class Node:

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

self.data=data

self.next=None

class LinkedList:

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

self.head=None

def push(self,new\_data):

if self.head is None:

newnode=Node(new\_data)

self.head=newnode

else:

current=self.head

while(current.next!=None):

current=current.next

current.next=Node(new\_data)

def print(self):

temp=self.head

while(temp):

print(temp.data,end='')

temp=temp.next

def middle(self):

slow\_ptr=self.head

fast\_ptr=self.head

while(fast\_ptr.next.next!=None and fast\_ptr.next!=None):

slow\_ptr=slow\_ptr.next

fast\_ptr=fast\_ptr.next.next

if fast\_ptr.next != None:

slow\_ptr = slow\_ptr.next

return slow\_ptr.data

if \_\_name\_\_=="\_\_main\_\_":

list1=LinkedList()

list1.push(5)

list1.push(4)

list1.push(2)

list1.push(3)

list1.push(1)

list1.push(6)

list1.print()

print()

print(list1.middle())