

## 第二节课 控制结构

控制结构用于控制代码的执行流程。它们决定哪些代码应该执行，什么时候执行，以及执行多少次。

控制结构主要分为**条件语句**和**循环语句**两大类。

### 条件语句

条件语句根据给定的条件执行代码块。如果条件为真（True），则执行特定代码块；如果条件为假（False），则可以执行另一个代码块。

#### --if 基本语法

if condition: 如果这个条件是true代码将执行

#### --if-else 基本语法

if condition: 如果这个条件是true代码将执行

else: 如果这个条件是false代码将执行

#### --if-elif-else

```
In [2]: # 判断是否为成年人
age = 13
if age >= 18:
    print("you are an adult")
else:
    print("you are a teenager")
```

you are a teenager

```
In [4]: # 判断成绩等级
score = 81
if score >= 90:
    print("A")
elif score >= 80:
    print("B")
elif score >= 60:
    print("C")
else:
    print("D")
```

B

---

### 循环语句

#### for & while

#### --for基本语法:

for variable in sequence:

为序列中的每个项目执行代码

--while基本语法:

while condition:

如果符合condition就执行

-- break continue

```
In [7]: # 打印每个水果f的名称
fruits = ['apple', 'banana', 'orange']

for fruit in fruits:
    print(fruit)
```

apple  
banana  
orange

```
In [12]: # 计算1-1000的和
sum = 0
for i in range(1, 1001):
    sum = sum + i
print(sum)
```

500500

```
In [11]: # 计算1-1000的和
sum = 0
i = 1
while i <= 1000:
    sum = sum + i
    i = i + 1
print(sum)

# while 第一步执行完后 i = 2 sum = 1
# 第二次循环后 i = 3 sum = 3

# 计算1到10的和
sum = 0
i = 1
while i <= 10:
    sum = sum + i
    i = i + 1
print(sum)
```

500500

sum (执行后的sum的值) = sum(执行这个语句前的sum的值) + i

这个语句 sum = sum + i 是一个常见的编程操作，用于将变量 i 的值累加到变量 sum 上。

### 语句解释：

sum 和 i 是两个变量。

sum = sum + i 表示将当前 sum 的值加上 i 的值，然后将结果重新赋值给 sum。

### 具体步骤：

读取 sum 的当前值。

读取 i 的当前值。

计算 sum + i 的结果。

将结果赋值回 sum，即更新 sum 的值。

### 具体示例：

假设 sum 的初始值是 0，i 的值是 1，然后我们执行 sum = sum + i：

初始 sum = 0，i = 1

计算 sum + i: 0 + 1 = 1

更新 sum: sum = 1

接下来如果 i 的值是 2，再次执行 sum = sum + i：

当前 sum = 1，i = 2

计算 sum + i: 1 + 2 = 3

更新 sum: sum = 3

这个过程会在循环中反复进行，累加每次循环中的 i 的值。

```
In [15]: # 使用break和continue来输出1-10的奇数
i = 0
while i < 10:
    i = i + 1
    if i % 2 == 0:
        continue
    print(i)
```

1  
3  
5  
7  
9

```
In [1]: # 学生成绩管理系统示例

students = [] # 存储学生信息的列表

# 获取学生数量
num_students = int(input("Enter the number of students: "))

# 输入每个学生的姓名和成绩
```

```
for i in range(num_students):
    name = input("Enter the name of student {}: ".format(i+1))
    score = float(input("Enter the score of {}: ".format(name)))
    students.append({'name': name, 'score': score})

# 计算平均成绩、最高成绩和最低成绩
total_score = 0
max_score = -1
min_score = 101
for student in students:
    total_score += student['score']
    if student['score'] > max_score:
        max_score = student['score']
    if student['score'] < min_score:
        min_score = student['score']

average_score = total_score / num_students

# 输出每个学生的成绩等级
for student in students:
    if student['score'] >= 90:
        grade = 'A'
    elif student['score'] >= 80:
        grade = 'B'
    elif student['score'] >= 70:
        grade = 'C'
    elif student['score'] >= 60:
        grade = 'D'
    else:
        grade = 'F'
    print("Student: {}, Score: {}, Grade: {}".format(student['name'], student['score'], grade))

# 输出统计信息
print("Average Score:", average_score)
print("Highest Score:", max_score)
print("Lowest Score:", min_score)
```

---

**作者:** Yming

**邮箱:** yuemingn@student.unimelb.edu.au

**版权声明:** 本代码仅用于个人学习目的, 未经许可, 不得用于商业用途。