class Node:

def \_\_init\_\_(self, data):

self.data = data

self.next = None

class NodeOperation:

def pushNode(self, head\_ref, data\_val):

new\_node = Node(data\_val)

new\_node.next = head\_ref

head\_ref = new\_node

return head

def printNode(self, head):

while (head != None):

print('%d->' % head.data, end="")

head = head.next

print("NULL")

def getLen(self, head):

temp = head

len = 0

while (temp != None):

len += 1

temp = temp.next

return len

def printMiddle(self, head):

if head != None:

# find length

len = self.getLen(head)

temp = head

# traverse till we reached half of length

midIdx = len // 2

while midIdx != 0:

temp = temp.next

midIdx -= 1

print('The middle element is [ %d ]' % temp.data)

head = None

temp = NodeOperation()

for i in range(5, 0, -1):

head = temp.pushNode(head, i)

temp.printNode(head)

temp.printMiddle(head)