一、简答题（4分/题\*5=20分）

1）a=29.765789 格式化输出小数点后三位小数

print(**'{:.3f}'**.format(a))

2）b=’TOM’ 格式化输出按右对齐，保证占10位，不够用空格对齐print(**'{:>10}'**.format(b))

1. 写出a, b两个变量交换的语句

a,b = 1,2

a,b = b,a

1. 写出语句print(str(2.5)+str(2.5))的结果

2.52.5

1. 写出字符串s=”Introduction to Python”的下标范围

[0, 21]

二、读程序，写结果（5分/题\*6=30分）

j=1

for i in range(10):

j += j

1. j的值是多少：

1024

f = 0

g=1

for i in range(6):

print(f,”,”)

f = f + g

f = f - g

1. 写出输出结果：

0 ,

0 ,

0 ,

0 ,

0 ,

0 ,

1. 写出输出结果

for s in "PYTHON and C":

if s=="T":

continue

print(s, end="")

PYHON and C

4)写出程序输出

ls = []

def func(a, b=2):

ls = []

ls.append(b)

return a\*b

s = func("knock~")

print(s, ls)

knock~knock~ []

注意：，要空格

关于全局与局部变量：

1.函数内部的变量名如果第一次出现，且出现在=左边，即被视为定义一个局部变量，会将全局变量覆盖

2.变量被使用时(=右边/调用)，在全局域中有定义，而在局部没有定义，则会使用全局变量

3.如果使用的变量在全局域中有定义，在局部域中也有定义，则默认会使用局部变量

5）函数定义如下：

def change(i):

i=i\*i

请问以下程序while 循环多少次

i = 0

while i < 1000:

change(i)

i += 1

1000

数值类型（int和float）、字符串str、元组tuple都是不可变类型。而列表list、字典dict、集合set是可变类型

可变类型传入相当于引用传递，在函数中会改变值，不可变类型相当于值传递

6）写出a[]的结果

n=10

a=[0,1]

for i in range(2,8):

a += [a[i-1]+a[i-2]]

[0, 1, 1, 2, 3, 5, 8, 13]

编程题：（50分）

1. 编写一程序实现输入一数值n，实现输出2+4+6+...+2n的和（10分）

n = int(input())  
sum = 0  
for i in range(2,(2\*n)+1,2):  
 sum+=i  
print(sum)

2）甲、乙、丙、丁、戊、己、庚、辛、壬、癸称为十天干，分别对应阿拉伯数字1,2,3,4,5,6,7,8,9,10.

编写一个函数，实现阿拉伯数字（1-10）转换为十天干。（10分）

s = **"甲乙丙丁戊己庚辛壬癸"**n = int(input(**"请输入N:"**))  
print(s[n-1])

3）编写一程序，将100分制度，转换为四分制。（10分）

A [90,100] ，B [75,90) ，C [60,75)，F [0,60)

1. 编写一程序输入n值，构造一个n\*n的矩阵，方阵每个位置的值=行索引+列索引，然后以次对角线（右上左下）为对称轴交换数据生成一个新的矩阵，并输出这两个矩阵。（10分）

n = int(input(**"请输入N:"**))  
a = [[0 **for** i **in** range(n)] **for** j **in** range(n)]  
b = [[0 **for** i **in** range(n)] **for** j **in** range(n)]  
**for** i **in** range(n):  
 **for** j **in** range(n):  
 a[i][j] = i + j  
 print(a[i][j], end=**" "**)  
 print()  
  
**for** i **in** range(n):  
 **for** j **in** range(n):  
 b[i][j] = a[n-1-i][n-1-j]  
 print(b[i][j], end=**" "**)  
 print()

5)编写函数sigma(n)，用递归实现1+2+3+...+n的和（10分）

**def** sigma(n):  
 **if** n == 1:  
 **return** 1  
 **else**:  
 **return** n + sigma(n-1)  
  
n = int(input(**"输入n:"**))  
print(sigma(n))