

《CH3程序控制结构》 PPT课件

制作人：
时间：2024年X月

目录

- 第1章 程序控制结构概述
- 第2章 顺序结构
- 第3章 选择结构
- 第4章 循环结构
- 第5章 多重循环结构
- 第6章 总结与展望

• 01

第一章 程序控制结构概述

什么是程序控制结构？

程序控制结构是程序设计中控制程序流程的一种机制。它根据一定的条件或规则，决定程序中各部分的执行顺序，可以分为顺序结构、选择结构和循环结构。程序控制结构的作用是提高程序的执行效率，使代码逻辑更清晰，便于维护和修改。

程序控制结构的重要性

提高程序运行效率

优化算法设计，减少不必要的计算

便于维护和修改

修改代码更容易，减少错误产生

代码逻辑清晰

使程序结构更清晰易懂

程序控制结构的发展历程

早期的程序控制结构

顺序结构

分支结构

经典的程序控制结构

if语句

switch语句

for循环

while循环

现代的程序控制结构

高级语言控制结构

异常处理机制

01

程序控制结构与算法的概念

控制结构影响算法执行效率

02

不同程序控制结构对算法性能的影响

选择合适的结构可以提高算法效率

03

如何选择合适的程序控制结构

根据问题需求和效率考虑选择合适的结构

总结

程序控制结构是程序设计的基础，合理选择和运用不同的控制结构可以提高程序的效率和可维护性，同时也对算法的性能有着重要影响。深入理解和熟练掌握程序控制结构对于编程能力的提升至关重要。

• 02

第2章 顺序结构

什么是顺序结构 ？

顺序结构是一种程序控制结构，按照代码中的顺序依次执行每个语句。程序执行过程中，从上到下逐行执行，具有明确的执行顺序。顺序执行的特点是简单明了，符合人类的思维习惯。

顺序结构的应用场景

日常生活中的 顺序结构

如做早餐、刷牙、
上班等

顺序结构的优 缺点

优点是逻辑清晰，
缺点是不灵活

编程中的顺序 结构

如变量赋值、函数
调用等

01 实例一：打印输出信息

使用print函数输出内容

02 实例二：数据处理操作

对数据进行加减乘除等操作

03 实例三：简单的计算程序

实现简单的算术运算

顺序结构的示例代码

```
def main():
```

```
    print("Hello, World!")
```

```
    num1 = 10
```

```
    num2 = 20
```

```
    result = num1 + num2
```

```
    print("The result is:", result)
```

```
if __name__ ==  
"__main__":  
    main()
```

总结

顺序结构是程序控制结构的基础，通过顺序执行不同的任务，实现了程序的逻辑顺序性。在编程中，合理运用顺序结构能够更好地组织代码，使程序更加清晰易读。

• 03

第3章 选择结构

选择结构的应用场景

根据条件执行不同的代码块

实现多路选择的逻辑

解决实际问题中的分支情况

选择结构的示例 代码

```
def main(): num =  
int(input('Enter a  
number: ')) if num % 2  
== 0: print('The  
number is even.') else:  
print('The number is  
odd.') if __name__ ==  
'__main__': main()
```

选择结构的实例分析

**实例一：判断
用户输入的数
是奇数还是偶
数**

根据余数判断

**实例三：根据
成绩等级输出
评价**

根据不同分数段给
予评价

**实例二：比较
两个数的大小**

基于大小关系进行
比较

选择结构的应用场景

根据条件执行
不同的代码块

灵活处理不同情况

解决实际问题
中的分支情况

适用于解决具体问
题

实现多路选择
的逻辑

分支条件多样

01

实例一：判断用户输入的数是奇数还是偶数

根据余数判断

02

实例二：比较两个数的大小

基于大小关系进行比较

03

实例三：根据成绩等级输出评价

根据不同分数段给予评价

选择结构的实例分析

**实例一：判断
用户输入的数
是奇数还是偶
数**

根据余数判断

**实例三：根据
成绩等级输出
评价**

根据不同分数段给
予评价

**实例二：比较
两个数的大小**

基于大小关系进行
比较

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：
<https://d.book118.com/307103101002006056>