



石家莊鐵道大學  
SHIJIAZHUANG TIEDAO UNIVERSITY

在线开放课程

绪论

# 抽象数据类型的表示与实现

主讲：刘辉

# 目录

---

- ◆ 1 数据类型
- ◆ 2 抽象数据类型
- ◆ 3 抽象数据类型的表示



# 一、数据类型

## ◆ 数据类型

- 相同**特性值的集合**和定义在这个值集上的一组操作的总称

C语言：

**基本数据类型：** char int float double void

**构造数据类型：** 数组、结构体、共用体、文件

## 二、抽象数据类型

- ◆ 抽象数据类型
  - 与计算机实现无关的数据类型

抽象数据类型可以用以下的三元组来表示：

$$\text{ADT} = (\text{D}, \text{R}, \text{P})$$

数据对象

D上的关系集合

D上的操作集合

## 二、抽象数据类型

### ◆ 抽象数据类型常用定义格式

*ADT*抽象数据类型名 {

**数据对象**：〈数据对象的定义〉

**数据关系**：〈数据关系的定义〉

**基本操作**：〈基本操作的定义〉

} *ADT*抽象数据类型名

## 二、抽象数据类型

### ◆ 抽象数据类型常用定义格式—复数的ADT描述

**ADT complex{**

**数据对象:**  $D = \{ e_1, e_2 \mid e_1, e_2 \in \mathbb{R}, \mathbb{R} \text{是实数集} \}$

**数据关系:**  $R = \{ \langle e_1, e_2 \rangle \mid e_1 \text{是复数的实部}, e_2 \text{是复数的虚部} \}$

**基本操作:**

```
Create(&C,x,y); //创建复数C;  
GetReal(C); //返回复数C的实部;  
GetImag(C); //返回复数C的虚部;  
Add(C1,C2); //返回复数C1,C2的和;  
Sub(C1,C2); //返回复数C1,C2的差;  
..... ;
```

**} ADT complex;**

## 三、抽象数据类型的表示与实现

- ◆ 抽象数据类型可以通过**固有的数据类型**（如整型、实型、字符型等）来表示和实现
- ◆ 它类似C语言中的**结构（struct）类型**，但增加了相关的**操作**
- ◆ 教材中用的是**类C语言**（介于伪码和C语言之间）作为描述工具

# 三、抽象数据类型的表示与实现

## ◆ 抽象数据类型表示—C语言

### (1) 预定义常量及类型

//函数结果状态代码

```
#define OK 1
```

```
#define ERROR 0
```

```
#define INFEASIBLE -1
```

```
#define OVERFLOW -2
```

// Status是函数返回值类型，其值是函数结果状态代码。

```
typedef int Status;
```



## 三、抽象数据类型的表示与实现

### ◆ 抽象数据类型表示—C语言

(2)数据元素被约定为ElemType 类型，用户需要根据具体情况，自行定义该数据类型。

(3)算法描述为以下的函数形式：

函数类型 函数名（函数参数表）

{

    语句序列；

}

# 三、抽象数据类型的表示与实现

## ◆ 抽象数据类型表示—C语言

### (4) 内存的动态分配与释放

使用new和delete动态分配和释放内存空间

分配空间 指针变量=new数据类型;

释放空间 delete指针变量;

(5) 赋值语句

(6) 选择语句

(7) 循环语句

## 三、抽象数据类型的表示与实现

### ◆ 抽象数据类型表示—C语言

(8) 使用的结束语句形式有：

函数结束语句 `return`

循环结束语句 `break;`

异常结束语句 `exit`（异常代码）；

(9) 输入输出语句

(10) 扩展函数

# 小结

- 了解抽象数据类型的表示形式
- 了解抽象数据类型表示及C语言实现方式

# 谢谢！