

```
#!/usr/bin/env python
#coding: utf-8
```

```
'''
```

【程序 1】

题目：有 1、2、3、4 个数字，能组成多少个互不相同且无重复数字的三位数？都是多少？

1.程序分析：可填在百位、十位、个位的数字都是 1、2、3、4。组成所有的排列后再去掉不满足条件的排列。

2.程序源代码：

```
'''
```

```
for i in range(1,5):
    for j in range(1,5):
        for k in range(1,5):
            if( i != k ) and ( i != j ) and ( j != k ):
                print i,j,k
```

```
'''
```

【程序 2】

题目：企业发放的奖金根据利润提成。利润(I)低于或等于 10 万元时，奖金可提 10%；利润高于

10 万元，低于 20 万元时，低于 10 万元的部分按 10%提成，高于 10 万元的部分，可提

成 7.5%；20 万到 40 万之间时，高于 20 万元的部分，可提成 5%；40 万到 60 万之间时高于

40 万元的部分，可提成 3%；60 万到 100 万之间时，高于 60 万元的部分，可提成 1.5%，高于

100 万元时，超过 100 万元的部分按 1%提成，从键盘输入当月利润 I，求应发放奖金总数？

1.程序分析：请利用数轴来分界，定位。注意定义时需把奖金定义成长整型。

2.程序源代码：

```
'''
```

```
bonus1 = 100000 * 0.1
bonus2 = bonus1 + 100000 * 0.500075
bonus4 = bonus2 + 200000 * 0.5
bonus6 = bonus4 + 200000 * 0.3
bonus10 = bonus6 + 400000 * 0.15

i = int(raw_input('input gain:\n'))
if i <= 100000:
    bonus = i * 0.1
elif i <= 200000:
    bonus = bonus1 + (i - 100000) * 0.075
elif i <= 400000:
    bonus = bonus2 + (i - 200000) * 0.05
```

```

elif i <= 600000:
    bonus = bonus4 + (i - 400000) * 0.03
elif i <= 1000000:
    bonus = bonus6 + (i - 600000) * 0.015
else:
    bonus = bonus10 + (i - 1000000) * 0.01
print 'bonus = ',bonus
'''

```

【程序 3】

题目：一个整数，它加上 100 后是一个完全平方数，再加上 268 又是一个完全平方数，请问该数是多少？

1.程序分析：在 10 万以内判断，先将该数加上 100 后再开方，再将该数加上 268 后再开方，如果开方后

的结果满足如下条件，即是结果。请看具体分析：

2.程序源代码：

```

#include "math.h"
main()
{
long int i,x,y,z;
for (i=1;i<100000;i++)
    { x=sqrt(i+100);    /*x 为加上 100 后开方后的结果*/
      y=sqrt(i+268);    /*y 为再加上 268 后开方后的结果*/
      if(x*x==i+100&& y*y==i+268)/*如果一个数的平方根的平方等于该数，这说明此数是完
全平方数*/
          printf("\n%d\n",i);
    }
}
'''

import math
for i in range(10000):
    #转化为整型值
    x = int(math.sqrt(i + 100))
    y = int(math.sqrt(i + 268))
    if(x * x == i + 100) and (y * y == i + 268):
        print i
'''

```

【程序 4】

题目：输入某年某月某日，判断这一天是这一年的第几天？

1.程序分析：以 3 月 5 日为例，应该先把前两个月的加起来，然后再加上 5 天即本年的第几天，特殊

情况，闰年且输入月份大于 3 时需考虑多加一天。

2.程序源代码：

```
'''
```

```

year = int(raw_input('year:\n'))
month = int(raw_input('month:\n'))
day = int(raw_input('day:\n'))

months = (0,31,59,90,120,151,181,212,243,273,304,334)
if 0 <= month <= 12:
    sum = months[month - 1]
else:
    print 'data error'
sum += day
leap = 0
if (year % 400 == 0) or ((year % 4 == 0) and (year % 100 != 0)):
    leap = 1
if (leap == 1) and (month > 2):
    sum += 1
print 'it is the %dth day.' % sum

'''

```

【程序 5】

题目：输入三个整数 x,y,z ，请把这三个数由小到大输出。

1.程序分析：我们想办法把最小的数放到 x 上，先将 x 与 y 进行比较，如果 $x>y$ 则将 x 与 y 的值进行交换，

然后再用 x 与 z 进行比较，如果 $x>z$ 则将 x 与 z 的值进行交换，这样能使 x 最小。

2.程序源代码：

```

'''
l = []
for i in range(3):
    x = int(raw_input('integer:\n'))
    l.append(x)
l.sort()
print l
'''

```

【程序 6】

题目：用*号输出字母 C 的图案。

1.程序分析：可先用*号在纸上写出字母 C，再分行输出。

2.程序源代码：

```

'''
print 'Hello Python world!\n'
print '*' * 10
for i in range(5):
    print '*          *'
print '*' * 10
print '*\n' * 6
'''

```

'''

【程序 7】

题目：输出特殊图案，请在 c 环境中运行，看一看，Very Beautiful!

1.程序分析：字符共有 256 个。不同字符，图形不一样。

2.程序源代码：

'''

```
a = 176
b = 219
print chr(b),chr(a),chr(a),chr(a),chr(b)
print chr(a),chr(b),chr(a),chr(b),chr(a)
print chr(a),chr(a),chr(b),chr(a),chr(a)
print chr(a),chr(b),chr(a),chr(b),chr(a)
print chr(b),chr(a),chr(a),chr(a),chr(b)
```

'''

【程序 8】

题目：输出 9*9 口诀。

1.程序分析：分行与列考虑，共 9 行 9 列，i 控制行，j 控制列。

2.程序源代码：

```
#include "stdio.h"
main()
{
    int i,j,result;
    printf("\n");
    for (i=1;i<10;i++)
        { for(j=1;j<10;j++)
            {
                result=i*j;
                printf("%d*%d=%-3d",i,j,result);/*-3d 表示左对齐，占 3 位*/
            }
            printf("\n");/*每一行后换行*/
        }
}
```

'''

```
for i in range(1,10):
    for j in range(1,10):
        result = i * j
        print '%d * %d = %-3d' % (i,j,result)
    print "
```

'''

【程序 9】

题目：要求输出国际象棋棋盘。

1.程序分析：用 i 控制行，j 来控制列，根据 i+j 的和的变化来控制输出黑方格，还是白方格。

2.程序源代码:

```
#include "stdio.h"
main()
{
int i,j;
for(i=0;i<8;i++)
{
for(j=0;j<8;j++)
if((i+j)%2==0)
printf("%c%c",219,219);
else
printf(" ");
printf("\n");
}
}
'''
import sys
for i in range(8):
for j in range(8):
if(i + j) % 2 == 0:
sys.stdout.write(chr(219))
sys.stdout.write(chr(219))
else:
sys.stdout.write(' ')
print ""
'''
```

【程序 10】

题目：打印楼梯，同时在楼梯上方打印两个笑脸。

1.程序分析：用 i 控制行，j 来控制列，j 根据 i 的变化来控制输出黑方格的个数。

2.程序源代码:

```
'''
import sys
sys.stdout.write(chr(1))
sys.stdout.write(chr(1))
print ""

for i in range(1,11):
for j in range(1,i):
sys.stdout.write(chr(219))
sys.stdout.write(chr(219))
print ""
'''
```

【程序 11】

题目：古典问题：有一对兔子，从出生后第3个月起每个月都生一对兔子，小兔子长到第三个月

后每个月又生一对兔子，假如兔子都不死，问每个月的兔子总数为多少？

1.程序分析：兔子的规律为数列 1,1,2,3,5,8,13,21....

2.程序源代码：

```
main()
{
long f1,f2;
int i;
f1=f2=1;
for(i=1;i<=20;i++)
    { printf("%12ld %12ld",f1,f2);
      if(i%2==0) printf("\n");/*控制输出，每行四个*/
      f1=f1+f2; /*前两个月加起来赋值给第三个月*/
      f2=f1+f2; /*前两个月加起来赋值给第三个月*/
    }
}
'''
f1 = 1
f2 = 1
for i in range(1,21):
    print '%12d %12d' % (f1,f2)
    if (i % 2) == 0:
        print "
    f1 = f1 + f2
    f2 = f1 + f2
'''
```

【程序 12】

题目：判断 101-200 之间有多少个素数，并输出所有素数。

1.程序分析：判断素数的方法：用一个数分别去除 2 到 sqrt(这个数)，如果能被整除，则表明此数不是素数，反之是素数。

2.程序源代码：

```
'''
h = 0
leap = 1
from math import sqrt
from sys import stdout
for m in range(101,201):
    k = int(sqrt(m + 1))
    for i in range(2,k + 1):
        if m % i == 0:
            leap = 0
            break
```

```
    if leap == 1:
        print '%-4d' % m
        h += 1
        if h % 10 == 0:
            print "
    leap = 1
print 'The total is %d' % h
'''
```

【程序 13】

题目：打印出所有的“水仙花数”，所谓“水仙花数”是指一个三位数，其各位数字立方和等于该数

本身。例如：153 是一个“水仙花数”，因为 $153=1$ 的三次方+5 的三次方+3 的三次方。

1.程序分析：利用 for 循环控制 100-999 个数，每个数分解出个位，十位，百位。

2.程序源代码：

```
'''
for n in range(100,1001):
    i = n / 100
    j = n / 10 % 10
    k = n % 10
    if i * 100 + j * 10 + k == i + j ** 2 + k ** 3:
        print "%-5d" % n
'''
```

【程序 14】

题目：将一个正整数分解质因数。例如：输入 90,打印出 $90=2*3*3*5$ 。

程序分析：对 n 进行分解质因数，应先找到一个最小的质数 k，然后按下述步骤完成：

(1)如果这个质数恰等于 n，则说明分解质因数的过程已经结束，打印出即可。

(2)如果 $n < > k$ ，但 n 能被 k 整除，则应打印出 k 的值，并用 n 除以 k 的商,作为新的正整数你 n,

重复执行第一步。

(3)如果 n 不能被 k 整除，则用 k+1 作为 k 的值,重复执行第一步。

2.程序源代码：

```
'''
from sys import stdout
n = int(raw_input("input number:\n"))
print "n = %d" % n

for i in range(2,n + 1):
    while n != i:
        if n % i == 0:
            stdout.write(str(i))
            stdout.write("*")
```

```
        n = n / i
    else:
        break
print "%d" % n
'''
```

【程序 15】

题目：利用条件运算符的嵌套来完成此题：学习成绩 ≥ 90 分的同学用 A 表示，60-89 分之间的用 B 表示，

60 分以下的用 C 表示。

1.程序分析：(a>b)?a:b 这是条件运算符的基本例子。

2.程序源代码：

不支持这个运算符

```
'''
score = int(raw_input('input score:\n'))
if score >= 90:
    grade = 'A'
elif score >= 60:
    grade = 'B'
else:
    grade = 'C'

print '%d belongs to %s' % (score,grade)
'''
```

【程序 17】

题目：输入一行字符，分别统计出其中英文字母、空格、数字和其它字符的个数。

1.程序分析：利用 while 语句,条件为输入的字符不为'\n'.

2.程序源代码：

```
'''
import string
s = raw_input('input a string:\n')
letters = 0
space = 0
digit = 0
others = 0
for c in s:
    if c.isalpha():
        letters += 1
    elif c.isspace():
        space += 1
    elif c.isdigit():
        digit += 1
    else:
        others += 1
'''
```



```
print 'char = %d,space = %d,digit = %d,others = %d' % (letters,space,digit,others)
'''
```

题目：求 $s=a+aa+aaa+aaaa+aa\dots a$ 的值，其中 a 是一个数字。例如 $2+22+222+2222+22222$ (此时

共有 5 个数相加)，几个数相加有键盘控制。

1.程序分析：关键是计算出每一项的值。

2.程序源代码：

```
'''
Tn = 0
Sn = []
n = int(raw_input('n = :\n'))
a = int(raw_input('a = :\n'))
for count in range(n):
    Tn = Tn + a
    a = a * 10
    Sn.append(Tn)
    print Tn

Sn = reduce(lambda x,y : x + y,Sn)
print Sn
'''
```

【程序 19】

题目：一个数如果恰好等于它的因子之和，这个数就称为“完数”。例如 $6=1+2+3$ 。编程找出 1000 以内的所有完数。

1. 程序分析：请参照程序<--上页程序 14.

2.程序源代码：

```
'''
from sys import stdout
for j in range(2,1001):
    k = []
    n = -1
    s = j
    for i in range(1,j):
        if j % i == 0:
            n += 1
            s -= i
            k.append(i)

    if s == 0:
        print j
        for i in range(n):
            stdout.write(k[i])
            stdout.write(' ')
        print k[n]
'''
```

'''

【程序 20】

题目：一球从 100 米高度自由落下，每次落地后反跳回原高度的一半；再落下，求它在第 10 次落地时，共经过多少米？第 10 次反弹多高？

1.程序分析：见下面注释

2.程序源代码：

'''

```
Sn = 100.0
```

```
Hn = Sn / 2
```

```
for n in range(2,11):
```

```
    Sn += 2 * Hn
```

```
    Hn /= 2
```

```
print 'Total of road is %f' % Sn
```

```
print 'The tenth is %f meter' % Hn
```

'''

【程序 21】

题目：猴子吃桃问题：猴子第一天摘下若干个桃子，当即吃了一半，还不瘾，又多吃了一个第二天早上又将剩下的桃子吃掉一半，又多吃了一个。以后每天早上都吃了前一天剩下的一半零一个。到第 10 天早上想再吃时，见只剩下一个桃子了。求第一天共摘了多少。

1.程序分析：采取逆向思维的方法，从后往前推断。

2.程序源代码：

'''

```
x2 = 1
```

```
for day in range(9,0,-1):
```

```
    x1 = (x2 + 1) * 2
```

```
    x2 = x1
```

```
print x1
```

'''

【程序 22】

题目：两个乒乓球队进行比赛，各出三人。甲队为 a,b,c 三人，乙队为 x,y,z 三人。已抽签决定

比赛名单。有人向队员打听比赛的名单。a 说他不和 x 比，c 说他不和 x,z 比，请程序找出

三队赛手的名单。

1.程序分析：判断素数的方法：用一个数分别去除 2 到 sqrt(这个数)，如果能被整除，则表明此数不是素数，反之是素数。

2.程序源代码：

'''

```
for i in range(ord('x'),ord('z') + 1):
```

```
for j in range(ord('x'),ord('z') + 1):
    if i != j:
        for k in range(ord('x'),ord('z') + 1):
            if (i != k) and (j != k):
                if (i != ord('x')) and (k != ord('x')) and (k != ord('z')):
                    print 'order is a -- %s\t b -- %s\tc--%s' % (chr(i),chr(j),chr(k))
'''
```

【程序 23】

题目：打印出如下图案（菱形）

```
  *
 ***
*****
*****
 *****
  ***
   *
```

1.程序分析：先把图形分成两部分来看待，前四行一个规律，后三行一个规律，利用双重 for 循环，第一层控制行，第二层控制列。

2.程序源代码：

```
'''
from sys import stdout
for i in range(4):
    for j in range(2 - i + 1):
        stdout.write(' ')
    for k in range(2 * i + 1):
        stdout.write('*')
    print

for i in range(3):
    for j in range(i + 1):
        stdout.write(' ')
    for k in range(4 - 2 * i + 1):
        stdout.write('*')
    print
'''
```

【程序 24】

题目：有一分数序列：2/1，3/2，5/3，8/5，13/8，21/13...求出这个数列的前 20 项之和。

1.程序分析：请抓住分子与分母的变化规律。

2.程序源代码：

```
'''
#方法一
a = 2.0
b = 1.0
```

```
s = 0
for n in range(1,21):
    s += a / b
    t = a
    a = a + b
    b = t
print s
#方法二
s = 0.0
for n in range(1,21):
    s += a / b
    b,a = a , a + b
print s

s = 0.0
for n in range(1,21):
    s += a / b
    b,a = a , a + b
print s
#方法三
l = []
for n in range(1,21):
    b,a = a,a + b
    l.append(a / b)
print reduce(lambda x,y: x + y,l)
'''
```

【程序 25】

题目：求 $1+2!+3!+\dots+20!$ 的和

1.程序分析：此程序只是把累加变成了累乘。

2.程序源代码：

```
'''
#方法一
n = 0
s = 0
t = 1
for n in range(1,21):
    t *= n
    s += t
print '1! + 2! + 3! + ... + 20! = %d' % s

#方法二
s = 0
l = range(1,21)
def op(x):
```

```
    r = 1
    for i in range(1,x + 1):
        r *= i
    return r
s = sum(map(op,l))
print '1! + 2! + 3! + ... + 20! = %d' % s
'''
```

【程序 26】

题目：利用递归方法求 5!。

1.程序分析：递归公式：fn=fn_1*4!

2.程序源代码：

```
'''
def fact(j):
    sum = 0
    if j == 0:
        sum = 1
    else:
        sum = j * fact(j - 1)
    return sum

for i in range(5):
    print '%d! = %d' % (i,fact(i))
'''
```

【程序 27】

题目：利用递归函数调用方式，将所输入的 5 个字符，以相反顺序打印出来。

1.程序分析：

2.程序源代码：

```
'''
def palin(n):
    next = 0
    if n <= 1:
        next = input()
        print
        print next
    else:
        next = input()
        palin(n - 1)
        print next

i = 5
palin(i)
print
'''
```

【程序 28】

题目：有 5 个人坐在一起，问第五个人多少岁？他说比第 4 个人大 2 岁。问第 4 个人岁数，他说比第

3 个人大 2 岁。问第三个人，又说比第 2 人大两岁。问第 2 个人，说比第一个人大两岁。最后

问第一个人， he 说是 10 岁。请问第五个人多大？

1.程序分析：利用递归的方法，递归分为回推和递推两个阶段。要想知道第五个人岁数，需知道

第四人的岁数，依次类推，推到第一人（10 岁），再往回推。

'''

```
def age(n):
    if n == 1: c = 10
    else: c = age(n - 1) + 2
    return c
print age(5)
'''
```

【程序 29】

题目：给一个不多于 5 位的正整数，要求：一、求它是几位数，二、逆序打印出各位数字。

1. 程序分析：学会分解出每一位数，如下解释：(这里是一种简单的算法，师专数 002 班赵鑫提供)

2.程序源代码：

'''

```
x = int(raw_input("input a number:\n"))
a = x / 10000
b = x % 10000 / 1000
c = x % 1000 / 100
d = x % 100 / 10
e = x % 10
```

```
if a != 0:
    print "there are 5 ",e,d,c,b,a
elif b != 0:
    print "there are 4 ",d,c,b,a
elif c != 0:
    print "there are 3 ",e,d,c
elif d != 0:
    print "there are 2 ",e,d
else:
    print "there are 1",e
'''
```

题目：一个 5 位数，判断它是不是回文数。即 12321 是回文数，个位与万位相同，十位与千位相同。

1.程序分析：同 29 例

2.程序源代码：

'''

```
x = int(raw_input("input a number:\n"))
x = str(x)
for i in range(len(x)/2):
    if x[i] != x[-i - 1]:
        print 'this number is not a huiwen'
        break
print 'this number is a huiwen'
'''
```

程序 31】

题目：请输入星期一的第一个字母来判断一下是星期几，如果第一个字母一样，则继续判断第二个字母。

1.程序分析：用情况语句比较好，如果第一个字母一样，则判断用情况语句或 if 语句判断第二个字母。

2.程序源代码：

```
'''
from sys import stdin
letter = stdin.read(1)
stdin.flush()
while letter != 'Y':
    if letter == 'S':
        print 'please input second letter'
        letter = stdin.read(1)
        stdin.flush()
        if letter == 'a':
            print 'Saturday'
        elif letter == 'u':
            print 'Sunday'
        else:
            print 'data error'
            break
    elif letter == 'F':
        print 'Friday'
        break
    elif letter == 'M':
        print 'Monday'
        #break
    elif letter == 'T':
        print 'please input second letter'
        letter = stdin.read(1)
        stdin.flush()
        if letter == 'u':
            print 'Tuesday'
        elif letter == 'h':
            print 'Thursday'
```

```
    else:
        print 'data error'
        break
    elif letter == 'W':
        print 'Wednesday'
    else:
        print 'data error'
    letter = stdin.read(1)
    stdin.flush()

'''
```

【程序 32】

题目：Press any key to change color, do you want to try it. Please hurry up!

1.程序分析：

2.程序源代码：

不知道写呢 :(,先空着吧

```
'''
'''
```

【程序 33】

题目：学习 gotoxy()与 clrscr()函数

1.程序分析：

2.程序源代码：

不知道如何写 :(先空着吧

```
'''
'''
```

【程序 34】

题目：练习函数调用

1. 程序分析：

2.程序源代码：

```
'''
```

```
def hello_world():
    print 'hello world'
```

```
def three_hellos():
    for i in range(3):
        hello_world()
if __name__ == '__main__':
    three_hellos()
'''
```

【程序 35】

题目：文本颜色设置

1.程序分析：

2.程序源代码：

```
#include <conio.h>
```



```
void main(void)
{
int color;
for (color = 1; color < 16; color++)
    {
    textcolor(color);/*设置文本颜色*/
    cprintf("This is color %d\r\n", color);
    }
textcolor(128 + 15);
cprintf("This is blinking\r\n");
}
'''
'''
```

【程序 36】

题目：求 100 之内的素数

1.程序分析：

2.程序源代码：

```
'''
from math import sqrt
if __name__ == '__main__':
    N = 100
    a = range(0,N)
    for i in range(2,int(sqrt(N))):
        for j in range(i + 1,N):
            if (a[i] != 0) and (a[j] != 0):
                if a[j] % a[i] == 0:
                    a[j] = 0
    print
    for i in range(2,N):
        if a[i] != 0:
            print "%5d" % a[i]
            if (i - 2) % 10 == 0:
                print
'''
```

【程序 37】

题目：对 10 个数进行排序

1.程序分析：可以利用选择法，即从后 9 个比较过程中，选择一个最小的与第一个元素交换，下次类推，即用第二个元素与后 8 个进行比较，并进行交换。

2.程序源代码：

```
'''
if __name__ == "__main__":
    N = 10
    # input data
    print 'please input ten num:\n'
```

```
l = []
for i in range(N):
    l.append(int(raw_input('input a number:\n')))
print
for i in range(N):
    print l[i]
print

# sort ten num
for i in range(N - 1):
    min = i
    for j in range(i + 1, N):
        if l[min] > l[j]: min = j
    l[i], l[min] = l[min], l[i]
print 'after sorted'
for i in range(N):
    print l[i]

'''
```

【程序 38】

题目：求一个 3*3 矩阵对角线元素之和

1.程序分析：利用双重 for 循环控制输入二维数组，再将 a[i][i]累加后输出。

2.程序源代码：

```
'''
if __name__ == '__main__':
    a = []
    sum = 0.0
    for i in range(3):
        a.append([])
        for j in range(3):
            a[i].append(float(raw_input("input num:\n")))
    for i in range(3):
        sum += a[i][i]
    print sum

'''
```

【程序 39】

题目：有一个已经排好序的数组。现输入一个数，要求按原来的规律将它插入数组中。

1. 程序分析：首先判断此数是否大于最后一个数，然后再考虑插入中间的数的情况，插入后

此元素之后的数，依次后移一个位置。

2.程序源代码：

```
'''
if __name__ == '__main__':
```

```
# 方法一
a = [1,4,6,9,13,16,19,28,40,100,0]
print 'original list is:'
for i in range(len(a)):
    print a[i]
number = int(raw_input("insert a new number:\n"))
end = a[9]
if number > end:
    a[10] = number
else:
    for i in range(10):
        if a[i] > number:
            temp1 = a[i]
            a[i] = number
            for j in range(i + 1,11):
                temp2 = a[j]
                a[j] = temp1
                temp1 = temp2
            break
    for i in range(11):
        print a[i]
# 方法二
# insrt another number
number = int(raw_input('input a number:\n'))
if number > a[len(a) - 1]:
    a.append(number)
else:
    for i in range(len(a)):
        if a[i] > number:
            a.insert(i,number)
print a
```

'''

【程序 40】

题目：将一个数组逆序输出。

1.程序分析：用第一个与最后一个交换。

2.程序源代码：

'''

```
if __name__ == '__main__':
    a = [9,6,5,4,1]
    N = len(a)
    print a
    for i in range(len(a) / 2):
        a[i],a[N - i - 1] = a[N - i - 1],a[i]
```

```
    print a
'''
```

【程序 41】

题目：学习 `static` 定义静态变量的用法

1.程序分析：

2.程序源代码：

```
'''
```

python 没有这个功能了,只能这样了:)

```
def varfunc():
```

```
    var = 0
```

```
    print 'var = %d' % var
```

```
    var += 1
```

```
if __name__ == '__main__':
```

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

```
        varfunc()
```

```
# attribut of class
```

```
# 作为类的一个属性吧
```

```
class Static:
```

```
    StaticVar = 5
```

```
    def varfunc(self):
```

```
        self.StaticVar += 1
```

```
        print self.StaticVar
```

```
print Static.StaticVar
```

```
a = Static()
```

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

```
    a.varfunc()
```

```
'''
```

题目：学习使用 `auto` 定义变量的用法

1.程序分析：

2.程序源代码：

没有 `auto` 关键字，使用变量作用域来举例吧

```
'''
```

```
num = 2
```

```
def autofunc():
```

```
    num = 1
```

```
    print 'internal block num = %d' % num
```

```
    num += 1
```

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

```
    print 'The num = %d' % num
```

```
    num += 1
```

```
    autofunc()
```

```
'''
```

【程序 43】

题目：学习使用 `static` 的另一用法。

1.程序分析：

2.程序源代码：

有一个 `static` 变量的用法，python 是没有，演示一个 python 作用域使用方法

```
'''
```

```
class Num:
    nNum = 1
    def inc(self):
        self.nNum += 1
        print 'nNum = %d' % self.nNum

if __name__ == '__main__':
    nNum = 2
    inst = Num()
    for i in range(3):
        nNum += 1
        print 'The num = %d' % nNum
        inst.inc()
'''
```

【程序 44】

题目：学习使用 `external` 的用法。

1.程序分析：

2.程序源代码：

`external.py` 代码：

```
'''
```

```
import external
if __name__ == '__main__':
    print external.add(10,20)
'''
```

【程序 45】

题目：学习使用 `register` 定义变量的方法。

1.程序分析：

2.程序源代码：

没有 `register` 关键字，用整型变量代替

```
'''
```

```
tmp = 0
for i in range(1,101):
    tmp += i
print 'The sum is %d' % tmp
'''
```

【程序 46】

题目：宏#define 命令练习(1)

1.程序分析：

2.程序源代码：

没有 C 语言的宏，就这么写了

```
"""
TRUE = 1
FALSE = 0
def SQ(x):
    return x * x
print 'Program will stop if input value less than 50.'
again = 1
while again:
    num = int(raw_input('Please input number'))
    print 'The square for this number is %d' % (SQ(num))
    if num >= 50:
        again = TRUE
    else:
        again = FALSE
"""
```

题目：宏#define 命令练习(2)

1.程序分析：

2.程序源代码：

```
#include "stdio.h"
```

```
#define exchange(a,b) { \ /*宏定义中允许包含两道衣裳命令的情形，此时必须在最右边加上  
"\*/
```

```
int t;\
t=a;\
a=b;\
b=t;\
}
```

这个宏定义 python 不支持

```
"""
def exchange(a,b):
    a,b = b,a
    return (a,b)

if __name__ == '__main__':
    x = 10
    y = 20
    print 'x = %d,y = %d' % (x,y)
    x,y = exchange(x,y)
    print 'x = %d,y = %d' % (x,y)
"""
```

【程序 48】

题目：宏#define 命令练习(3)

1.程序分析:

2.程序源代码:

```
#define LAG >
#define SMA <
#define EQ ==
#include "stdio.h"
void main()
{
    int i=10;
    int j=20;
    if(i LAG j)
        printf("\40: %d larger than %d \n",i,j);
    else if(i EQ j)
        printf("\40: %d equal to %d \n",i,j);
    else if(i SMA j)
        printf("\40:%d smaller than %d \n",i,j);
    else
        printf("\40: No such value.\n");
}
```

不知道如何用 python 实现类似的功能

'''

```
if __name__ == '__main__':
    i = 10
    j = 20
    if i > j:
        print '%d larger than %d' % (i,j)
    elif i == j:
        print '%d equal to %d' % (i,j)
    elif i < j:
        print '%d smaller than %d' % (i,j)
    else:
        print 'No such value'
```

'''

【程序 49】

题目：#if #ifdef 和 #ifndef 的综合应用。

1. 程序分析:

2.程序源代码:

```
#include "stdio.h"
#define MAX
#define MAXIMUM(x,y) (x>y)?x:y
#define MINIMUM(x,y) (x>y)?y:x
void main()
```

```
{
    int a=10,b=20;
#ifdef MAX
    printf("\40: The largar one is %d\n",MAXIMUM(a,b));
#else
    printf("\40: The lower one is %d\n",MINIMUM(a,b));
#endif
#endif
#endif MIN
    printf("\40: The lower one is %d\n",MINIMUM(a,b));
#else
    printf("\40: The largar one is %d\n",MAXIMUM(a,b));
#endif
#undef MAX
#ifdef MAX
    printf("\40: The largar one is %d\n",MAXIMUM(a,b));
#else
    printf("\40: The lower one is %d\n",MINIMUM(a,b));
#endif
#define MIN
#endif MIN
    printf("\40: The lower one is %d\n",MINIMUM(a,b));
#else
    printf("\40: The largar one is %d\n",MAXIMUM(a,b));
#endif
}
```

这个还是预处理的用法，python 不支持这样的机制，演示 lambda 的使用。

```
"""
MAXIMUM = lambda x,y : (x > y) * x + (x < y) * y
MINIMUM = lambda x,y : (x > y) * y + (x < y) * x

if __name__ == '__main__':
    a = 10
    b = 20
    print 'The largar one is %d' % MAXIMUM(a,b)
    print 'The lower one is %d' % MINIMUM(a,b)
"""
```

【程序 51】

题目：学习使用按位与&。

1.程序分析：0&0=0; 0&1=0; 1&0=0; 1&1=1

2.程序源代码：

```
"""
if __name__ == '__main__':
    a = 077
    b = a & 3
```



```

    print 'a & b = %d' % b
    b &= 7
    print 'a & b = %d' % b
'''

```

题目：学习使用按位或 | 。

1.程序分析：0|0=0; 0|1=1; 1|0=1; 1|1=1

2.程序源代码：

'''

```

if __name__ == '__main__':
    a = 077
    b = a | 3
    print 'a | b is %d' % b
    b |= 7
    print 'a | b is %d' % b
'''

```

【程序 53】

题目：学习使用按位异或 ^ 。

1.程序分析：0^0=0; 0^1=1; 1^0=1; 1^1=0

2.程序源代码：

'''

```

if __name__ == '__main__':
    a = 077
    b = a ^ 3
    print 'The a ^ 3 = %d' % b
    b ^= 7
    print 'The a ^ b = %d' % b
'''

```

【程序 54】

题目：取一个整数 a 从右端开始的 4~7 位。

程序分析：可以这样考虑：

(1)先使 a 右移 4 位。

(2)设置一个低 4 位全为 1,其余全为 0 的数。可用~(~0<<4)

(3)将上面二者进行&运算。

'''

```

if __name__ == '__main__':
    a = int(raw_input('input a number:\n'))
    b = a >> 4
    c = ~(~0 << 4)
    d = b & c
    print '%o\t%o' % (a,d)
'''

```

【程序 55】

题目：学习使用按位取反~。

- 1.程序分析: $\sim 0=1; \sim 1=0;$
- 2.程序源代码:
如何查看复数的 16 进制数

```
'''  
if __name__ == '__main__':  
    a = 234  
    b = ~a  
    print 'The a\'s 1 complement is %d' % b  
    a = ~a  
    print 'The a\'s 2 complement is %d' % a  
'''
```

【程序 56】

题目: 画图, 学用 circle 画圆形。

- 1.程序分析:
- 2.程序源代码:

```
#include "graphics.h"  
main()  
{  
    int driver,mode,i;  
    float j=1,k=1;  
    driver=VGA;mode=VGAHI;  
    initgraph(&driver,&mode,"");  
    setbkcolor(YELLOW);  
    for(i=0;i<=25;i++)  
    {  
        setcolor(8);  
        circle(310,250,k);  
        k=k+j;  
        j=j+0.3;  
    }  
}  
'''  
if __name__ == '__main__':  
    from Tkinter import *  
  
    canvas = Canvas(width=800, height=600, bg='yellow')  
    canvas.pack(expand=YES, fill=BOTH)  
    k = 1  
    j = 1  
    for i in range(0,26):  
        canvas.create_oval(310 - k,250 - k,310 + k,250 + k, width=1)  
        k += j  
        j += 0.3
```

```
    mainloop()
'''
```

【程序 57】

题目：画图，学用 line 画直线。

1.程序分析：

2.程序源代码：

```
'''
```

```
if __name__ == '__main__':
    from Tkinter import *

    canvas = Canvas(width=300, height=300, bg='green')
    canvas.pack(expand=YES, fill=BOTH)
    x0 = 263
    y0 = 263
    y1 = 275
    x1 = 275
    for i in range(19):
        canvas.create_line(x0,y0,x0,y1, width=1, fill='red')
        x0 = x0 - 5
        y0 = y0 - 5
        x1 = x1 + 5
        y1 = y1 + 5

    x0 = 263
    y1 = 275
    y0 = 263
    for i in range(21):
        canvas.create_line(x0,y0,x0,y1,fill = 'red')
        x0 += 5
        y0 += 5
        y1 += 5
```

```
    mainloop()
'''
```

【程序 58】

题目：画图，学用 rectangle 画方形。

1.程序分析：利用 for 循环控制 100-999 个数，每个数分解出个位，十位，百位。

2.程序源代码：

```
'''
```

```
if __name__ == '__main__':
    from Tkinter import *
    root = Tk()
    root.title('Canvas')
```

```

canvas = Canvas(root,width = 400,height = 400,bg = 'yellow')
x0 = 263
y0 = 263
y1 = 275
x1 = 275
for i in range(19):
    canvas.create_rectangle(x0,y0,x1,y1)
    x0 -= 5
    y0 -= 5
    x1 += 5
    y1 += 5

canvas.pack()
root.mainloop()
'''

```

题目：画图，综合例子。

1.程序分析：

2.程序源代码：

```

'''
if __name__ == '__main__':
    from Tkinter import *
    canvas = Canvas(width = 300,height = 300,bg = 'green')
    canvas.pack(expand = YES,fill = BOTH)
    x0 = 150
    y0 = 100
    canvas.create_oval(x0 - 10,y0 - 10,x0 + 10,y0 + 10)
    canvas.create_oval(x0 - 20,y0 - 20,x0 + 20,y0 + 20)
    canvas.create_oval(x0 - 50,y0 - 50,x0 + 50,y0 + 50)
    import math
    B = 0.809
    for i in range(16):
        a = 2 * math.pi / 16 * i
        x = math.ceil(x0 + 48 * math.cos(a))
        y = math.ceil(y0 + 48 * math.sin(a) * B)
        canvas.create_line(x0,y0,x,y,fill = 'red')
    canvas.create_oval(x0 - 60,y0 - 60,x0 + 60,y0 + 60)

    for k in range(501):
        for i in range(17):
            a = (2 * math.pi / 16) * i + (2 * math.pi / 180) * k
            x = math.ceil(x0 + 48 * math.cos(a))
            y = math.ceil(y0 + 48 * math.sin(a) * B)
            canvas.create_line(x0,y0,x,y,fill = 'red')
'''

```

```

        for j in range(51):
            a = (2 * math.pi / 16) * i + (2 * math.pi / 180) * k - 1
            x = math.ceil(x0 + 48 * math.cos(a))
            y = math.ceil(y0 + 48 * math.sin(a) * B)
            canvas.create_line(x0,y0,x,y,fill = 'red')
    mainloop()
'''

```

【程序 60】

题目：画图，综合例子。

1.程序分析：

2.程序源代码：

键盘不知道如何响应，先不写这个

```

#include "graphics.h"
#define LEFT 0
#define TOP 0
#define RIGHT 639
#define BOTTOM 479
#define LINES 400
#define MAXCOLOR 15
main()
{
    int driver,mode,error;
    int x1,y1;
    int x2,y2;
    int dx1,dy1,dx2,dy2,i=1;
    int count=0;
    int color=0;
    driver=VGA;
    mode=VGAHI;
    initgraph(&driver,&mode,"");
    x1=x2=y1=y2=10;
    dx1=dy1=2;
    dx2=dy2=3;
    while(!kbhit())
    {
        line(x1,y1,x2,y2);
        x1+=dx1;y1+=dy1;
        x2+=dx2;y2+=dy2;
        if(x1<=LEFT | x1>=RIGHT)
            dx1=-dx1;
        if(y1<=TOP | y1>=BOTTOM)
            dy1=-dy1;
        if(x2<=LEFT | x2>=RIGHT)
            dx2=-dx2;
    }
}

```

```

        if(y2<=TOP| |y2>=BOTTOM)
            dy2=-dy2;
        if(++count>LINES)
        {
            setcolor(color);
            color=(color>=MAXCOLOR)?0:++color;
        }
    }
    closegraph();
}
"
'''

```

【程序 61】

题目：打印出杨辉三角形（要求打印出 10 行如下图）

1.程序分析：

'''

```

if __name__ == '__main__':
    a = []
    for i in range(10):
        a.append([])
        for j in range(10):
            a[i].append(0)
    for i in range(10):
        a[i][0] = 1
        a[i][i] = 1
    for i in range(2,10):
        for j in range(1,i):
            a[i][j] = a[i - 1][j-1] + a[i - 1][j]
    from sys import stdout
    for i in range(10):
        for j in range(i + 1):
            stdout.write(a[i][j])
            stdout.write(' ')
        print
'''

```

【程序 62】

题目：学习 putpixel 画点。

1.程序分析：

2.程序源代码：

```

#include "stdio.h"
#include "graphics.h"
main()

```

```

{
    int i,j,driver=VGA,mode=VGAHI;
    initgraph(&driver,&mode,"");
    setbkcolor(YELLOW);
    for(i=50;i<=230;i+=20)
        for(j=50;j<=230;j++)
            putpixel(i,j,1);
    for(j=50;j<=230;j+=20)
        for(i=50;i<=230;i++)
            putpixel(i,j,1);
}
'''
'''

```

题目：画椭圆 ellipse

1.程序分析：

2.程序源代码：

```

'''
if __name__ == '__main__':
    from Tkinter import *
    x = 360
    y = 160
    top = y - 30
    bottom = y - 30

    canvas = Canvas(width = 400,height = 600,bg = 'white')
    for i in range(20):
        canvas.create_oval(250 - top,250 - bottom,250 + top,250 + bottom)
        top -= 5
        bottom += 5
    canvas.pack()
    mainloop()
'''

```

题目：利用 ellipse and rectangle 画图。

1.程序分析：

2.程序源代码：

```

'''
if __name__ == '__main__':
    from Tkinter import *
    canvas = Canvas(width = 400,height = 600,bg = 'white')
    left = 20
    right = 50
    top = 50
    num = 15
    for i in range(num):

```

```

        canvas.create_oval(250 - right,250 - left,250 + right,250 + left)
        canvas.create_oval(250 - 20,250 - top,250 + 20,250 + top)
        canvas.create_rectangle(20 - 2 * i,20 - 2 * i,10 * (i + 2),10 * (i + 2))
        right += 5
        left += 5
        top += 10

    canvas.pack()
    mainloop()

'''

```

【程序 65】

题目：一个最优美的图案。

1.程序分析：

2.程序源代码：

```

'''
import math
class PTS:
    def __init__(self):
        self.x = 0
        self.y = 0
points = []

def LineToDemo():
    from Tkinter import *
    screenx = 400
    screeny = 400
    canvas = Canvas(width = screenx,height = screeny,bg = 'white')

    AspectRatio = 0.85
    MAXPTS = 15
    h = screeny
    w = screenx
    xcenter = w / 2
    ycenter = h / 2
    radius = (h - 30) / (AspectRatio * 2) - 20
    step = 360 / MAXPTS
    angle = 0.0
    for i in range(MAXPTS):
        rads = angle * math.pi / 180.0
        p = PTS()
        p.x = xcenter + int(math.cos(rads) * radius)
        p.y = ycenter - int(math.sin(rads) * radius * AspectRatio)
        angle += step

```



```

        points.append(p)
    canvas.create_oval(xcenter - radius,ycenter - radius,
                      xcenter + radius,ycenter + radius)
    for i in range(MAXPTS):
        for j in range(i,MAXPTS):
            canvas.create_line(points[i].x,points[i].y,points[j].x,points[j].y)

    canvas.pack()
    mainloop()
if __name__ == '__main__':
    LineToDemo()
'''

```

【程序 66】

题目：输入 3 个数 a,b,c，按大小顺序输出。

1.程序分析：利用指针方法。

2.程序源代码：

```

'''
if __name__ == '__main__':
    n1 = int(raw_input('n1 = :\n'))
    n2 = int(raw_input('n2 = :\n'))
    n3 = int(raw_input('n3 = :\n'))

    def swap(p1,p2):
        return p2,p1

    if n1 > n2 : n1,n2 = swap(n1,n2)
    if n1 > n3 : n1,n3 = swap(n1,n3)
    if n2 > n3 : n2,n3 = swap(n2,n3)

    print n1,n2,n3
'''

```

题目：输入数组，最大的与第一个元素交换，最小的与最后一个元素交换，输出数组。

1.程序分析：谭浩强的书中答案有问题。

2.程序源代码：

```

'''
def inp(numbers):
    for i in range(9):
        numbers.append(int(raw_input('input a number:\n')))
    numbers.append(int(raw_input('input a number:\n')))
p = 0
def max_min(array):
    max = min = 0
    for i in range(1,len(array) - 1):
        p = i

```

```

        if array[p] > array[max] : max = p
        elif array[p] < array[min] : min = p
    k = max
    l = min
    array[0],array[l] = array[l],array[0]
    array[9],array[k] = array[k],array[9]

def outp(numbers):
    for i in range(len(numbers)):
        print numbers[i]

if __name__ == '__main__':
    array = []
    inp(array)
    max_min(array)
    outp(array)

```

'''

【程序 68】

题目：有 n 个整数，使其前面各数顺序向后移 m 个位置，最后 m 个数变成最前面的 m 个数

1.程序分析：

2.程序源代码：

'''

```

if __name__ == '__main__':
    n = int(raw_input('the total number is:\n'))
    m = int(raw_input('back m:\n'))

    def move(array,n,m):
        array_end = array[n - 1]
        for i in range(n - 1,-1,- 1):
            array[i] = array[i - 1]
        array[0] = array_end
        m -= 1
        if m > 0:move(array,n,m)

    number = []
    for i in range(n):
        number.append(int(raw_input('input a number:\n')))
    print 'original number:',number

    move(number,n,m)

    print 'after moved:',number

```

'''

【程序 69】

题目：有 n 个人围成一圈，顺序排号。从第一个人开始报数（从 1 到 3 报数），凡报到 3 的人退出

圈子，问最后留下的是原来第几号的那位。

1. 程序分析：

2. 程序源代码：

'''

```
if __name__ == '__main__':
    nmax = 50
    n = int(raw_input('please input the total of numbers:'))
    num = []
    for i in range(n):
        num.append(i + 1)

    i = 0
    k = 0
    m = 0

    while m < n - 1:
        if num[i] != 0 : k += 1
        if k == 3:
            num[i] = 0
            k = 0
            m += 0
        i += 1
        if i == n : i = 0

    i = 0
    while num[i] == 0: i += 1
    print num[i]
```

'''

题目：写一个函数，求一个字符串的长度，在 main 函数中输入字符串，并输出其长度。

1. 程序分析：

2. 程序源代码

就这样吧

'''

```
if __name__ == '__main__':
    s = raw_input('please input a string:\n')
    print 'the string has %d characters.' % len(s)
```

'''

【程序 71】

题目：编写 input() 和 output() 函数输入，输出 5 个学生的数据记录。

1. 程序分析：

2.程序源代码:

使用 list 来模拟结构 (不使用 class)

```
stu = [string,string,list]
'''
N = 3
#stu
    # num : string
    # name : string
    # score[4]: list
student = []
for i in range(5):
    student.append(["",[],[]])

def input_stu(stu):
    for i in range(N):
        stu[i][0] = raw_input('input student num:\n')
        stu[i][1] = raw_input('input student name:\n')
        for j in range(3):
            stu[i][2].append(int(raw_input('score:\n')))

def output_stu(stu):
    for i in range(N):
        print '%-6s%-10s' % ( stu[i][0],stu[i][1] )
        for j in range(3):
            print '%-8d' % stu[i][2][j]

if __name__ == '__main__':
    input_stu(student)
    print student
    output_stu(student)
'''
```

【程序 72】

题目: 创建一个链表。

1.程序分析:

2.程序源代码:

```
'''
if __name__ == '__main__':
    ptr = []
    for i in range(5):
        num = int(raw_input('please input a number:\n'))
        ptr.append(num)
    print ptr
'''
```

题目: 反向输出一个链表。

- 1.程序分析:
- 2.程序源代码:

```
"""
if __name__ == '__main__':
    ptr = []
    for i in range(5):
        num = int(raw_input('please input a number:\n'))
        ptr.append(num)
    print ptr
    ptr.reverse()
    print ptr
"""
```

【程序 74】

题目：连接两个链表。

- 1.程序分析:
- 2.程序源代码:

代码上好像只有，列表排序

```
"""
if __name__ == '__main__':
    arr1 = (3,12,8,9,11)
    ptr = list(arr1)
    print ptr
    ptr.sort()
    print ptr
"""
```

【程序 75】

题目：放松一下，算一道简单的题目。

- 1.程序分析:
- 2.程序源代码:

```
"""
if __name__ == '__main__':
    for i in range(5):
        n = 0
        if i != 1: n += 1
        if i == 3: n += 1
        if i == 4: n += 1
        if i != 4: n += 1
        if n == 3: print 64 + i
"""
```

【程序 76】

题目：编写一个函数，输入 n 为偶数时，调用函数求 $1/2+1/4+\dots+1/n$ ，当输入 n 为奇数时，调用函数

$1/1+1/3+\dots+1/n$ (利用指针函数)

- 1.程序分析:

2.程序源代码:

```
'''
def peven(n):
    i = 0
    s = 0.0
    for i in range(2,n + 1,2):
        s += 1.0 / i
    return s

def podd(n):
    s = 0.0
    for i in range(1, n + 1,2):
        s += 1 / i
    return s

def dcall(fp,n):
    s = fp(n)
    return s

if __name__ == '__main__':
    n = int(raw_input('input a number:\n'))
    if n % 2 == 0:
        sum = dcall(peven,n)
    else:
        sum = dcall(podd,n)
    print sum
'''
```

【程序 89】

题目: 某个公司采用公用电话传递数据, 数据是四位的整数, 在传递过程中是加密的, 加密规则如下:

每位数字都加上 5, 然后用和除以 10 的余数代替该数字, 再将第一位和第四位交换, 第二位和第三位交换。

1.程序分析:

2.程序源代码:

```
'''
from sys import stdout
if __name__ == '__main__':
    a = int(raw_input('input a number:\n'))
    aa = []
    aa.append(a % 10)
    aa.append(a % 100 / 10)
    aa.append(a % 1000 / 100)
    aa.append(a / 1000)
```

```

for i in range(4):
    aa[i] += 5
    aa[i] %= 10
for i in range(2):
    aa[i],aa[3 - i] = aa[3 - i],aa[i]
for i in range(3,-1,-1):
    stdout.write(aa[i])
'''

```

【程序 90】

题目：专升本一题，读结果。

1.程序分析：

2.程序源代码：

```

'''
if __name__ == '__main__':
    M = 5
    a = [1,2,3,4,5]
    i = 0
    j = M - 1
    while i < M:
        a[i],a[j] = a[j],a[i]
        print a
        i += 1
        j -= 1
    for i in range(5):
        print a[i]
'''

```

【程序 91】

题目：时间函数举例 1

1.程序分析：

2.程序源代码：

```

'''
if __name__ == '__main__':
    import time
    print time.ctime(time.time())
    print time.asctime(time.localtime(time.time()))
    print time.asctime(time.gmtime(time.time()))
'''

```

【程序 92】

题目：时间函数举例 2

1.程序分析：

2.程序源代码：

```

'''
if __name__ == '__main__':
    import time

```

```

start = time.time()
for i in range(3000):
    print i
end = time.time()

print end - start
'''

```

【程序 93】

题目：时间函数举例 3

1.程序分析：

2.程序源代码：

```

'''
if __name__ == '__main__':
    import time
    start = time.clock()
    for i in range(10000):
        print i
    end = time.clock()
    print 'different is %6.3f' % (end - start)

'''

```

【程序 94】

题目：时间函数举例 4,一个猜数游戏，判断一个人反应快慢。（版主初学时编的）

1.程序分析：

2.程序源代码：

```

'''
if __name__ == '__main__':
    import time
    import random

    play_it = raw_input('do you want to play it.(\'y\' or \'n\')')
    while play_it == 'y':
        c = raw_input('input a character:\n')
        i = random.randint(0,2**32) % 100
        print 'please input number you guess:\n'
        start = time.clock()
        a = time.time()
        guess = int(raw_input('input your guess:\n'))
        while guess != i:
            if guess > i:
                print 'please input a little smaller'
                guess = int(raw_input('input your guess:\n'))
            else:
                print 'please input a little bigger'

```



```

        guess = int(raw_input('input your guess:\n'))
    end = time.clock()
    b = time.time()
    var = (end - start) / 18.2
    print var
    # print 'It took you %6.3 seconds' % time.difftime(b,a)
    if var < 15:
        print 'you are very clever!'
    elif var < 25:
        print 'you are normal!'
    else:
        print 'you are stupid!'
    print 'Congradulations'
    print 'The number you guess is %d' % i
    play_it = raw_input('do you want to play it.')

```

'''

【程序 96】

题目：计算字符串中子串出现的次数

1.程序分析：

2.程序源代码：

'''

```

if __name__ == '__main__':
    str1 = raw_input('input a string:\n')
    str2 = raw_input('input a sub string:\n')
    ncount = str1.count(str2)
    print ncount

```

'''

【程序 97】

题目：从键盘输入一些字符，逐个把它们送到磁盘上去，直到输入一个#为止。

1.程序分析：

2.程序源代码：

'''

```

if __name__ == '__main__':
    from sys import stdout
    filename = raw_input('input a file name:\n')
    fp = open(filename,"w")
    ch = raw_input('input string:\n')
    while ch != '#':
        fp.write(ch)
        stdout.write(ch)
        ch = raw_input("")
    fp.close()

```

'''

【程序 98】

题目：从键盘输入一个字符串，将小写字母全部转换成大写字母，然后输出到一个磁盘文件“test”中保存。

输入的字符串以！结束。

1.程序分析：

2.程序源代码：

'''

```
if __name__ == '__main__':
    fp = open('test.txt','w')
    string = raw_input('please input a string:\n')
    string = string.upper()
    fp.write(string)
    fp = open('test.txt','r')
    print fp.read()
    fp.close()
```

'''

程序 99】

题目：有两个磁盘文件 A 和 B,各存放一行字母,要求把这两个文件中的信息合并(按字母顺序排列),

输出到一个新文件 C 中.

1.程序分析：

2.程序源代码：

'''

```
if __name__ == '__main__':
    import string
    fp = open('JCP099.py')
    a = fp.read()
    fp.close()

    fp = open('JCP098.py')
    b = fp.read()
    fp.close()

    fp = open('C.txt','w')
    l = list(a + b)
    l.sort()
    s = ""
    s = s.join(l)
    fp.write(s)
    fp.close()
```