#################################################################

class ListItem(object):

def \_\_init\_\_(self ,key,value):

self.key = key

self.value = value

class SortSequenceList(object):

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

self.SeqList=[]

def CreateSequenceListByInput(self,nElement):

self.SeqList.append(ListItem(int(0), 0))

print("请输入数据：")

for i in range(1,nElement+1):

a = input()

self.SeqList.append(ListItem(int(a), i))

def TraverseElementSet(self):

for i in range(1,len(self.SeqList)):

print(self.SeqList[i].key)

#############################

#算法8-3 折半插入排序

#############################

def BinInsertSort(self):

SeqListLen = len(self.SeqList)

for i in range(2,SeqListLen):

SeqLeft = 1

SeqRight = i - 1

self.SeqList[0].key =self.SeqList[i].key

while SeqLeft <= SeqRight:

SeqMid = (SeqLeft + SeqRight) // 2

if self.SeqList[SeqMid].key > self.SeqList[0].key:

SeqRight = SeqMid - 1

else:

SeqLeft = SeqMid + 1

j = i - 1

while j >= SeqLeft:

self.SeqList[j + 1].key = self.SeqList[j].key

j = j-1

self.SeqList[SeqLeft].key = self.SeqList[0].key

if \_\_name\_\_ =='\_\_main\_\_':

SL=SortSequenceList()

SL.CreateSequenceListByInput(5)

SL.BinInsertSort()

print('折半插入排序算法结果为:')

SL.TraverseElementSet()