



C语言数据结构教程

C语言基础

1. C语言简介

C语言是一种结构化编程语言，由Dennis Ritchie在1972年为Unix操作系统设计。它简洁、高效，提供了低级访问内存的能力，同时也支持高级编程概念。C语言是许多操作系统、嵌入式系统和应用程序的基础，其影响力延伸至其他编程语言如C++、Objective-C和C#。

C语言的特点包括：- 简洁性：C语言的语法简单，易于学习。- 可移植性：C语言编写的程序可以在不同的操作系统和硬件平台上运行，只需进行少量修改。- 效率：C语言编译后的代码运行效率高，接近汇编语言。- 模块化：C语言支持函数和模块化编程，便于代码的重用和维护。

2. C语言变量和数据类型

在C语言中，变量是用于存储数据的标识符。变量声明包括数据类型和变量名。C语言支持多种数据类型，包括基本类型和复合类型。

2.1 基本数据类型

- 整型：int、short、long、long long
- 字符型：char
- 浮点型：float、double
- 布尔型：_Bool (C99标准)

2.2 复合数据类型

- 数组：用于存储相同类型数据的集合。
- 结构体：用于存储不同类型数据的集合。
- 联合体：用于存储不同类型数据，但同一时间只能存储一种类型的数据。
- 枚举：用于定义一组命名的整型常量。

2.3 示例：变量声明和使用

```
#include <stdio.h>

int main() {
    // 声明变量
    int age = 25;
    char grade = 'A';
    float salary = 5000.50;
    double pi = 3.14159265358979323846;
```

```

    _Bool is_student = true;

    // 输出变量值
    printf("年龄: %d\n", age);
    printf("成绩: %c\n", grade);
    printf("薪水: %.2f\n", salary);
    printf("圆周率: %.10f\n", pi);
    printf("是否是学生: %s\n", is_student ? "是" : "不是");

    return 0;
}

```

3. C语言控制结构

C语言的控制结构用于控制程序的执行流程，包括条件语句、循环语句和跳转语句。

3.1 条件语句

- **if**语句：根据条件执行代码块。
- **if-else**语句：根据条件执行不同的代码块。
- **switch**语句：基于不同条件执行不同的代码块。

3.2 循环语句

- **for**循环：用于已知循环次数的情况。
- **while**循环：在条件为真时执行循环。
- **do-while**循环：至少执行一次循环，然后在条件为真时继续执行。

3.3 跳转语句

- **break**：用于跳出循环。
- **continue**：用于跳过循环中的当前迭代。
- **goto**：用于无条件跳转到程序中的指定位置。

3.4 示例：控制结构的使用

```

#include <stdio.h>

int main() {
    int i, number = 10;

    // if-else语句
    if (number > 0) {
        printf("这是一个正数。\\n");
    }
}

```

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/206041050032010201>