

第三部分

Python 编程练习实例

实例 1 绘制正弦曲线

1. 编程思路

利用屏幕的像素点坐标,通过 $y = \sin(x)$ 计算出点的坐标,使画笔移动到该位置即可。

2. 程序代码

```
from turtle import *
from math import *
setup(800, 600, 300, 200)
pensize(10)
hideturtle()
speed(10)
color('red')
up()
goto(-300, 0)
down()
for i in range(618):
    x=i
    y=sin(i/100) * 200
    goto(x-300, y)
```



实例 2 模拟评分

1. 编程思路

在比赛中,有 10 名评委为参赛的选手打分,分数为 1~100 分。选手最后得分为去掉一个最高分和一个最低分,其余 8 个分数的平均值。

2. 程序代码

```
from random import *
Max=0
Min=100
S=0
for i in range(10):
    f=randint(0,100)
    print(i+1,f)
    S=S+f
    if Max<f:
        Max=f
    if Min>f:
        Min=f
print('Max=',Max,' Min=',Min)
print('平均分:',(S-Max-Min)/8)
```

实例3 求 $S=A+AA+AAA+\cdots+AA\cdots A$ 的值

1. 编程思路

a 是一个 1~9 范围内的数字。例如 $2+22+222+2222+22222$ (此时共有 5 个数相加), 几个数相加由键盘控制, 关键是计算出每一项的值。

2. 程序代码



```
Tn=0
Sn=[]
n=eval(input('n='))
a=eval(input('a='))
for i in range(n):
    Tn=Tn+a
    a=a*10
    Sn.append(Tn)
    print(Tn,end='')
    if (i<n-1):
        print("+",end='')
    else:
        print("=",end='')
print(sum(Sn)) # sum 是列表求和函数
```

实例 4 球的反弹距离和高度计算

1. 编程思路



用户输入球的初始高度以及允许球持续弹跳的次数,每次落地后反弹的高度是原高度的一半,输出是球每次落地的距离和反弹的高度。需要注意,每次的距离和高度不是同一个值。

2. 程序代码

```
n=eval(input("请输入弹跳次数: "))
h=eval(input("请输入初始高度: "))
s=h
print('n   s   h')
for i in range(n):
    print(i+1,s,h/2)
    s=s+h
    h=h/2
```

实例5 鸡兔同笼问题

1. 编程思路

一个笼子里面关了鸡和兔子(鸡有 2 只脚,兔子有 4 只脚)。已知笼子里面脚的总数是 100,可利用穷举法列出笼子里面有多少只鸡和多少只兔子。



2. 程序代码

```
for ji in range(51):                # 最多 50 只鸡
    for tu in range(26):            # 最多 25 只兔子
        if ji * 2 + tu * 4 == 100:
            print("鸡有%d只,兔子有%d只"%(ji, tu))
```

实例 6 在屏幕上显示杨辉三角形

1. 编程思路

杨辉三角形的形式如下：



```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1
```

将杨辉三角形左对齐,可以看出杨辉三角形的规律。

- (1) 第 i 行有 i 个数。
 - (2) 第 1 列和主对角线上的数字都是 1。
 - (3) 其他位置的数字为上一行前一列和上一行同一列两个位置的数字相加。
- 利用循环计算出数据并以列表的形式存储,最后打印输出。

2. 程序代码

```
N=eval(input('N='))
y=[[1]] # 初始列表
for i in range(1,N):
    s=[1]
    for j in range(1,i):
        x=y[i-1][j-1]+y[i-1][j]
        s.append(x)
    s.append(1)
    y.append(s)
for i in range(N):
    print(' '* (10+N-i) * 3,end=' ') # 定位输出
    for j in range(i+1):
        print('{0: '^6}'.format(y[i][j]),end=' ')
    print()
```

实例7 统计选票

1. 编程思路

某次选举活动中有 6 个候选人,其代号分别用 1~6 表示。

假设有若干选民,每个选民只能选一个候选人,即每张选票上出现的数字只能是 1~6 范围内的某一个数字,每张选票上所投候选人的代号由键盘输入,当输入完所有选票后用数字 0 作为终止数据输入的标志。

要求统计输出每个候选人的得票数。

2. 程序代码

```
dicall={1:'张三红',2:'李明杨',3:'汪乐',4:'赵宏',5:'谭晓方',\
        6:'王静'}
lst=dicall.keys()
dic={}
for n in lst:
    dic[n]=0
while True:
    n=int(input('请输入你的选票 1~6,输入 0,则退出: \n'))
    if n==0:
        break
    if 1<=n<=6:
        dic[n]=dic[n]+1
print()
for n in lst:
    print('%s \t %d 票'%(dicall[n],dic[n]))
print()
```



实例 8 验证四方定理

1. 编程思路



所有自然数至多只要用 4 个整数的平方和就可以表示,这就是数论中著名的四方定理。用 4 个变量采用试探的方法进行计算,满足要求时输出计算结果。

例如, $N=5$ 时,输出下列结果:

```
5=1*1 + 2*2 + 0*0 + 0*0
5=2*2 + 0*0 + 0*0 + 1*1
5=2*2 + 0*0 + 1*1 + 0*0
5=2*2 + 1*1 + 0*0 + 0*0
```

2. 程序代码

```
N=eval(input('N='))
for i in range(0,N//2+1):
    for j in range(i+1):
        for k in range(i+1):
            for l in range(i+1):
                if N==i*i+j*j+k*k+l*l:
                    print(N,'=',i,'*',i,'+',j,'*',j,'+',k,'*',k,'+',l,'*',l)
```