

数组抽象数据类型 (Array ADT)

定义

数组 (Array) 是一种线性数据结构，用于存储一组相同类型的元素，支持通过索引高效访问元素。动态数组可支持扩容。

基本操作

- **初始化**
 - `Array(capacity)`: 创建指定容量的新数组
 - 时间复杂度: $O(1)$
- **拷贝与赋值**
 - `Array(const Array& other)`: 拷贝构造
 - `Array& operator=(const Array& other)`: 赋值操作符
 - 时间复杂度: $O(n)$
- **析构**
 - `~Array()`: 释放内存
- **访问操作**
 - `get(index)`: 获取指定索引的元素，越界抛出 `std::out_of_range`
 - `set(index, value)`: 设置指定索引的元素，越界抛出 `std::out_of_range`
 - 时间复杂度: $O(1)$
- **插入**
 - `insert(index, value)`: 在指定位置插入元素，数组满时抛出 `std::overflow_error`，越界抛出 `std::out_of_range`
 - 时间复杂度: $O(n)$
- **删除**
 - `remove(index)`: 删除指定位置的元素，越界抛出 `std::out_of_range`
 - 时间复杂度: $O(n)$
- **扩容**
 - `extend(enlarge)`: 扩展数组容量，`enlarge <= 0` 时无操作
 - 时间复杂度: $O(n)$
- **查找**
 - `search(value)`: 查找元素首次出现的位置，未找到返回 -1
 - 时间复杂度: $O(n)$
- **其他操作**
 - `size()`: 获取当前元素个数
 - `isEmpty()`: 判断数组是否为空
 - `isFull()`: 判断数组是否已满
 - 时间复杂度: $O(1)$

异常与边界

- 所有越界操作均抛出 `std::out_of_range` 异常
- 插入时数组已满抛出 `std::overflow_error`
- 扩容参数小于等于0时无操作

接口定义（伪代码）

```
interface ArrayADT<T> {  
    // 构造与析构  
    constructor(capacity: number);  
    copyConstructor(other: ArrayADT<T>);  
    assign(other: ArrayADT<T>): ArrayADT<T>;  
    destructor();  
  
    // 基本操作  
    get(index: number): T; // O(1), 越界抛异常  
    set(index: number, value: T): void; // O(1), 越界抛异常  
    insert(index: number, value: T): void; // O(n), 满/越界抛异常  
    remove(index: number): void; // O(n), 越界抛异常  
    extend(enlarge: number): void; // O(n)  
    search(value: T): number; // O(n), 未找到返回-1  
    size(): number; // O(1)  
    isEmpty(): boolean; // O(1)  
    isFull(): boolean; // O(1)  
}
```

空间复杂度

- $O(n)$, n 为数组容量

优点

- 元素访问为常数时间 $O(1)$
- 连续内存，缓存友好
- 支持动态扩容
- 实现简单，接口清晰

局限性

- 插入/删除操作代价高 ($O(n)$)
- 扩容时有内存拷贝开销
- 静态数组未满时会浪费内存

适用场景

- 需要高效随机访问的场景
- 元素数量变化不频繁或可预估的场景

交互式测试（中文版）

本模块附带交互式测试程序，所有命令行交互均为中文，便于中文用户体验和学习。详见 [./test/test_array.cpp](#)。

示例命令：

- `insert 0 10` 在下标0插入10
- `remove 1` 删除下标1的元素
- `set 2 99` 设置下标2为99
- `get 0` 获取下标0的值
- `search 99` 查找99首次出现的下标

- `extend 5` 扩容5
- `size` 当前元素个数
- `isEmpty` 是否为空
- `isFull` 是否已满
- `print` 打印数组内容
- `help` 显示菜单
- `exit` 或 `0` 退出程序

所有异常和错误提示也为中文，便于理解和调试。