

例 4.1

```
temperature=float(input("请输入摄氏温度的数值: "))
if temperature>=35:
    print(f"{temperature}℃，高温天气预警！")
```

例 4.2 输入一个单词，如果在给定的字符串中，

#输出该单词在字符串中的位置信息，否则输出“找不到”信息。

```
str="This is a python book. I think it is a good book."
word=input("请输入要查找的单词: ").strip() #忽略单词两端的空格
if word in str:
    print(f"找到{word}，位置: {str.find(word)}")
else:
    print(f"找不到{word}！")
```

###例 4.3 同例 4.2 中要求

```
#只是考虑忽略用户输入的单词的两端空格，
#查找不区分大小写。本例练习几个字符串函数的使用。
str="This is a python book. I think it is a good book."
word=input("请输入要查找的单词: ")
str=str.upper() #转化为大写
word_upper=word.strip().upper() #忽略单词两端的空格并转换为大写
if word_upper in str:
    print(f"找到{word}，位置: {str.find(word_upper)}")
else:
    print(f"找不到{word}！")
```

例 4.4 使用单分支结构实现五级记分制成绩输出

```
score=int(input("请输入成绩（0-100 分的整数）: "))
if score<60:
    grade=f"{score}分，不及格！"
if 60<=score<70:
    grade=f"{score}分，及格！"
if 70<=score<80:
    grade=f"{score}分，中等！"
if 80<=score<90:
    grade=f"{score}分，良好！"
if 90<=score<=100:
    grade=f"{score}分，优秀！"
print(grade)
```

###例 4.5（a）使用多分支结构（省略 else 子句）实现五级记分制。

```
score=int(input("请输入成绩（0-100 分的整数）: "))
if score<60:
    grade=f"{score}分，不及格！"
```

```
elif score<70:
    grade=f"{score}分，及格！"
elif score<80:
    grade=f"{score}分，中等！"
elif score<90:
    grade=f"{score}分，良好！"
elif score<=100:
    grade=f"{score}分，优秀！"
print(grade)
```

###例 4.5(b) 多分支结构实现五级记分制（考虑条件覆盖和程序健壮性）

```
score=int(input("请输入成绩（0-100 分的整数）："))
if score<0:
    grade=f"输入{score}，分数无效，成绩不能为负数！"
elif score<60:
    grade=f"{score}分，不及格！"
elif score<70:
    grade=f"{score}分，及格！"
elif score<80:
    grade=f"{score}分，中等！"
elif score<90:
    grade=f"{score}分，良好！"
elif score<=100:
    grade=f"{score}分，优秀！"
else:
    grade=f"输入{score}，分数无效，成绩不能大于 100 分！"
print(grade)
```

###例 4.6 使用嵌套分支结构实现五级记分制成绩输出

```
score=int(input("请输入成绩（0-100 分的整数）："))
if score<0:
    grade=f"{score}分数无效，成绩不能为负数！"
else:
    if score<60:
        grade=f"{score}分，不及格！"
    else:
        if score<70:
            grade=f"{score}分，及格！"
        else:
            if score<80:
                grade=f"{score}分，中等！"
            else:
                if score<90:
                    grade=f"{score}分，良好！"
```

```

else:
    if score<=100:
        grade=f"{score}分，优秀！"
    else:
        grade=f"{score}分数无效，不能大于 100 分！"
print(grade)

```

例 4.7 单行 if-else 演示

```

score=50
grade='合格' if score>=60 else '不合格' ##基本形式#if 后面的语句提前到 if 前面去了
print(grade)
###嵌套形式
grade='优秀' if score>=90 else '良好' if score>=80 else '中等' if score>=70 else '及格' if score>=60 else '不及格'
print(grade)
# y=a>60?80:50  #没有类似于 C 语言的三元运算符，异常

```

###例 4.8 使用 while 循环求 1 到 100 的和

```

i=s=0
while i<100:
    i+=1  #注意，循环变量先+1
    s+=i  #注意，不能用 i++, python 没有“++”之类的运算符
print(s)

i,s=1,0
while i<101:
    s+=i  #先累加 i 的值
    i+=1  #循环变量后+1
print(s)

```

###例 4.9 猜数游戏

#系统随机生成一个整数，用户输入一个整数，看是否猜准了。

```

import random                #导入 random 库
number = random.randint(1,100)  #生成随机数
guess = int(input("输入您猜的数:"))
while guess != number:
    if(guess > number):
        print("您猜大了，再小点。")
    else:
        print("您猜小了，再大点。")
    guess = int(input("重新输入:"))
print("赢了。游戏结束。")

```

###例 4.10 输入一个字符串，每个字符连续重复 3 次组成新的字符串

```
word='abcdef'
doubleWord=""
for w in word:
    doubleWord += w*3
print("三倍字符串是：" + doubleWord + "。")
```

###例 4.11 使用 for 循环求 1 到 100 的和

```
s=0
for i in range(0,100):#使用了内置 range 函数
    s+=i+1
print(s)    # 5050
```

###例 4.12 打印 1900-2021 年之间的闰年年份

```
s=0
for year in range(1900,2022):
    if year%4==0 and (year%100!=0 or year%400==0): #判断闰年的条件
        print(year,end=" ") #每个年份用一个空格隔开，使用了 end 参数
```

###例 4.13 break 和 continue 语句的理解

```
for letter in 'Python':
    if letter == 'h':
        break
    print('当前字母为 :' + letter)
print('-----')
for letter in 'Python':
    if letter == 'h':
        continue
    print('当前字母为 :' + letter)
```

###例 4.14 输入学生成绩，

#若成绩小于 0 或者大于 100 则重新输入。打印学生的数量和平均分数。

```
count = 0
sum = 0
while True:
    s = input(f"请输入第{count+1}位同学的分数（按 Q 或 q 结束）:")
    if s.upper() == 'Q':
        print("录入完成，退出")
        break
    if float(s) < 0 or float(s)>100:
```

```

        continue
    count += 1
    sum += float(s)
print(f"学生数: {count}")
print(f"平均成绩为: {sum/count}")

```

###例 4.15 使用循环的 else 子句。判断输入的整数是不是素数

```

n = int(input("请输入一个正整数:"))
for x in range(2, n): #想一下是否可以改为: for x in range(2, n//2), 为什么?
    if n % x == 0:
        print('不是素数',n,'=',x,'*',n//x)
        break #结束循环, 跳转到 "print('结束')" 语句执行
else: #如果循环条件不再满足则执行 else 子句
    print(n, '是一个素数')
print('结束')
#思考: 如果最开始 n 的输入值为-10, 结果会是什么?

```

###例 4.16 判断输入的整数是不是素数, 加入输入数据的合法性判断

```

#下面的 while 循环保证输入的大于 1 的正整数
while True:
    n = input("请输入一个正整数:").strip()#输入的字符串去掉两端的空格
    if n.isdigit():#如果是数字
        print("is number")
        n=int(n)
        if n>1:
            break #结束循环, 跳转到 for 语句执行
        else:
            continue #执行下一次循环, 重新输入 n
    else:
        continue #执行下一次循环, 重新输入 n

#下面的语句判断是不是素数
for x in range(2, n//2): #将 n 改为 n//2, 减少循环次数, 提高效率。为什么?
    if n % x == 0:
        print('不是素数',n,'=',x,'*',n//x)
        break #结束循环, 跳转到 "print('结束')" 语句执行
else: #如果循环条件不再满足则执行 else 子句
    print(n, '是一个素数')
print('结束')

```

#请运行程序, 分别输入 abc、-10、0、1、2、11、20 看一下输出结果。

###例 4.17 判断 2 到 5 的整数是不是素数

```

for n in range(2, 6):

```

```

for x in range(2, n):
    if n % x == 0:
        print(f"{n}={x} * {n/x}")
        break #只作用于内层的 for 循环
    else:
        print(f"{n}是素数")
print("over")

```

例 4.18 打印九九乘法表

#乘法表-for 循环嵌套

```

for i in range(1,10):
    for j in range(1,i+1):
        print(f'{i}×{j}={i*j}',end='\t')
    print()

```

#使用 while 实现九九乘法表

```

i=1
while i<10:
    j=1
    while j<=i:
        print(f'{j}×{i}={i*j}',end='\t')
        j+=1
    print()
    i+=1

```

###例 4.19 打印斐波那契数列 1 1 2 3 5 8 13....打印 30 项,每行打印 5 个

```

f0=f1=1
print(f0,end='\t')
print(f1,end='\t')
for i in range(2,30):
    f2=f0+f1
    if i%5==0:print( )
    print(f2,end='\t')
    f0,f1=f1,f2
else:
    print('\n 斐波那契数列打印完毕。')

```

例 4.20 判断 100-999 之间的水仙花数。水仙花数是指一个 n 位数($n \geq 3$), 它的每个位上的数字的 n 次幂之和等于它本身。

```

import math
for i in range(100,1000):
    a=math.pow(i//100,3)

```

```

b=math.pow((i%100)//10,3)
c=math.pow(i%10,3)
sum=a+b+c
if sum==i:
    print(i,end='\t')

```

```

for i in range(100,1000):
    s=str(i)
    sum=0
    for j in s:
        sum+=math.pow((int(j)), len(s))
    if sum==i:
        print(i,end='\t')

```

例 4.21 著名的一个例子（常规代码）

```

#1-50 之间的数，如果被 3 整除打印'dizz',
#如果被 5 整除打印'fuzz',
#同时被 3 和 5 整除，打印 dizzfuzz,
#其他的数打印这个数本身
#每 5 个一行，每行元素左对齐
s=""
for i in range(1,51):
    if (i-1)%5==0:#每五个换行输出
        print()
    if i%3==0:
        if i%5==0:
            s="dizzfuzz"
        else:
            s="dizz"
    elif i%5==0:
        s="fuzz"
    else:
        s=str(i)
    print(s,end='\t')

```

例 4.21 著名的一个例子（大神代码）

```

#1-50 之间的数，如果被 3 整除打印'dizz',
#如果被 5 整除打印'fuzz',
#同时被 3 和 5 整除，打印 dizzfuzz,
#其他的数打印这个数本身
#每 5 个一行，每行元素左对齐

for i in range(1,51):
    print(('\\n' if (i-1)%5 == 0 else '') +

```

```
('dizz'[i%3*len('dizz')::] +  
  'fuzz'[i%5*len('fuzz')::] or str(i)) , end='\t')
```

结果:

```
1   2   dizz 4   fuzz  
dizz 7   8   dizz fuzz  
11  dizz 13  14  dizzfuzz  
16  17  dizz 19  fuzz  
dizz 22  23  dizz fuzz  
26  dizz 28  29  dizzfuzz  
31  32  dizz 34  fuzz  
dizz 37  38  dizz fuzz  
41  dizz 43  44  dizzfuzz  
46  47  dizz 49  fuzz
```