**第一题：格式化输出，这题直接运行记熟用法就好**

# 第一题

a = 1.23456

b = 123

# 利用format输出a和b

print("{} {}".format(a, b))

# 利用format保留小数点

print("{:.3f}".format(a))

# 利用format右对齐10个单位, (居中是^, 向左是<)

print("{:>10}, {:<10}, {:^10}".format(a, a, a))

# 利用format填充字符

print("{:#>10}".format(a))

# 利用format调整输出顺序

print("{0}, {1}".format(a, b))

print("{1}, {0}".format(a, b))

# 利用sep控制分隔符

print(a, b, sep='%')

# 利用end控制结尾符

print("emmm", end="$")

**第二题：注意for循环的局部变量，写出循环次数还有i的最终值**

# 第二题

for i in range(10):

i += 2

# 执行10次，i=11

# 说明执行循环多少次，并写出i最终的值

**第三题：字符串的“运算符”**

# 第三题

s = '456'

print(s \* 3) # 456456456

print(s + '123') # 456123

**第四题：字符串的下标范围，是 *-长度 ~ 长度-1***

# 第四题

s = 'abcde'

print(s[4]) # e

print(s[-5]) # a

**第五题：\*\*幂运算符，就是xx次方**

# 第五题

num = 2

print(num \*\* 3) # 8

for i in range(5):

print(i \*\* i) # 1 1 4 27 256

**第六题：了解%运算，跟C语言有点不同，python支持浮点数运算**

# 第六题

a = 5.5 % 2 # 1.5

b = 10 % 3 # 1

c = 2.2 % 2.2 # 0.0

# 输出abc的值

**第七题：ASCII与字符之间的转换**

# 第七题

# 将字符串字母全部往后移动一位，

s1 = 'abcde'

for i in s1:

print(chr(ord(i) + 1), end='') # bcdef

print() # 我是一个可爱的换行

# 输出字符串中英文在字母表中的序数

s2 = 'wadawdawo'

for i in s2:

print(ord(i) - 96, end=' ')

**第八题：三目运算符**

# 第八题

a = 1

b = 2

num1 = a if a > b else b

print(num1) # 2

num2 = a + b if 12345 < 54321 else 666

print(num2) # 3

**第九题：切片操作**

# 第九题

s = 'abcdefg'

print(s[2:]) # cdefg

print(s[3:6]) # def

print(s[::2]) # aceg

print(s[-2:2]) # 空

print(s[-4:4]) # d

**第十题：for生成列表的运用**

# 第十题

l1 = [i for i in range(5)]

print(l1) # [0, 1, 2, 3, 4]

l2 = [i \*\* i for i in range(5) if i % 2 == 0]

print(l2) # [1, 4, 256]

**第十一题：列表的“运算符”**

# 第十一题

a = [1, 2, 3]

b = [4, 5, 6]

print(a + b) # [1, 2, 3, 4, 5, 6]

print(a \* 3) # [1, 2, 3, 1, 2, 3, 1, 2, 3]

**第十二题：数学运算问题**

# 第十二题

print(5 / 2) # 2.5

print(5 // 2) # 2

print(2.2 - 1.0 == 1.2) # False, 因为精度不够准确

print(100 - 25 \* 3 % 4) # 97

**第十三题：for的break 还有 continue**

# 第十三题

num = 1

for i in range(10):

if i == 2:

num += 1

break

else:

num += 2

print(num) # 6

for i in range(1, 10):

if i % 2 == 0:

continue

print(i, end=' ') # 1 3 5 7 9

**第十四题：统计数字出现次数**

# 第十四题

# 统计1-10数字的出现次数

num = list(map(int, input().split()))

count = [0] \* 11

for i in num:

count[i] += 1 # 列表对应下标的值加一，完成统计次数目的

# 这样输出有瑕疵，会重复输出

# for i in num:

# print("{} 出现的次数是: {}".format(i, count[i]))

# 我们num转换成集合，即可去重

for i in set(num):

print("{} 出现的次数是: {}".format(i, count[i]))

**以下两题似乎没讲过，应该不会考，简单看一看得了，不懂也没关系**

***第十五题：局部变量相关问题***

# 第十五题

num = 666

def bb(num):

num = 233

print(num) # 666

bb(num)

print(num) # 666

# 可以看出，bb函数里面的num是局部变量，并不会影响到函数外部的num

def cc():

global num

num = 233

cc()

print(num) # 233

# 利用global声明是全局变量中的num，以此改变num的值

***第十六题：二维数组（矩阵的生成）***

# 第十六题

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

print(a) # [[1, 2, 3], [4, 5, 6], [7, 8, 9]]

for i in a:

print(i)

'''

[1, 2, 3]

[4, 5, 6]

[7, 8, 9]

'''

# 假设我要生成一个3 \* 3矩阵，里面元素都是0

# 不过不推荐这样生成，赋值会麻烦，因为只是浅拷贝，赋值会重复赋值

# 推荐下面那样用for生成

arr1 = [[0] \* 3] \* 3

print(arr1) # [[0, 0, 0], [0, 0, 0], [0, 0, 0]]

# 假设我们要生成一个4行3列的矩阵，矩阵的值为行索引+列索引

arr2 = [[0 for i in range(3)] for j in range(4)]

for i in range(4):

for j in range(3):

arr2[i][j] = i + j

print(arr2) # [[0, 1, 2], [1, 2, 3], [2, 3, 4], [3, 4, 5]]

# 打印二维数组

for i in range(len(arr2)):

for j in range(len(arr2[0])):

print(arr2[i][j], end=' ')

print()

# 3 \* 3矩阵以次对角线为轴交换值

arr3 = [[0 for i in range(3)] for j in range(3)]

for i in range(3):

for j in range(3):

arr3[i][j] = i + j

print(arr3) # [[0, 1, 2], [1, 2, 3], [2, 3, 4]]

for i in range(3):

for j in range(3):

arr3[3 - i - 1][3 - j - 1] = i + j

print(arr3)