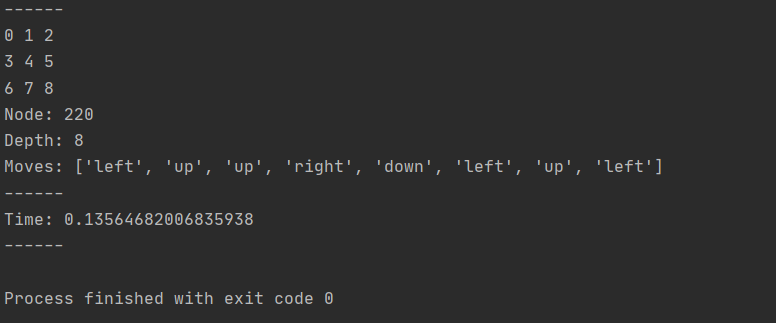
**Output**:

The Output is very long so only taking the screenshot of some code.





**Final Output:**

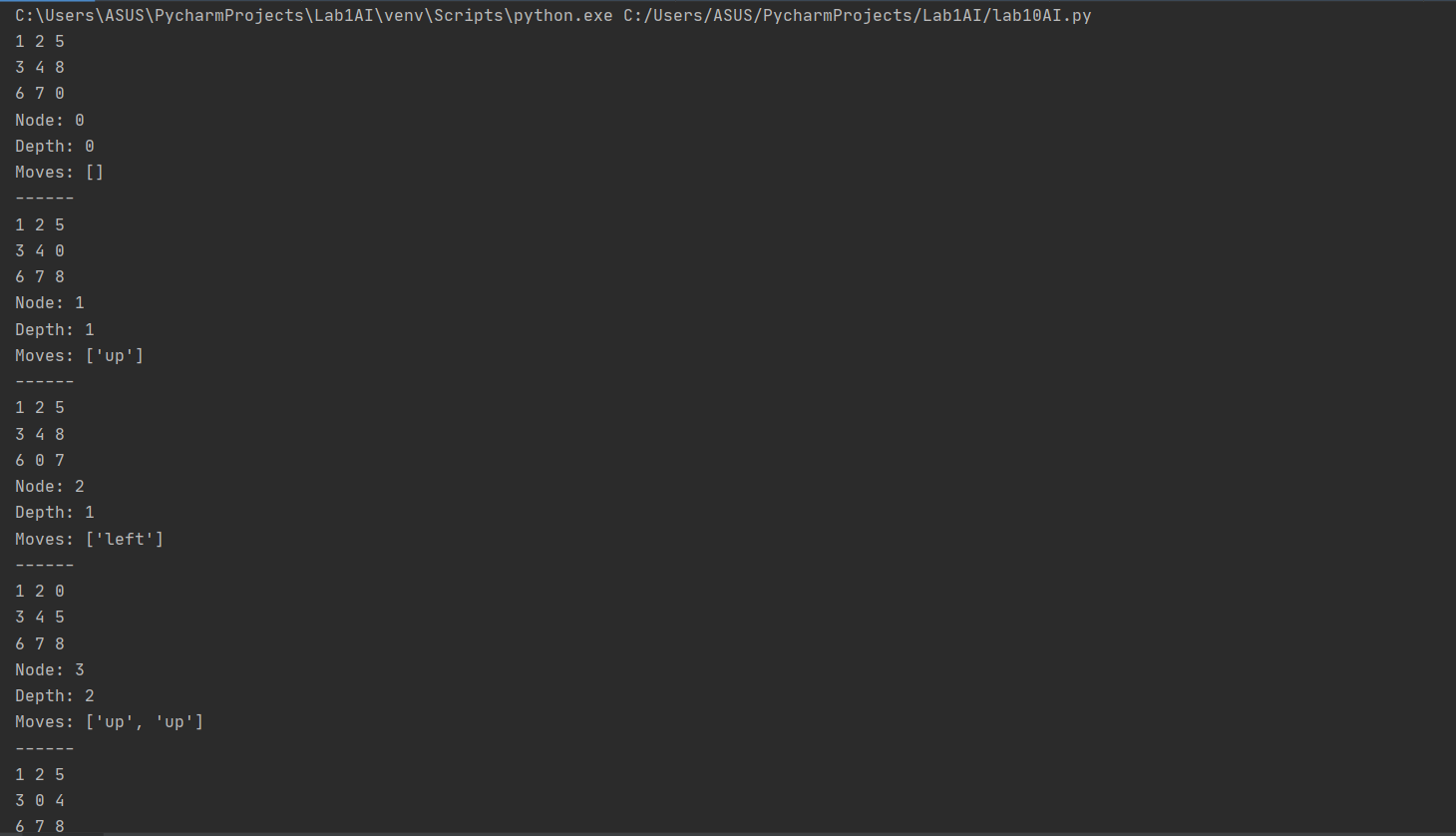


**Task#2**

**Code with DFS:**

import copy  
import time  
  
  
def printNode(node):  
 print(node[0], node[1], node[2])  
 print(node[3], node[4], node[5])  
 print(node[6], node[7], node[8])  
 global nodeNumber  
 print('Node:', nodeNumber)  
 print('Depth:', len(node[9:]))  
 print('Moves:', node[9:])  
 print('------')  
 nodeNumber += 1  
  
  
def checkFinal(node):  
 if node[:9] == finalNode:  
 printNode(node)  
 return True  
 global insertIndex  
 if node[:9] not in visitedList:  
 printNode(node)  
 stack.insert(insertIndex, node)  
 insertIndex += 1  
 visitedList.append(node[:9])  
 return False  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 startNode = [1, 2, 5, 3, 4, 8, 6, 7, 0]  
 finalNode = [0, 1, 2, 3, 4, 5, 6, 7, 8]  
  
 found = False  
 nodeNumber = 0  
 visitedList = []  
 stack = []  
 stack.append(startNode)  
 visitedList.append(startNode)  
 printNode(startNode)  
 t0 = time.time()  
  
 while (not found and not len(stack) == 0):  
 currentNode = stack.pop(0)  
 blankIndex = currentNode.index(0)  
 insertIndex = 0  
 if blankIndex != 0 and blankIndex != 1 and blankIndex != 2:  
 upNode = copy.deepcopy(currentNode)  
 upNode[blankIndex] = upNode[blankIndex - 3]  
 upNode[blankIndex - 3] = 0  
 upNode.append('up')  
 found = checkFinal(upNode)  
 if blankIndex != 0 and blankIndex != 3 and blankIndex != 6 and found == False:  
 leftNode = copy.deepcopy(currentNode)  
 leftNode[blankIndex] = leftNode[blankIndex - 1]  
 leftNode[blankIndex - 1] = 0  
 leftNode.append('left')  
 found = checkFinal(leftNode)  
 if blankIndex != 6 and blankIndex != 7 and blankIndex != 8 and found == False:  
 downNode = copy.deepcopy(currentNode)  
 downNode[blankIndex] = downNode[blankIndex + 3]  
 downNode[blankIndex + 3] = 0  
 downNode.append('down')  
 found = checkFinal(downNode)  
 if blankIndex != 2 and blankIndex != 5 and blankIndex != 8 and found == False:  
 rightNode = copy.deepcopy(currentNode)  
 rightNode[blankIndex] = rightNode[blankIndex + 1]  
 rightNode[blankIndex + 1] = 0  
 rightNode.append('right')  
 found = checkFinal(rightNode)  
  
 t1 = time.time()  
 print('Time:', t1 - t0)  
 print('------')

**Output:**

****

