**Python习题选编**

华东师大信息学院计算中心

**一．程序结构与算法部分：**

1. 编写一个python程序，输入两个数，比较它们的大小并输出其中较大者。

参考代码：

x = int(input("Please enter first integer: "))

y = int(input("Please enter second integer: "))

if (x == y):

print("两数相同！")

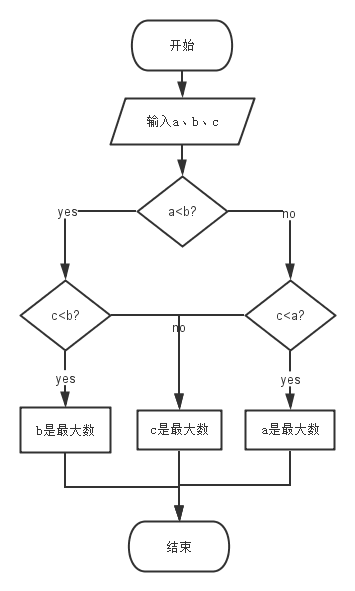
elif (x > y):

print("较大数为：",x)

else:

print("较大数为：",y);

1. 写一个算法（流程图和python程序）：输入三个数，输出其最大者。

参考代码：

a,b,c=3,4,5

if a <= b:

if c < b:

print ("b是最大的数")

else:

print ("c是最大的数")

else:

if c < a:

print ("a是最大的数")

else:

print ("c是最大的数")

1. 使用Python编程，求1～100间所有偶数的和。

参考代码：

sum=0

for x in range(1,101):

if x % 2==0:

print(x)

sum=sum+x

print("累加和是:",sum)

1. 用Python编写程序，输入一年份，判断该年份是否是闰年并输出结果。

**注：凡符合下面两个条件之一的年份是闰年。**

1. **能被4整除但不能被100整除。**
2. **能被400整除。**

参考代码：

year = int(input("Please enter the year: "))

if ((year%4==0 and year%100!=0) or (year%400==0)):

print(year,"is a leap year.")

else:

print(year,"is not a leap year.")

1. 用Python编程，假设一年期定期利率为3.25%，计算一下需要过多少年，一万元的一年定期存款连本带息能翻番？

参考代码：

cunkuan = 10000 #本金10000元

years=0

while cunkuan<20000:

years+=1

cunkuan=cunkuan\*(1+0.0325)

print(str(years)+"年以后，存款会翻番")

1. **从键盘接收一百分制成绩（0~100），要求输出其对应的成绩等级A~E。其中，90分以上为'A'，80~89分为'B'，70~79分为'C'，60~69分为'D'，60分以下为'E'。**

参考答案：

score=int(input('请输入成绩（0～100）：'))

if score > 100:

grade = "输入错误！"

elif score >= 90:

grade = 'A'

elif score >= 80:

grade = 'B'

elif score >= 70:

grade = 'C'

elif score >= 60:

grade = 'D'

elif score >= 0:

grade = 'E'

else:

grade = "输入错误！"

print(grade)

1. **猜数游戏。预设一个0~9之间的整数，让用户猜一猜并输入所猜的数，如果大于预设的数，显示“太大”；小于预设的数，显示“太小”，如此循环，直至猜中该数，显示“恭喜！你猜中了！”。**

参考答案：

num=7

while True:

guess=int(input('请输入你猜的数（0～9）：'))

if guess == num:

print("恭喜！你猜中了！")

break;

elif guess > num:

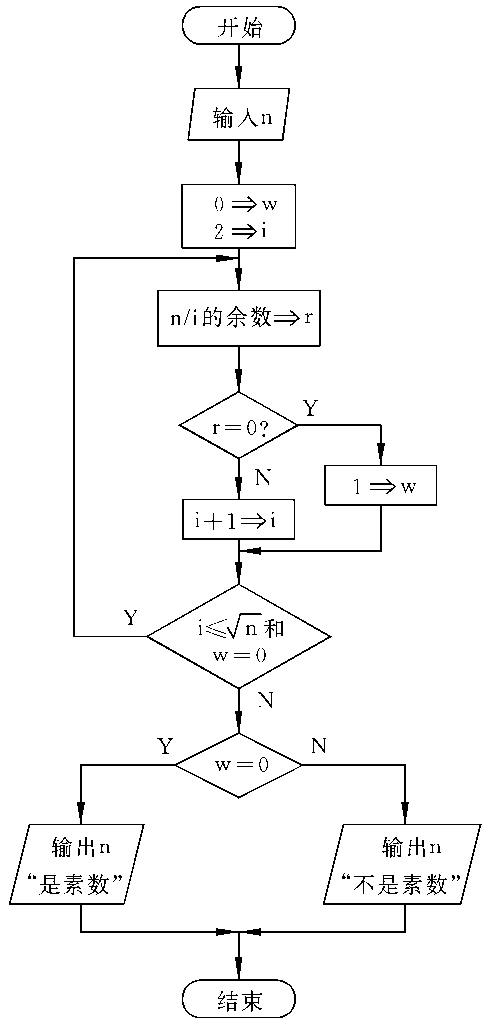
print("太大")

else:

print("太小")

1. **输入一个数，判断这个数是否为素数，并输出判断结果。**

（所谓素数，是指除了1和该数本身之外，不能被其它任何整数整除的数。下图为参考流程图）



参考答案：

import math

n=int(input("请输入一个数:"))

x=int(math.sqrt(n))

i,w=2,0

for i in range (2,x+1):

if n%i==0:

w=1

if w==1:

print(n,"不是素数。")

else:

print(n,"是素数。")

或

import math

n=int(input('请输入一个数：'))

i,w=2,0

while i <= int(math.sqrt(n)) and w==0:

if n%i == 0:

w = 1

break

else:

i = i+1

if w==0:

print(n,"是素数！")

else:

print(n,"不是素数！")

或

import math

n=int(input('请输入一个数：'))

i=2

while i <= int(math.sqrt(n)) :

if n%i == 0:

print(n,"不是素数！")

break

else:

i=i+1

else:

print(n,"是素数！")

1. **输入一个时间（小时:分钟:秒），输出该时间经过5分30秒后的时间。**

参考答案：

hour,minute,second=input('请输入一个时间（h:m:s）:').split(':')

hour=int(hour)

minute=int(minute)

second=int(second)

second+=30

if second >= 60:

second = second-60

minute += 1

minute+=5

if minute >= 60:

minute = minute-60

hour += 1

if hour == 24:

hour = 0

print('%d:%d:%d'%(hour,minute,second))

1. **一个数如果恰好等于它的因子之和，这个数就称为“完数”。例如，6的因子为1、2、3，而6=1+2+3，因此6是完数。编程，找出1000之内的所有完数，并输出该完数及对应的因子。（枚举法）**

参考答案：

m=1000

for a in range(2,m+1):

s=a

L1=[]

for i in range(1,a):

if a%i==0:

s-=i

L1.append(i)

if s==0:

print("完数：%d，因子包括："%a,end="")

for j in range(1,len(L1)):

print("%d"%L1[j],end=",")

print("\n")

**11．编程，解决猴子吃桃问题。**

**猴子第一天摘下若干个桃子，当即吃了一半，还不过瘾，又多吃了一个。第二天早上又将剩下的桃子吃掉一半，又多吃了一个。以后每天早上都吃了前一天剩下的一半零一个。到第10天早上想吃时，只剩下一个桃子了。求第一天共摘多少个桃子。（迭代法）**

参考答案：

day=9

x=1

while day>0:

x=(x+1)\*2

day-=1

print("total=",x)

**二．输入输出与文件部分：**

1. 编写一个python程序，输入两个数，输出两数之和。

参考代码：

x = int(input("Please enter first integer: "))

y = int(input("Please enter second integer: "))

print("The sum is:");

print(x+y);

**2. 在当前目录下有一个文件名为temp.txt的文件，存放着上海从2014年3月10日（周一）到3月16日（周日）间一周的最高和最低气温（单位为摄氏度）。其中，第一行为最高气温，第二行为最低气温。编程，找出这一周中第几天最热（按最高气温计算）？最高多少度？这一周中第几天最冷（按最低气温计算）？最冷多少度？**

参考答案：

flname="temp.txt"

f=open(flname)

ht=(f.readline()).strip()

L1=list(ht.split(','))

lt=(f.readline()).strip()

L2=list(lt.split(','))

f.close()

for i in range(len(L1)):

L1[i]=int(L1[i])

L2[i]=int(L2[i])

maxVal=L1[0]

maxDay=0

minVal=L2[0]

minDay=0

for i in range(1,len(L1)):

if L1[i]>maxVal:

maxVal=L1[i]

maxDay=i

if L2[i]<minVal:

minVal=L2[i]

minDay=i

print("这周第"+str(maxDay+1)+"天最热，最高"+str(maxVal)+"摄氏度")

print("这周第"+str(minDay+1)+"天最冷，最低"+str(minVal)+"摄氏度")

**3．在上题的基础上，求出全周的平均气温（这一周各天平均温度的平均值，取整数）。假设在气象意义上，入春标准是连续5天日均气温超过10℃，根据这一周的气象数据是否能判断上海已经入春？**

参考答案：

flname="temp.txt"

f=open(flname)

ht=(f.readline()).strip()

L1=list(ht.split(','))

lt=(f.readline()).strip()

L2=list(lt.split(','))

f.close()

L3=[]

for i in range(len(L1)):

L1[i]=int(L1[i])

L2[i]=int(L2[i])

L3.append(int((L1[i]+L2[i])/2))

sum=0

k=0

for i in range(len(L3)):

sum=sum+L3[i]

if L3[i]>=10:

k+=1

else:

k=0

avg=int(sum/len(L3))

print("周平均气温为：",avg)

if k>=5:

print("上海这周已入春。")

else:

print("上海这周未入春。")

**4．当前目录下有一个文件名为score1.txt的文本文件，存放着某班学生的计算机课成绩，共有学号、平时成绩、期末成绩三列。请根据平时成绩占40%，期末成绩占60%的比例计算总评成绩（取整数），并分学号、总评成绩两列写入另一文件score2.txt。同时在屏幕上输出学生总人数，按总评成绩计90以上、80～89、70～79、60～69、60分以下各成绩档的人数和班级总平均分（取整数）。**

参考答案：

f=open("score1.txt")

a=f.readline()

line=(f.readline()).strip()

f2=open("score2.txt",'w')

f2.write("学号 平均成绩\n");

L2=[0,0,0,0,0]

count=0

sum=0

while (len(line) != 0):

#print(line)

L1=line.split()

f2.write(L1[0]+" ")

f\_score=int(int(L1[1])\*0.4+int(L1[2])\*0.6)

if 90<f\_score<=100:

L2[0]+=1

elif f\_score>=80:

L2[1]+=1

elif f\_score>=70:

L2[2]+=1

elif f\_score>=60:

L2[3]+=1

else:

L2[4]+=1

count+=1

sum+=f\_score

f2.write(str(f\_score)+"\n")

line=(f.readline()).strip()

f.close()

f2.close()

avg\_score=int(sum/count)

print("学生总人数为%d，按总评成绩计,90以上%d人、80～89间%d人、70～79间%d人、60～69间%d人、60分以下%d人。班级总平均分为%d分。"%(count,L2[0],L2[1],L2[2],L2[3],L2[4],avg\_score))

或

f=open("score1.txt")

a=f.readlines()

del a[0]

L3=[]

for line in a:

line=line.strip()

L1=line.split()

f\_score=int(int(L1[1])\*0.4+int(L1[2])\*0.6)

L3.append([L1[0],f\_score])

f.close()

c=[0,0,0,0,0]

count=0

sum=0

f2=open("score2.txt",'w')

f2.write("学号 平均成绩\n");

for L2 in L3:

if 90<L2[1]<=100:

c[0]+=1

elif L2[1]>=80:

c[1]+=1

elif L2[1]>=70:

c[2]+=1

elif L2[1]>=60:

c[3]+=1

else:

c[4]+=1

count+=1

sum+=L2[1]

f2.write(L2[0]+" "+str(L2[1])+"\n")

f2.close()

avg\_score=int(sum/count)

print("学生总人数为%d，按总评成绩计,90以上%d人、80～89间%d人、70～79间%d人、60～69间%d人、60分以下%d人。班级总平均分为%d分。"%(count,c[0],c[1],c[2],c[3],c[4],avg\_score))

**5．当前目录下有一个文本文件sample12.txt，其内容包含小写字母和大写字母。请将该文件复制到另一文件sample12\_copy.txt，并将原文件中的小写字母全部转换为大写字母，其余格式均不变。**

参考答案：

f=open("sample12.txt")

L1=f.readlines()

f2=open("sample12\_copy.txt",'w')

for line in L1:

f2.write(line.upper())

f.close()

f2.close()

**6．当前目录下有一个文件名为class\_score.txt的文本文件，存放着某班学生的学号、数学课成绩（第2列）和语文课成绩（第3列）。请编程完成下列要求：**

**（1）分别求这个班数学和语文的平均分（保留1位小数）并输出。**

**（2）找出两门课都不及格（<60）的学生，输出他们的学号和各科成绩。**

**（3）找出两门课的平均分在90分以上的学生，输出他们的学号和各科成绩。**

**建议用三个函数分别实现以上要求。**

参考答案：

def output\_avg(L):

sum1,sum2=0,0

for line in L:

L1=line.strip().split()

sum1+=int(L1[1])

sum2+=int(L1[2])

count=len(L)

avg1=round(sum1/count,1)

avg2=round(sum2/count,1)

print("这个班的数学平均分为：%4.1f，语文平均分为：%4.1f"%(avg1,avg2))

def output\_notpass(L):

print("两门课均不及格的学生学号及数学、语文成绩为：")

for line in L:

L1=line.strip().split()

if int(L1[1])<60 and int(L1[2])<60:

print(line)

def output\_good(L):

print("两门课平均分在90分以上的学生学号及数学、语文成绩为：")

for line in L:

L1=line.strip().split()

f\_score=round((int(L1[1])+int(L1[2]))/2)

if f\_score>=90:

print(line)

f=open("class\_score.txt")

L=f.readlines()

del L[0]

output\_avg(L)

output\_notpass(L)

output\_good(L)

1. **算法分析与设计部分**
2. **编程，从键盘接收若干个整数（直接输入回车表示结束），用冒泡法或选择法进行排序（从小到大），并将排序结果在屏幕上输出。同时估计算法的复杂度。**

参考答案：

选择排序：

def bubble(List):

num=len(List)

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

for j in range(i+1,num):

if List[i]>List[j]:

List[i],List[j]=List[j],List[i]

return List

L1=[]

num\_str=input('请输入一个需排序的整数：')

while len(num\_str) != 0:

L1.append(int(num\_str))

num\_str=input('请输入一个需排序的整数：')

print('排序后结果:', bubble(L1))

或

冒泡排序：

def bubble(List):

for i in range(0,len(List)-1):

for j in range(len(List)-1,i,-1):

if List[j-1]>List[j]:

List[j-1],List[j]=List[j],List[j-1]

return List

L1=[]

num\_str=input('请输入一个需排序的整数：')

while len(num\_str) != 0:

L1.append(int(num\_str))

num\_str=input('请输入一个需排序的整数：')

print('排序后结果:', bubble(L1))

上述两个程序算法的时间复杂度均为O(n2).

**2．从键盘接收一个正整数n，输出对应斐波那契（Fibonacci）数列的前n项（计算数列中某项的值请用递归函数实现）。另外，请指出所用算法的复杂度。**

**有能力的同学还可进一步改进算法的效率。**

参考答案：

def fib(n):

if n==0 or n==1:

return n

else:

return fib(n-1) + fib(n-2)

n=int(input('n='))

for i in range(n+1):

print(fib(i),end=" ")

算法时间复杂度为O(n\*2n).

可改进为：

def fib(n,List):

a,b=0,1

List.append(a)

while b<=n:

List.append(b)

a,b = b,a+b

n=int(input('n='))

L1=[]

fib(n,L1)

print(L1)

算法的时间复杂性为O(n)

**3．当前目录下有一个文件名为score2.txt的文本文件，存放着某班学生的计算机课成绩，共有学号、总评成绩两列。请查找最高分和最低分的学生，并在屏幕上显示其学号和成绩。另外，请指出所用算法的复杂度。**

参考答案：

f=open("score2.txt")

a=f.readlines()

del a[0]

L2=[]

L3=[]

for line in a:

line=line.strip()

L1=line.split()

L2.append(L1[0])

L3.append(L1[1])

f.close()

maxScore=L3[0]

maxIndex=0

minScore=L3[0]

minIndex=0

for i in range(1,len(L3)):

if L3[i]>maxScore:

maxScore=L3[i]

maxIndex=i

if L3[i]<minScore:

minScore=L3[i]

minIndex=i

print("最高分为："+str(maxScore)+"分，该学生学号为："+str(L2[maxIndex]))

print("最低分为："+str(minScore)+"分，该学生学号为："+str(L2[minIndex]))

时间复杂度为O(n).

1. **数据结构部分：**
2. **编程，输入一行字符，分别统计出其中英文字母、空格、数字和其他字符的个数。**

参考答案：

letters,space,digit,other=0,0,0,0

s=input("请输入一行字符：")

for i in range(len(s)):

if (s[i]>='a' and s[i]<='z') or (s[i]>='A' and s[i]<='Z'):

letters+=1

elif s[i]==' ':

space+=1

elif s[i]>='0' and s[i]<='9':

digit+=1

else:

other+=1

print("字母数：%d\n空格数：%d\n数字数：%d\n其他字符数：%d\n"%(letters,space,digit,other))

或

p=input('请输入一行字符:')

a,b,c,d=0,0,0,0

for i in p:

if((i<='Z' and i>='A') or (i<='z' and i>='a')):

a+=1

elif (i==' '):

b+=1

elif(i>='0' and i<='9'):

c+=1

else:

d+=1

print ('英文字母的个数为：'+str(a))

print ('空格的个数为：'+str(b))

print ('数字的个数为：'+str(c))

print ('其他字符的个数为：'+str(d))

或

letter,space,digit,other=0,0,0,0

s = input('input a string:')

for c in s:

if c.isalpha():

letter +=1

elif c.isspace():

space +=1

elif c.isdigit():

digit +=1

else:

other +=1

print("字母数：%d\n空格数：%d\n数字数：%d\n其他字符数：%d\n"%(letter,space,digit,other))

**2．小王希望用电脑记录他每天掌握的英文单词。请设计程序和相应的数据结构，使小王能记录新学的英文单词和其中文翻译，并能很方便地根据英文来查找中文。**

**（参考：数据结构建议用集合。集合添加：dic[key]=value 判断key是否在集合中：if key in dic)。**

参考答案：

def add\_dic(dic):

while True:

word=input("请输入英文单词（直接按回车结束）：")

if len(word)==0:

break;

meaning=input("请输入中文翻译：")

dic[word]=meaning

print("该单词已添加到字典库。")

return

def search\_dic(dic):

while True:

word=input("请输入要查询的英文单词（直接按回车结束）：")

if len(word)==0:

break;

if word in dic:

print("%s的中文翻译是%s"%(word,dic[word]))

else:

print("字典库中未找到这个单词")

return

worddic=dict()

while True:

print("请选择功能：\n1:输入\n2：查找\n3：退出")

c=input()

if c=="1":

add\_dic(worddic)

elif c=="2":

search\_dic(worddic)

elif c=="3":

break

else:

print("输入有误！")

**五．异常处理部分：**

**1．以下是两数相加的程序：**

**x = int(input("x="))**

**y = int(input("y="))**

**print("x+y=",x+y);**

**该程序要求接收两个整数，并输出相加结果。但如果输入的不是整数（如字母、浮点数等），程序就会终止执行并输出异常信息。请对程序进行修改，要求输入非整数时，给出“输入内容必须为整数！”的提示，并提示用户重新输入，直至输入正确。**

参考答案：

while True:

try:

x = int(input("x="))

except ValueError:

print("输入内容必须为整数！")

else:

break

while True:

try:

y = int(input("y="))

except ValueError:

print("输入内容必须为整数！")

else:

break

print("x+y=",x+y)

**2．编程，请输入一个文件路径名或文件名，查看该文件是否存在，如存在，打开文件并在屏幕上输出该文件内容；如不存在，显示“输入的文件未找到！”并要求重新输入；如文件存在但在读文件过程中发生异常，则显示“文件无法正常读出！”并要求重新输入。**

**（提示：请使用异常处理。“文件未找到”对应的异常名为：FileNotFoundError，其他异常直接用except匹配）**

参考答案：

while True:

try:

filename=input('请输入文件路径名或文件名：')

f=open(filename.strip())

print(f.read())

except FileNotFoundError:

print("输入的文件未找到！")

except:

print("文件无法正常读出！")

else:

break

f.close()

1. **函数部分：**
2. **写一判素数的函数，在主函数中输入一个整数，调用该函数进行判断并输出结果。**

参考答案：

def shushu(n):

import math

i,w=2,0

if n<=1:

w=1

while i <= int(math.sqrt(n)) and w==0:

if n%i == 0:

w = 1

break

else:

i = i+1

return w

n=int(input('n='))

if shushu(n)==0:

print(n,"是素数！")

else:

print(n,"不是素数！")

1. **当前目录下有一个文件名为score3.txt的文本文件，存放着某班学生的学号和其两门专业课的成绩。分别用函数实现以下功能：**
2. **定义函数function1，计算每个学生的平均分（取整数），并将所有学生的学号和平均分在屏幕上输出。（函数参数为要读取文件的文件名）**

def function1(flname):

#函数代码

function1("c:\\test\\score3.txt")

参考答案：

def function1(flname):

f=open(flname)

a=f.readlines()

del a[0]

L3=[]

for line in a:

line=line.strip()

L1=line.split()

avg\_score=int((int(L1[1])+int(L1[2]))/2)

L3.append([L1[0],avg\_score])

f.close()

print("学号 平均分")

for L2 in L3:

print(L2[0]+" "+str(L2[1]))

1. **定义函数calAvg()，计算某一门课程的平均分（函数参数为某门课成绩对应的列表名，返回值为该门课的平均分）**

def calAvg(L):

#函数代码

f=open("c:\\test\\score3.txt")

a=f.readlines()

del a[0]

L2=[]

L3=[]

for line in a:

line=line.strip()

L1=line.split()

L2.append(int(L1[1]))

L3.append(int(L1[2]))

f.close()

print("专业课1的总平均分为",calAvg(L2))

print("专业课2的总平均分为",calAvg(L3))

参考答案：

def calAvg(L):

sum,count=0,0

for score in L:

sum+=score

count+=1

avg\_score=int(sum/count)

return avg\_score

1. **用函数或函数的递归实现求n!的算法。（主程序已给出）。同时估计程序的复杂度。**

def fact(n):

//函数代码

n=int(input("Calculate n! Enter n="))

print(n,'!=',fact(n))

参考答案：

def fact(n):

value=1

for count in range(1,n+1):

value\*=count

return value

或

def fact(n):

if n==1:

value=1

else:

value=n\*fact(n-1)

return value

程序复杂度为O(n).

**4．分别编写求两个整数的最大公约数的函数hcf和求最小公倍数的函数lcd。主函数已给出，其从键盘接收两个整数，调用这两个函数后输出结果。**

**（提示：求最大公约数可用辗转相除法。即将大数作为被除数，小数作为除数，若二者余数不为0，则将小数作为被除数，余数作为除数，…直到余数为0。求最小公倍数则用两数的积除以最大公约数即可。）**

参考答案：

def hcf(u,v):

if v>u:

u,v=v,u

r=u%v

while r!=0:

u=v

v=r

r=u%v

return v

def lcd(u,v,h):

return u\*v/h

u=int(input("请输入第一个整数："))

v=int(input("请输入第二个整数："))

h=hcf(u,v)

print("%d和%d的最大公约数为%d："%(u,v,h))

l=lcd(u,v,h)

print("%d和%d的最小公倍数为%d："%(u,v,l))

**5．编程，统计列表中各数据的方差和标准差。主函数已给出，请编写计算方差的函数var。**

**（提示：方差的计算公式为： ∑Xi2/n-(∑Xi/n)2 ，其中，n为列表中元素个数，Xi为列表中的第i项。标准差则为方差的算术平方根。）**

参考答案：

import math

def var(L1):

s,psum=0,0

for i in range(len(L1)):

v=L1[i]

s+=v

psum+=v\*v

s=s/len(L1)

mse=psum/len(L1)-s\*s

return mse

L1=[5,3,7,8,14,9,12,6]

dx=var(L1)

print('方差为：%.2f'%dx)

mse=math.sqrt(dx)

print('标准差为：%.2f'%mse)

**6．主程序中已有一个排好序的列表，请编写函数insertList，将从键盘接收的整数按原来从小到大的排序规律插入到该列表中。**

def insertList(L1,x):

#函数代码

L1=[1,4,6,9,13,16,28,40,100]

x=int(input('请输入一个要插入的整数：'))

insertList(L1,x)

print(L1)

参考答案：

def insertList(L1,x):

if x>L1[len(L1)-1]:

L1.append(x)

return

for i in range(0,len(L1)):

if x<L1[i]:

L1.insert(i,x)

break

return

L1=[1,4,6,9,13,16,28,40,100]

x=int(input('请输入一个要插入的整数：'))

insertList(L1,x)

print(L1)