Rust数据结构与算法

基本数据结构

eat more fish - 2022-12

目录

线性数据结构

线性数据结构

定义

- 数组、栈、队列、双端队列这一类数据结构都是保存数据的容器、数据项之间的顺序由添加或者删除的顺序决定，一旦数据项被添加，它相对于前后元素一直保持不变，诸如此类的此类的数据结构被称为线性数据结构。

特点

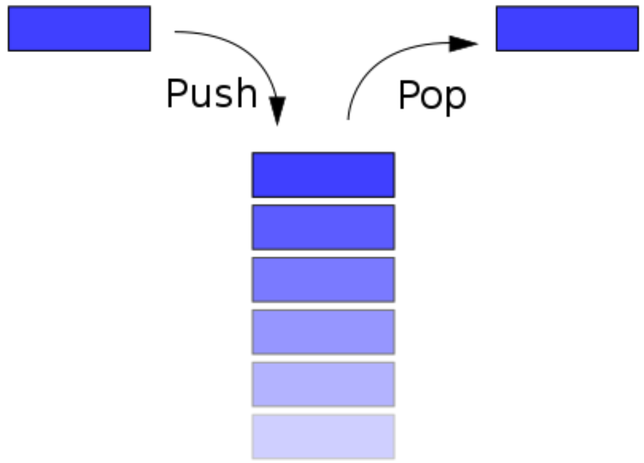
- 有两端：左和右、前和后、顶部和底部...
- 和内存紧密相关，因为内存也是一种线性硬件

栈

栈

定义

堆栈（stack）又称为**栈**或**堆叠**，是计算机科学中的一种抽象资料类型，只允许在有序的线性资料集合的一端（称为堆栈顶端，**top**）进行加入数据（**push**）和移除数据（pop）的运算。因而按照后进先出（**LIFO**, Last In First Out）的原理运作，堆栈常用一维**数组**或**链表**来实现。



特点

1.先入后出，后入先出。
2.除头尾节点之外，每个元素有一个前驱，一个后继。

栈

抽象数据类型

- new() 创建一个空栈，它不需要参数，返回一个空栈。
- push(item) 将数据项 item 添加到栈顶，它需要 item 做参数，不返回任何内容。
- pop() 从栈中删除顶部数据项，它不需要参数，返回数据项，栈被修改。
- peek() 从栈返回顶部数据项，但不会删除它，不需要参数，不修改栈。
- is_empty() 测试栈是否为空，不需要参数，返回布尔值。
- size() 返回栈中数据项的数量，不需要参数，返回一个 usize 型整数。

栈

Rust实现栈

- 在 Rust 中，抽象数据类型的实现多选择创建新的结构体 **struct**，栈操作实现为结构体的函数。
- 这里使用 Vec 这种集合容器来作为栈的底层实现，因为 Rust 中的 Vec 提供了有序集合 机制和一组操作方法，只需要选定 Vec 的哪一端是栈顶部就可以实现其他操作了。
- 因为不知 道插入数据类型，所以采用了泛型数据类型 **T**。

栈

经典实践

- 括号匹配 20. 有效的括号
- 进制转换 504. 七进制数 扩展：任意进制转换
- 前中后缀表达式 剑指 Offer II 036. 后缀表达式
- 中缀转前后缀表达式