**财经大数据A 期末考试题库**

考试时间 2022年5月12日 命题人：刘焕华

1. 创建变量revenue赋值为98765，创建变量costs并赋值为40000，计算两者的差值并存储为变量profit，打印其结果。

revenue = 98675

costs = 40000

profit = revenue - costs

print(profit)

1. 分别创建三个变量并进行赋值作为三门功课的成绩，计算总分和平均分，并打印

a = 60

b = 61

c = 62

print(a + b + c)

print((a + b + c) / 3)

1. 从键盘输入球的半径，计算球的表面积和体积并进行打印

radius = float(input("Radius: "))

area=4\*3.14\*radius\*radius

volume=4/3\*3.14\*radius\*\*3

print ( "球的表面积: %.2f"% area)

print ( "球的体积: %.2f" % volume)

1. 编写程序实现一个三位数字的反序输出（分别求出个、十、百位的数字）

number = input('请输入一个三位数：')

list1 = list(number)

list1.reverse()

a = int(list1[0])

b = int(list1[1])

c = int(list1[2])

re\_number = a\*100 + b\*10 + c

print(re\_number)

1. 利用input函数输入自己的名字，并打印出来

name = input("Name: ")

print(name)

1. 利用input函数输入自己的年龄，打印出自己的年龄+2的结果

age = int(input("Age: "))

print(age+2)

1. 利用input函数分别输入自己的名字和年龄，并打印出如下字符串：“哇塞，xx今年yy岁了！”

name = input("Name: ")

age = eval(input("Age: "))

print("哇塞，{0}今年{1}岁了！".format(name, age))

1. 利用求根公式计算方程x^2+2x-3=0的解

*import* math

x1=(-2 + math.sqrt(2\*\*2-4\*1\*-3))/(2\*1)

x2=(-2 - math.sqrt(2\*\*2-4\*1\*-3))/(2\*1)

print(x1, x2)

1. 通过键盘输入值x，计算cosx^2+sinx^2的值

*import* math

x = float(input("value: "))

print(math.ceil(math.cos(x)\*\*2 + math.sin(x)\*\*2))

1. 现有字符串： str1 = 'www.hufe.edu.cn'，根据题目要求，将截取后的新字符串赋值给str2并输出
   1. 截取字符串的第一位到第三位的字符
   2. 截取字符串最后三位的字符
   3. 截取字符串的全部字符
   4. 截取字符串的第七个字符到结尾
   5. 截取字符串的第三个字符
   6. 输出字符串的长度
   7. 将字符串中所有的"."换成"-"并输出
   8. 删除字符"."，拆分成四个字符串并输出

str1 = 'www.hufe.edu.cn'

str2 = ''

str2 = str1[:3]

print(str2)

str2 = str1[-3:]

print(str2)

str2 = str1

print(str2)

str2 = str1[6:]

print(str2)

str2 = str1[2]

print(str2)

print(len(str1))

str2 = str1.replace(".", "-")

print(str2)

str2 = str1.split(".")

*for* str *in* str2:

    print(str)

1. 按照一定格式输出（最好能够用常用方法和format方法两种实现）：
   1. 打印123456的二进制、八进制和十六进制
   2. 打印Python，要求格式：空白填充+左对齐+固定宽度20
   3. 打印70305084.0 要求格式: 千分位+空白填充+右对齐+固定宽度18+浮点精度.2+浮点数声明f

print(bin(123456))

print(oct(123456))

print(hex(123456))

print("{: <20}".format("Python"))

print("{: >18,.2f}".format(70305084.0))

1. 已知字符串 a = “aAsar3ide4bgs7Dlsf9eAF”,要求如下:
   1. 找出字符串中最先出现i字母的位置
   2. 计算a字母出现的次数
   3. 将a字符串的大写改为小写，小写改为大写。
   4. 将a字符串反转并输出

a = "aAsar3ide4bgs7Dlsf9eAF"

print(a.find("i"))

print(a.count("a"))

print(a.swapcase())

print(a[::-1])

1. 编写程序，自动生成宿舍的组合名，从键盘上分别输入宿舍人员的姓名，取其中的一个字连在一起，输出组合名

*import* random

n=int(input("请输入总人数: "))

strMix = ""

stuList = []

i=0

count = 1

*while* i<n:

    stuList.append(input("请输入第"+ str(count)+"个同学的姓名:"))

    i+=1

    count += 1

*for* stu *in* stuList:

    strMix += stu[random.randint(0,len(stu)-1)]

print(strMix)

1. 输入一个数，判断是否为偶数

num = int(input("输入一个数："))

*if* (num % 2) == 0:

    print("{0}是偶数".format(num))

*else*:

    print("{0}不是偶数".format(num))

1. 根据输入的年纪范围打印成年或者未成年，如果年龄不在正常范围内(0~150)打印这不是人!

num = int(input("输入年龄："))

*if* (num < 0 *or* num > 150):

    print("这不是人！")

*elif* (num > 18):

    print("成年")

*elif* (num < 18):

    print("未成年")

1. 输入一个四位整数（年份），判断是不是闰年

year = int(input("输入一个年份: "))

*if* (year % 4) == 0:

*if* (year % 100) == 0:

*if* (year % 400) == 0:

           print("{0}年是闰年".format(year))

*else*:

           print("{0}年不是闰年".format(year))

*else*:

       print("{0}年是闰年".format(year))

*else*:

   print("{0}年不是闰年".format(year))

1. 输入一个数判断是否同时满足被7整除且最后一位数字为3

number = int(input('请输入一个数：'))

list1 = list(str(number))

list1.reverse()

a = int(list1[0])

*if* (a == 3 *and* number % 7 == 0):

    print("满足被7整除且最后一位数字为3")

*else*:

    print("不满足被7整除且最后一位数字为3")

1. 密码安全测试：输入8位密码，必须包含英文大小写和数字的组合，根据输入的密码打印提示信息。如：“密码强度过低，没有输入英文大小写”、“请输入8位”等

password = input("输入密码: ")

error = "密码强度过低"

raiseerr = False

*if* len(password) < 7:

    error += "，长度不足8位"

    raiseerr = True

*if* password.islower():

    error += "，未包含英文大写"

    raiseerr = True

*if* password.isupper():

    error += "，未包含英文小写"

    raiseerr = True

*if* password.isdigit():

    error += "，未包含英文字母"

    raiseerr = True

*if* password.isalpha():

    error += "，未包含数字"

    raiseerr = True

*if* raiseerr:

    print(error)

*else*:

    print("密码OK")

1. 计算100以内的被3和5整除的数之和（分别用for和while循环）

i=1

sum=0

*for* num *in* range(1, 101):

*if* num % 3 ==0:

        sum += num

*while* i<=100:

*if* i % 5 == 0:

        sum += i

    i += 1

print(sum)

1. 输出26个英文字母

*for* code *in* range(97, 123):

    print(chr(code), *end*=" ")

1. 有0，2，3，4四个数字，求这四个数字能生成多少无重复数字的三位数

*from* random *import* randrange

count = 0

randrange = [0, 2, 3, 4]

*for* i *in* randrange:

*for* j *in* randrange:

*for* k *in* randrange:

*if* i != j *and* j != k *and* k != i:

                count +=1

print("可生成%d个无重复数字的三位数" % count)

1. 使用input输入n，通过键盘输入n个整数求和。

count = int(input("几个整数？"))

i = 1

sum = 0

*while* i < count+1:

    sum += int(input("整数{0}: ".format(str(i))))

    i += 1

print(sum)

1. 输出1-200的最大的质数。

*for* n *in* range(199,100,-2):

*for* i *in* range(3,int(n\*\*(1/2))+1,2):

*if* n % i == 0:

*break*

*else*:

        print(f'200以内最大素数是{n}')

*break*

1. 输出9\*9乘法口诀。

i = 1

j = 1

s = 0

*for* i *in* range(1, 10):

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

        s = i \* j

        print("%d \* %d = %s" % (i, j, s), *end* = "\t")

        j += 1

    print(" "\*30)

    i+=1

1. 编写程序，自动生成宿舍的组合名，从键盘上分别输入宿舍人员的姓名，随机选取其中的一个字连在一起，输出组合名。

*import* random

n=int(input("请输入总人数: "))

strMix = ""

stuList = []

i=0

count = 1

*while* i<n:

    stuList.append(input("请输入第"+ str(count)+"个同学的姓名:"))

    i+=1

    count += 1

*for* stu *in* stuList:

    strMix += stu[random.randint(0,len(stu)-1)]

print(strMix)

1. 随机生成一个100-1000的整数n，计算1-n之间所有偶数之和。

*import* random

n = random.randint(100, 1000)

num = 1

sum = 0

*while* num < n+1:

*if* num % 2 == 0:

        sum += num

    num += 1

print(sum)

1. 随机生成100个随机数，求其最大值。

*import* random

numbers = [random.randint(1, 100) *for* i *in* range(100)]

max = numbers[0]

*for* i *in* numbers:

*if* i > max:

        max = i

print(max)

1. 假设有三个列表：lst\_who=['我','你','他'] 、 lst\_where=['在宿舍','在教室','在操场']和lst\_what=['学习','打游戏','睡觉']。编写程序进行造句，随机生成三个0-2范围内的整数，将其作为索引访问对应的列表元素，连接起来进行造句。如，随机生成三个数为1，0，2，则输出句子''你在宿舍睡觉''。

*import* random

lst\_who = ['我','你','他']

lst\_where = ['在宿舍','在教室','在操场']

lst\_what = ['学习','打游戏','睡觉']

print("{0}{1}{2}".format(lst\_who[random.randint(0,2)], lst\_where[random.randint(0,2)], lst\_what[random.randint(0,2)]))

1. 调用列表操作的常用函数实现以下功能：
   1. 创建一个列表 score =[68,87,92,100,76,88,54,89,76,61]；
   2. 输出 score 列表中后5 个元素的数值；
   3. 输出 score 列表中第 1~6 个元素的值；
   4. 调用 insert()函数，在 score 列表第 3 个元素之前添加数 59；
   5. 利用变量 num 保存数值 76,调用 count()函数，查询 num 变量值在 score 列表中出现的次数；
   6. 调用 sort()函数，对列表中所有元素进行排序，输出考试的最高分和最低分；
   7. 查找元素100在列表的位置
   8. 求列表所有元素的平均值

score = [68,87,92,100,76,88,54,89,76,61]

print(score[-5:])

print(score[:6])

score.insert(2, 59)

num = 76

print(score.count(num))

score.sort(*reverse*=True)

print(score[0])

score.sort()

print(score[0])

print(score.index(100))

print(sum(score) / len(score))

1. 创建一个 english 列表（列表推导式），有 5 个元素均为 40—100 之间的随机数，并对列表降序排列

*import* random

english = [random.randint(40, 100) *for* i *in* range(5)]

english.sort(*reverse*=True)

print(english)

1. 有两个列表 l1=[11,22,33,55] 和 l2=[22,33,44,66,77]
   1. 输出内容相同的元素列表
   2. 输出l1中有l2中没有元素列表
   3. 合并列表，重复的元素只能出现一次

l1 = [11,22,33,55]

l2 = [22,33,44,66,77]

a = []

*for* i1 *in* l1:

*for* i2 *in* l2:

*if* i1 == i2:

            a.append(i1)

print(a)

a = []

*for* i1 *in* l1:

*if* i1 *not* *in* l2:

        a.append(i1)

print(a)

a = []

a = l1 + l2

a = list(set(a))

a.sort()

print(a)

1. 假设有列表student=[['001','李梅',19],['002','韩磊磊',20],['003','张亮',18]]依次存放了每名学生的学号、姓名和年龄，编写程序完成如下功能：
   1. 添加学生信息['004','王大锤',20]和['006','刘大刀',18]
   2. 2 在列表适当位置添加学生信息['005','赵钱孙',20]
   3. 输出学号为004的学生信息
   4. 输出所有学生的姓名
   5. 输出年龄大于19的所有学生的信息
   6. 将字符串s = "I want to study Python perfectly"转换为元组,并按照5个一行的格式输出

student = [['001','李梅',19],['002','韩磊磊',20],['003','张亮',18]]

student.append(['004','王大锤',20])

student.append(['006','刘大刀',18])

student.insert(4, ['005','赵钱孙',20])

*for* stu *in* student:

*if* "004" *in* stu:

        print(stu)

*for* stu *in* student:

    print(stu[1])

*for* stu *in* student:

*if* stu[2] > 19:

        print(stu)

s = "I want to study Python perfectly"

s = tuple(s)

*for* i *in* range(len(s)):

    print(s[i], *end*=' ')

*if* (i+1) % 5 == 0:

        print("")

1. 某餐厅推出了下午茶套餐活动，可以从指定的饮料和糕点中各选一款作为套餐，糕点包括提拉米苏（15）、芝士蛋糕（20)和三明治(10)，饮料包括红茶(6)、咖啡(12)和橙汁(8)，套餐的价格定位单价的两个之和乘以0.8取整，编程输出所有的套餐详情。提示：输出一个列表，每一个元素是一个元组，如[('提拉米苏','红茶',17)]，单价商品也可以存成一个列表，每一个元素是一个元组，如[('提拉米苏',17)]

snacks = {'提拉米苏': 15, '芝士蛋糕': 20, '三明治': 10}

drinks = {'红茶': 6, '咖啡': 12, '橙汁': 8}

menus = []

*for* snack *in* snacks:

*for* drink *in* drinks:

        menu = (snack,drink, int((snacks[snack] + drinks[drink]) \* 0.8))

        menus.append(menu)

*for* menu *in* menus:

        print(menu)

1. 有如下字典，按照要求实现每一个功能dict = {"k1":"v1","k2":"v2","k3":"v3"}
   1. 请循环遍历出所有的key
   2. 请循环遍历出所有的value
   3. 请循环遍历出所有的key和value
   4. 请在字典中增加一个键值对,"k4":"v4"，输出添加后的字典
   5. 请删除字典中键值对"k1":"v1",并输出删除后的结果

dict = {"k1":"v1","k2":"v2","k3":"v3"}

*for* key *in* dict:

    print(key)

*for* value *in* dict.values():

    print(value)

*for* item *in* dict.items():

    print(item)

dict.update({'k4':'v4'})

print(dict)

dict.pop("k1")

print(dict)

1. 以下是某电商家在销售产品的价目一览表。 表如下 产品名称 价格 辣条 9.9 快乐肥宅水 99 盲盒 199 毛拖 299 试编写程序，实现以下功能：
   * + 1. 使用字典myDict存放上表的信息，产品名称为键，价格为值。
       2. 输出所有在售产品的价目表。格式为 方糖.........99元 （其他的格式如上）
       3. 输出价格超过100的所有商品

*from* re *import* I

myDict = {"辣条": 9.9, "快乐肥宅水": 99, "盲盒": 199, "毛拖": 299}

*for* item *in* myDict:

    print("{0}.........{1}".format(item, myDict[item]))

*for* item, price *in* myDict.items():

*if* price > 100:

        print(item)

1. 已知有一个包含一些同学成绩的字典，现在需要计算所有成绩的最高分、最低分、平均分，并查找所有最高分同学：

scores = {"Zhang San": 45, "Li Si": 78, "Wang Wu": 40, "Zhou Liu": 96,"Zhao Qi": 65, "Sun Ba": 90, "Zheng Jiu": 78, "Wu Shi": 99,"Dong Shiyi": 60}

scores = {"Zhang San": 45, "Li Si": 78, "Wang Wu": 40, "Zhou Liu": 96, "Zhao Qi": 65, "Sun Ba": 90, "Zheng Jiu": 78, "Wu Shi": 99, "Dong Shiyi": 60}

max = 0

min = 100

avrg = 0

namelst = []

*for* grade *in* scores.values():

*if* grade > max:

        max = grade

*if* grade < min:

        min = grade

    avrg += grade

avrg /= len(scores)

print("最高分是{0}，最低分是{1}，平均分是{2}".format(max, min, avrg))

*for* name, score *in* scores.items():

*if* score == max:

        namelst.append(name)

print("{0}获得最高分".format(namelst))

1. 创建如下字典，键值是姓名，对应的值分别是数学、Python和英语的成绩scores = {"Zhang San": [45,60,80] "Li Si": [78,80,90], "Wang Wu": [40,59,60]}
   * 1. 基于上述字典创建一个平均分的字典
     2. 输出所有不及格的信息，如“数学 Zhang San、Wang Wu不及格
     3. 按照数学分数排序输出相应信息

scores = {"Zhang San": [45,60,80], "Li Si": [78,80,90], "Wang Wu": [40,59,60]}

averScores = {key:sum(scores[key])/len(scores.values()) *for* key *in* scores}

print("平均分字典: {}".format(averScores))

failMath = []

failPython = []

failEnglish = []

*for* name,grades *in* scores.items():

*for* i *in* range(3):

*if* grades[i] < 60:

*if* i == 0:

                failMath.append(name)

*elif* i == 1:

                failPython.append(name)

*elif* i == 2:

                failEnglish.append(name)

print("数学不及格: {}".format(",".join(failMath) *if* len(failMath) > 0 *else* "无"))

print("Python不及格: {}".format(",".join(failPython) *if* len(failPython) > 0 *else* "无"))

print("英语不及格: {}".format(",".join(failEnglish) *if* len(failEnglish) > 0 *else* "无"))

print("按照数学分数由低到高排序，输出信息")

scoresSorted = sorted(scores.items(),*key*=lambda *x*:*x*[1][0])

*for* name, grade *in* scoresSorted:

    print(name,grade)

1. 用列表推导式生成20个0-20之间的随机数的列表，删除重复的数字并输出结果。

*import* random

lst = [random.randint(0, 20) *for* i *in* range(20)]

lst = list(set(lst))

print(lst)

1. 创建连个集合s1={1,2,3,4},s2={2,3,5,6},求两个集合的交集、并集、差集和对称差集。

s1 = {1,2,3,4}

s2 = {2,3,5,6}

print(s1 & s2)

print(s1 | s2)

print(s1 - s2)

print(s1 ^ s2)

1. 编写isOdd（）函数，有一个整数参数，如果是奇数则返回True，否则返回False，并进行测试。

def isOdd(*x*):

*if* *x* % 2 == 0: *return* print(False)

*else*: *return* print(True)

x = eval(input("请输入一个整数："))

isOdd(x)

1. 编写一个函数，求一个正整数n的各位数字之和，并在主程序中进行测试。

def func(*n*):

    ls = list(map(int, str(*n*)))

*return* sum(ls)

count = 0

n = eval(input('输入一个正整数：'))

print(f'{n}各位数之和为：{func(n)}')

1. 编写函数求1~n之间能被7整除，但不能同时被5整除的所有整数。

def func(*n*):

*for* i *in* range(1, *n*):

*if* i % 7 == 0 *and* i % 5 != 0:

            print(i)

x = eval(input("请输入一个整数："))

func(x)

1. 编写一个函数，接收一个列表参数，函数返回该列表中所有正数之和，在主程序中进行测试。

def func(*ls*):

    \_ls = filter(lambda *n*: *n* > 0, *ls*)

*return* sum(\_ls)

ls = eval(input('请按照列表格式输入列表元素：'))

print(f'该列表中所有正数之和为：{func(ls)}')

1. 编写函数，接收任意多个实数，返回一个元组，其中第一个元素为所有参数的总和，第二个元素是平均值。

def receive(*lst*):

    li=[]

    li.append(sum(*lst*))

    avrg = sum(*lst*)/len(*lst*)

    li.append(avrg)

*return* li

lst = list(map(int,input().strip().split(" ")))

tp = tuple(receive(lst))

print(tp)

1. 编写可变长度参数的函数，输出字典的键值。

def func(\*\**nums*):

*for* item *in* *nums*.items():

        print(item)

func(*a*=11, *b*=22, *c*=33)

1. 编写函数，接收字符串参数，返回一个元组，其中第一个元素为大写字母个数，第二个元素为小写字母个数。

def func(*str*):

    upperct = 0

    lowerct = 0

*for* i *in* *str*:

*if* i.islower():

            lowerct += 1

*elif* i.isupper():

            upperct += 1

*return* (upperct, lowerct)

print(func(input("请输⼊⼀串字符串：")))

1. 编写函数，参数为a和n，求s=a+aa+aaa+aaaa+aa...a的值，其中a是一个数字。例如2+22+222+2222+22222(n=5)。

def sum(*a*, *n*):

    sn = 0

    sum = 0

*for* i *in* range(1, *n*+1):

        sn = sn \* 10 + *a*

        sum += sn

*return* sum

a = eval(input('请输入a：'))

n = eval(input('请输入n：'))

result = sum(a, n)

print(result)

1. 定义Lambda函数求一个数的平方，然后调用该函数求一个列表所有元素的平方之和。

numbers = list(range(10))

sum = 0

func = lambda *x*: *x* \*\* 2

*for* num *in* numbers:

   sum += func(num)

print(sum)