**2021春Python高级程序设计**

**一、数据类型、函数、if-else语句、for-while循环等**

1. 编写判断奇数偶数的函数isodd(x)，若x不是整数，给出提示后退出程序；如果x为奇数，返回True，如果x为偶数，返回False。

def isodd(x):

if int(x)!=x:

print("x不是整数，程序退出")

return

elif x//2 != x/2: （取整不等于除法） # x%2==1:

return True

else:

return False

print(isodd(3))

print(isodd(3.2))

print(isodd(32))

1. 输入三角形的边长和高，计算并输出三角形的面积。

import math

a=eval(input("")) input默认字符串 eval把字符串换成数字

h=eval(input(""))

s = a\*h/2

print("{:.2f}".format(s))

1. 输入两个正整数m和n，求其最大公约数和最小公倍数。

m=eval(input("请输入正整数m:”))

n=eval(input("请输入正整数n:”))

if m<n: m,n=n,m 大值给m小值给n

t=m\*n 最小公倍数等于两数的积最大公约数

r=m%n

while r!=0:

m=n

n=r

r=m%n

print("最大公约数是：{}".format(n))

print("最小公倍数是：{}".format(int(t/n)))

1. 判断101-200之间有多少个素数，并输出所有素数。

程序分析：判断素数的方法：用一个数分别去除2到sqrt(这个数)，如果能被整除，则表明此数不是素数，反之是素数。

import math

for i in range(100,200):

flag=0

for j in range(2,round(math.sqrt(i))+1):

if i%j==0:

flag=1

break

if flag==1:

continue

print(i)

print('\nSimplify the code with "else"\n')

for i in range(100,200):

for j in range(2,round(math.sqrt(i))+1):

if i%j==0:

break

else:

print(i)

for 循环正常执行结束后，else 语句里面的内容也会正常执行。

当 for 循环被 break 中断后，其后的 else 语句就不执行了。

1. 斐波那契数列

程序分析：斐波那契数列（Fibonacci sequence），从1,1开始，后面每一项等于前面两项之和。1，1，2，3，5。。。。。

# 递归实现

def Fib(n): n数列的第几个数

return 1 if n<=2 else Fib(n-1)+Fib(n-2)

print(Fib(int(input())))

# 朴素实现

target=int(input()) 输入值

res=0 累加和才用

a,b=1,1

for i in range(target-1):

a,b=b,a+b

print(a)

1. 递归求阶乘，利用递归方法求5!。

def factorial(n):

return n\*factorial(n-1) if n>1 else 1

print(factorial(5))

**二、列表、元组、字典和集合**

1. 列表元素数值分类运算

s=0

Ln=[1,2,3,'a',4,'5']

for item in Ln:

if type(item)==int: type判断元素类型

s+=item

print(s)

1. 按相反的顺序输出列表的值。

a = ['one', 'two', 'three']

print(a[::-1])

1. 用特殊符号将列表元素连接成字符串。将可迭代序列以（）间隔链接起来

L = [1,2,3,4,5]

print('##'.join(str(n) for n in L)) 先转换为字符串 然后用##隔开

1. 列表切片,lst=['hi',"program",'school','for','''chinese'''],输出"program", 'school',构成新的子列表。

lst=['hi',"program",'school','for','''chinese''']

lst\_1=lst[1:3]

print(lst\_1)

1. 在列表中给出若干字符串，计算并输出最长的字符串。

lst=['hi',"program",'school','for','''chinese''']

lst1=[]

for item in lst:

lst1.append(len(item))

p=max(lst1)

for item in lst:

if len(item)==p:

print(item,end=" ")

1. 字符串排序。 按照ASCII码排序

l=['baaa','aaab','aaba','aaaa','abaa']

l.sort()

print(l)

逆序排列l.sort(reverse=True) false正序

1. 循环输出列表

l=['moyu','niupi','xuecaibichi','shengfaji','42']

for i in range(len(l)):

print(l[i])

结果：moyu

niupi

xuecaibichi

shengfaji

42

1. 定义一个列表，存储一个班级所有学生某科目考试的成绩等级，编写程序，统计输出各级别的人数，并按级别排序输出。输出结果如下。

A:5

B:9

C:7

D:3

提示：首先定义一个空字典 d，然后采用 for 循环从 ls\_score 列表中逐一取出每个元素，即成绩等级，并判断这个元素是否在 d 中，这里使用了 d.get()函数判断字典里有没有这个值。字典的 get()函数用于从字典中获取键对应的值，如果键不在字典中，给出默认值。第 5 行，d.get(score,0)将成绩等级 score 作为键检索字典，如果该键在字典中，则返回该键的值，即该等级的次数，再加 1，表示该等级又出现了一次；如果该键不在字典中，则返回 0，再加 1，并将 1 赋值给 [score]，相当于在字典中增加一个新元素，其键为 score，值为 1。字典添加元素与修改元素的方法相同，都是 dicts[key]=value 的形式。

参考代码如下。

ls\_score=["C","A","B","A","C","D","B","C",\

"B","D","B","B","B","C","C","A",\

"A","B","D","B","C","B","C","A"]

d={}

for score in ls\_score:

d[score]=d.get(score,0)+1 没有score 返回0

l=sorted(d.keys())

for i in l:

print("{}:{}".format(i,d[i]))

1. 字典运算：value运算，将字典D = {'Sex': 'female', 'Age': 7, 'Name': 'Lily'}中的值，转换为列表输出。

D = {'Sex': 'female', 'Age': 7, 'Name': 'Lily'}

print ("字典所有值为 : ", D.values())

print ("转换为列表 : ", list(D.values()))

1. 使用input函数，输入若干单词，按字典顺序输出单词（如果某个单词出现多次，只输出一次）。集合去重

words=input("请输入若干单词，用英文逗号分割：")

word\_list=words.split(",")

aset=set() #定义空集合

for i in word\_list: # 去除重复元素

aset.add(i) #把列表取出的值添加到空集合里

word3=list(aset) # 将集合转换为列表后，排序

word3.sort()

print(word3)

**三、函数调用、闭包**

1. 利用闭包返回一个计数器函数，每次调用它返回递增正整数，如调用三次counter（）函数，返回1，2，3。

def createCounter():

a=0

def counter():

nonlocal a

a=a+1

return a

return counter

counter=createCounter()

print(“counter(),counter(),counter()=”,counter(),counter(),counter())

**四、类和对象成员方法 成员变量书上例题**

1. 设计一个StuGroup类，在该类中包括：一个数据成员score（每个学生的分数）及两个类成员变量total（总分）和count（人数）。成员方法setScore（score）和getScore（）用于设置和获得分数，成员方法sum（）用于累计总分，类方法average（）用于求平均值。交互式输入该组学生的成绩，显示该组学生的总分和平均分。

class StuGroup:

total = count = 0

def setScore(self, score): #定义成员方法

self.score = score #赋值

def getScore(self):

return self.score

def \_\_init\_\_(self, score): #构造方法 构造类 前面俩短线后面俩短线

self.setScore(score) #初始化成员方法

StuGroup.count += 1

self.sum()

u

**五、文件读写与修改 1道 with**

1. 将一个文件中的所有英文字母转换成大写，复制到另一文件中。

fi=open("ex0701.py",'r')

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

for line in fi:

line=line.upper()

fo.write(line) #往fo里逐行写line

fi.close()

fo.close()

**六、异常**

1. 编程实现索引超出范围异常IndexError类型。

try: #要执行的代码

chars=['a','b','c','d','e']

chars[4]=1

print(chars)

chars[5]='xx'

except IndexError: #捕捉异常

print("索引超过范围")

1. 文件读写异常（IOError）没有读写权限

source=target=None

try :

source=open("a.txt","r")

print(source.read(9)) #读取九个字符 指针落在第十个字符

#source.seek(0)

target=open("b.txt","w+") #

target.writelines(source.readlines())

except (FileNotFoundError,IOError):

print("没有找到文件或读写失败")

finally:

if source!=None:

source.close()

if target!=None:

target.close()

a.txt文件的内容是：

12345678

A

B

C

1. 设计一个一元二次方程类，并为这个类添加异常处理。

import math

class EquationException(Exception):#自己定义异常类 继承exception类sc

def \_\_init\_\_(self,eid,message): # 异常描述

self.eid=eid

self.message=message

class ExceptionDemo: #

def computing(self,a,b,c):

print("called computing()");

deta=b\*b-4\*a\*c

if deta<0:

raise EquationException(101,"deta值小于零")

else:

x1=(-b+math.sqrt(deta))/(2\*a)

x2=(-b-math.sqrt(deta))/(2\*a)

print("一元二次方程的根是{}，{}".format(x1,x2))

print("normal exit")

myobject=ExceptionDemo() # 功能测试

try:

myobject.computing(4,4,-2)

myobject.computing(2,3,2)

except EquationException as e:

print("Exception caught,id:{},message:{}".format(e.eid,e.message))

1. **tkinter，GUI**
2. GUI消息窗口弹出。设计GUI界面，模拟QQ登录界面，输入用户名和密码，如果输入正确，提示登陆成功；否则提示登陆失败。

import tkinter

import tkinter.messagebox

win=tkinter.Tk()

win.title('QQ')

win.geometry('300x200')

varName=tkinter.StringVar() #四种控制变量BooleanVar()布尔型，StringVar()字符串

#IntVar()整型，DonbleVar()双精度

varName.set('')

varPwd=tkinter.StringVar()

varPwd.set('')

labelName=tkinter.Label(win,text='username:',justify=tkinter.RIGHT)

labelName.place(x=50,y=30,width=80,height=20)

entryName=tkinter.Entry(win,textvariable=varName)

entryName.place(x=130,y=30,width=100,height=20)

labelPwd=tkinter.Label(win,text='password:',justify=tkinter.RIGHT)

labelPwd.place(x=50,y=70,width=80,height=20)

entryPwd=tkinter.Entry(win,show='\*',textvariable=varPwd)

entryPwd.place(x=130,y=70,width=100,height=20)

users={'zhang3':'123','admin':'123456','li4':'abc'}

def login():

name=entryName.get()

pwd=entryPwd.get()

flag=False

for item in users:

if item==name and users[item]==pwd:

flag=True

if flag==True:

tkinter.messagebox.showinfo(title='QQ',message='OK')

else:

tkinter.messagebox.showinfo(title='QQ',message='Error')

def cancel():

varName.set('')

varPwd.set('')

buttonOK=tkinter.Button(win,text='Login',command=login)

buttonOK.place(x=90,y=130,width=50,height=20)

buttonCancel=tkinter.Button(win,text='Reset',command=cancel)

buttonCancel.place(x=150,y=130,width=50,height=20)

win.mainloop()