Python

#!/usr/bin/env python 支持英文

# coding=utf-8 支持中文

10/3 答案为整数

10/3.0 小数

2\*\*3 2的3次方

10//3.0 3.0 一位精度

2<<3 0010向左移动3位 10000=16

2>>3 0010 向右移动3位 0010 》 0

3&2 0101 0010 > 表示且 =0 相同为1，不同为0

3| 2 或 有1为1

3^2 异或 不同为1

1000个数求和 seq -s + 1 99 man seq 把、/n 转化为 +

Rang (10)

For i in rang(10)

Break 退出整个循环

Continue 退出当前循环

Li = [1,2,45] 列表

Li[0]=1 li=[1,2,3,[‘a’,’b’],[2]] li[3][0]=a

Li.append 在末尾添加

Li.extend （[1,3,4,]) 在末尾扩展

Li.insert (1,100) 插入 在1的位置处插入100

Li.insert(-1,1000) 从后面插入

Li.insert(len(li),1000) 末尾插入

LI.POP（） 删除最后一个

a = li.pop( )

Li.count(‘a’) 计数

Ind = li.index(100) 查看所在的位置

Li.sort 排序

Li.reserse 倒叙

Li[100,3,4,55,9,7]

Li[2:4] [4,55]

Li[2:4:100] 在2位置和4位置之间 每各100去个数

Li[::-1] 倒叙取

Li[4:2:-1] 4位置和2位置之间每隔1个去一个

Li[1::2] 从1取到最后 每个1个

S = {1,3,4,6,6}

Type(s) set 类型

S.add 添加

S.remove 删除 s.pop() 删除第0个位置

S. Discard(3) 删除

S.clear 清空 s1 = {2,4,6,7} s2={6,7,8,90}

S1.difference(s2)

S1.difference.update(s2) 更新 去掉相同的

S1.intersection(s2) 求交集

S&S1 并集 S | s1 求交集

S.isdisjoint(s1) 判断TRUE（相同） 或者 FALSE （不同）

字典 d = { } type =dict

d = {‘name’:’tome’,’age’:10}

d2 = {‘name’:’tom’,(1,2):’jerry’}

d2.keys() [name,(1,2)]

Keys 0r values

It = d2.iterkeys()

It.next() 会出现keys,直到结束

另一中形式 for i in li: print i

a,b,\_ = (1,2,3)

For i in d.items():

Print i keys 和 values 都会打印出来

For k,v in d.items():

Print k,v keys 和 values 都会打印出来

D.get(“name”) 打印values

D.get(“phone”,100) 如果没有会添加keys 和 values

d[‘phone’]=11111 添加

D.has\_keys(“name”) 返回FALSE(没有定义keys) 或 true (有keys)

d4 = d2.copy() 复制 id 不一样

d4 =d3 复制 id 一样

Li = {1,2,3,4}

让每一个数加1 j = 0

for i in li:

Li[j] = i+1

J +=1

Li = (2,3,4,5)

Li2 = [i for i in li] id 一样

加 1 循环 li2 = [i+1 for i in li]

或者 for i in li:

Li3.append(i+1)

Li3 显示加1

定义 it = (i+1 for i in li)

Li2 = (i+1 for i in li)

Type (it)

T ype(li2) 显示加1

Lli2 = [x for li in li x % 2 ==0 and x>4] 判断条件