## Python课后题答案

### Python概述

#### 编程题。

1． 代码如下：

print("+++++++++++")  
 print("+ +")  
 print("+++++++++++")

2. 代码如下：

print("学Python，来传智播客黑马程序员")  
 print("学Python，来传智播客黑马程序员")  
 print("学Python，来传智播客黑马程序员")  
 print("学Python，来传智播客黑马程序员")  
 print("学Python，来传智播客黑马程序员")

### Python基本语法

#### 编程题

1. 代码如下：

import math  
 a=float(input("请输入斜边 1 的长度")) #输入实数  
 b=float(input("请输入斜边 2 的长度")) #输入实数  
 c=a\*a+b\*b #计算,得到的是斜边的平方  
 c=sqrt(c) #开方，得到的是斜边长  
 print("斜边长为:",c) #显示，一项是字符串，一项是 c 表示的斜边长

2. 代码如下：

# 用户输入

x = input('输入 x 值: ')

y = input('输入 y 值: ')

# 不使用临时变量

x,y = y,x

print('交换后 x 的值为: {}'.format(x))

print('交换后 y 的值为: {}'.format(y))

### Python常用语句

#### 编程题

1. 代码如下：

for i in range(1,11):  
 print(i)

1. 代码如下：

a=int(input("请输入一个数："))  
 if a>0:  
 print("a是一个正数")  
 elif a<0:  
 print("a是一个负数")  
 else:  
 print("a等于0")

1. 代码如下：

i=1  
 while i<10:  
 j=1  
 while j<=i:  
 print("%d\*%d=%-2d "%(i,j,i\*j),end='')  
 j+=1  
 print("\n")  
 i+=1

### 字符串

#### 程序分析题

1. 答：不能编译通过。因为num1和num2属于字符串类型，不能执行取余操作。
2. 结果为：J
3. 答：由于没有在字符串中找到子串，index方法默认会抛出ValueError异常。
4. 结果为：Hello World（默认删除两头的空格）
5. 结果为：HelloPython

#### 编程题

1. 答案：

myStr = input("请输入任意字符串:")

num = 0

for s in myStr:

if s.isdigit():

num += 1

print(num)

1. 答案：

week = ['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday']

while True:

myStr = input("请输入任意一个字母:")

upMyStr = myStr.upper()

for weekS in week:

if upMyStr == weekS[0]:

if weekS.startswith("M"):

print("星期一")

elif weekS.startswith("W"):

print("星期三")

elif weekS.startswith("F"):

print("星期五")

elif weekS.startswith("T") or weekS.startswith("S"):

secondStr = input("请再输入任意一个字母:")

newStr = upMyStr+secondStr

print(newStr)

for weekStr in week:

if weekStr.find(newStr,0,2) != -1:

if newStr == "Tu":

print("星期二")

elif newStr == "Th":

print("星期四")

elif newStr == "Sa":

print("星期六")

elif newStr == "Su":

print("星期日")

break

break

1. 答案：

str = "Hello,welcome to itheima!!"

print(str.count("o"))

### 列表、元组和字典

#### 程序分析题

1. 答案：程序运行错误，元组不能使用下标增加元素。
2. 结果为：3
3. 结果为：1，26

#### 编程题

1. 答案：

def message():  
 myStr = input("请输入密码，只能为字母和数字：")  
 if myStr.isalnum():  
 print("符合要求")  
 else:  
 print("不符合要求")

1. 答案：

arr = []

evenStr = ""

oddStr = ""

message = input("请输入任意字符串：")

for string in message:

arr.append(string)

for eStr in (arr[::2]):

evenStr = evenStr+eStr

for oStr in (arr[1::2]):

oddStr = oddStr+oStr

print(evenStr+oddStr)

1. 答案：

arr = []

def count():

myStr = input("请输入只包含字母的字符串：")

if myStr.isalpha():

newStr = myStr.lower()

for string in newStr:

arr.append(string)

a = {}

for i in arr:

if arr.count(i) >= 1:

a[i] = arr.count(i)

print(a)

else:

print("输入的内容有误")

count()

1. 答案：

arr = []

result = ""

myStr = input("请输入字符串：")

for string in myStr:

arr.append(string)

last = arr[-1]

arr.remove(last)

arr.insert(0,last)

for new in arr:

result = result + new

print(result)

1. 答案：

arr = []

length = int(input("请要输入数字的总个数（必须为奇数）:"))

i = 0

while i < length:

num = int(input("输入第%d个数字:"%(i+1)))

arr.append(num)

i+=1

arr.sort()

index = int(length/2)

print(arr[index])

1. 答案：

第一种：

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

arr.reverse()

print(arr)

第二种：

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

arr.sort(reverse=True)

print(arr)

1. 答案：

arr = []

length = int(input("请输入数字的总个数:"))

i = 0

while i < length:

num = int(input("输入第%d个数字:"%(i+1)))

arr.append(num)

i+=1

print("排序前：%s"%arr)

# 对列表排序

for i in range(length):

flag = 0

for j in range(1,length):

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

arr[j-1],arr[j] = arr[j],arr[j-1]

flag = 1

if not flag:

break

print("排序后：%s"%arr)

1. 答案：

arr = []

length = int(input("请输入列表的总个数:"))

i = 0

while i < length:

element = input("输入第%d个元素:"%(i+1))

arr.append(element)

i+=1

# 列表转为集合

newList = set(arr)

print(newList)

1. 答案：

def add(a, b):

# 下面两行保证输入的a和b是维数相同的矩阵，根据实际情况不要也可以

assert (len(a) == len(b))

assert (all([len(a[i]) == len(b[i]) == len(a[0]) for i in range(len(a))]))

i, j = len(a), len(a[0])

c = [[0] \* j] \* i

for m in range(i):

for n in range(j):

c[m][n] = a[m][n] + b[m][n]

return c

a = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]

b = [[9, 8, 7], [6, 5, 4], [3, 2, 1]]

c = add(a, b)

print(c)

1. 答案：

dic = {}

i=0

while i<3:

number = input("输入学生学号：")

name = input("输入学生姓名：")

dic.\_\_setitem\_\_(number,name)

i+=1

print("排序前：%s"%dic)

def dict2list(dic:dict):

''' 将字典转化为列表 '''

keys = dic.keys()

vals = dic.values()

lst = [(key, val) for key, val in zip(keys, vals)]

return lst

new = sorted(dict2list(dic), key=lambda x:x[0], reverse=False)

print("排序后：%s"%new)

1. 答案：

def delMale(dict):

keys = []

values = []

for (key,value) in dict.items():

keys.append(key)

values.append(value)

for value in values:

if value == 0:

index = values.index(value)

arrKey = keys[index]

del dict[arrKey]

values[index] = 3

keys[index] = "占位"

dic = {"小明":0, "小红":1,"小兰":1,"小白":0}

print("删除前：%s"%dic)

delMale(dic)

print("删除后：%s"%dic)

1. 答案：

arr = [12,3,37,7,91,67,27,45,6]

def delPrime(arr):

for element in arr:

# 质数大于 1

if element > 1:

# 查看因子

for i in range(2, element):

if (element % i) == 0:

break

else:

arr.remove(element)

delPrime(arr)

print(arr)

### 函数

#### 程序分析题

1．结果为100

2．结果为：200

3．结果为：200，100

#### 编程题

1. 答案：

def getMax():

num1 = int(input("请输入第1个整数："))

num2 = int(input("请输入第2个整数："))

num3 = int(input("请输入第3个整数："))

temp = 0 # 临时变量，存放中间产生的第2大数值

if num1 > num2:

temp = num1

else:

temp = num2

if temp>num3:

return "其中最大值为："+ str(temp)

else:

return "其中最大值为："+ str(num3)

maxValue = getMax()

print(maxValue)

1. 答案：

def sum(n):

res = 0

while n >= 1:

if n%2 == 0:

res -= 1.0/(n\*(n+1))

else:

res += 1.0/(n\*(n+1))

n-=1

return res

num = int(input("请输入一个整数："))

print(sum(num))

1. 答案：

def is\_palindrome(n):

n=str(n)

m=n[::-1]

return n==m

result = is\_palindrome(1247321)

print(result)

1. 答案

def sjx(a, b, c):

if (a + b > c and a + c > b and b + c > a):

return "能构成三角形"

else:

return "不能构成三角形"

result1 = sjx(3, 4, 5)

print(result1)

result2 = sjx(1, 1, 2)

print(result2)

1. 答案：

# 定义函数

def lcm(x, y):

# 获取最大的数

if x > y:

greater = x

else:

greater = y

while(True):

if((greater % x == 0) and (greater % y == 0)):

lcm = greater

break

greater += 1

return lcm

# 获取用户输入

num1 = int(input("输入第一个数字: "))

num2 = int(input("输入第二个数字: "))

print( num1,"和", num2,"的最小公倍数为", lcm(num1, num2))

### 高级函数

#### 程序分析题

1．能，结果为：

6

7

8

2．能，结果为：

6

6

6

#### 编程题

答案：

result = filter(lambda x:(x+1)%2, [1, 2, 3, 4, 5])

print(result)

### 异常

#### 编程题

1．答案：

class HeightException(Exception):

pass

try:

height = int(input("请输入您的身高："))

if height<30 or height>250:

raise HeightException

weight = int(input("请输入您的体重："))

weightS = height-100

if weight > weightS and weight-weightS < 0.05\*weightS:

print("体重达标")

elif weight < weightS and weightS-weight < 0.05\*weightS:

print("体重达标")

else:

print("体重不达标")

except HeightException:

print("您输入的身高有误")

2．答案：

try:

score = int(input("请输入学生的成绩："))

if score>=90 and score<=100:

print("A:优秀")

elif score>=80 and score<90:

print("B:良好")

elif score>=60 and score<80:

print("C:合格")

else:

assert score>60,"D:不及格"

except Exception as result:

print("低于60分：\n",result)

### 面向对象编程（上）

#### 程序分析题

1. 能，结果为：我的名字是小明
2. 能，结果为：

--del--

--end--

#### 编程题

1. 答案：

class Circle:

def \_\_init\_\_(self,tup, radius, color):

self.center = tup

self.radius = radius

self.color = color

def perimeter(self):

return 3.14 \* 2 \* self.radius

def area(self):

return 3.14 \* self.radius \* self.radius

circle = Circle((0,0),5,"蓝色")

print(circle.perimeter())

print(circle.area())

1. 答案：

class Curriculum:  
 def \_\_init\_\_(self):  
 self.number = 1001  
 self.name = "语文"  
 self.teacher = "小明"  
 self.\_\_address = "2号教学楼3层305室"  
 def \_\_str\_\_(self):  
 return """  
 课程编号：%d  
 课程名称：%s  
 任课教师：%s  
 上课地点：%s  
 """%(self.number,self.name,self.teacher,self.\_\_address)  
curri = Curriculum()  
print(curri)

### 面向对象编程（下）

#### 编程题

1. 答案：

class Student(object):

# 构造方法

def \_\_init\_\_(self, name, age, scores):

self.\_\_name = name # 姓名

self.\_\_age = age # 年龄

self.\_\_scores = scores # 分数

def get\_name(self):

return self.\_\_name

def get\_age(self):

return self.\_\_age

def get\_course(self):

return max(self.\_\_scores)

stu = Student('小丸子', 18, [89, 90, 91])

print("姓名：%s"%(stu.get\_name()))

print("年龄：%s"%(stu.get\_age()))

print("最高分：%s"%(stu.get\_course()))

1. 答案：

class Animal(object):

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

self.color = color #颜色

def call(self):

print("动物叫")

class Fish(Animal):

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

super().\_\_init\_\_(color)

self.tail = True

def call(self):

print("-%s的鱼在吐泡泡-"%self.color)

fish = Fish("蓝色")

fish.call()