1. 单向链表-企业版
   1. 设计：节点只维护指针域，用户数据预留前4个字节由底层使用
   2. 接口：
      1. 初始化链表
      2. 插入链表
      3. 遍历链表
      4. 删除链表
      5. 销毁链表
2. 栈的基本概念
   1. 栈符合 先进后出的数据结构
   2. 入栈 push
   3. 出栈 pop
   4. 栈顶 top
   5. 栈大小 size
   6. 是否为空 isEmpty
   7. 栈底 --- 高地址 栈顶 --- 低地址
   8. 栈是否可以遍历 ---- 不可以
3. 栈的顺序存储
   1. 利用数组模拟出 先进后出数据结构
   2. 数组中首地址 做栈底 方便数组尾部做插入删除
   3. 对外接口
      1. 初始化栈 init
      2. 入栈 push
      3. 出栈 pop
      4. 栈顶 top
      5. 栈大小 size
      6. 是否为空 isEmpty
      7. 销毁栈 destroy
4. 栈的链式存储
   1. 利用链表模拟出 先进后出的数据结构
   2. 头节点端做栈顶 比较方便做入栈和出栈
   3. 对外接口
      1. 初始化栈 init
      2. 入栈 push
      3. 出栈 pop
      4. 栈顶 top
      5. 栈大小 size
      6. 是否为空 isEmpty
      7. 销毁栈 destroy
   4. 测试
5. 栈的应用案例-就近匹配
   1. 从第一个字符开始扫描所有字符
   2. 遇到普通字符 直接忽略
   3. 遇到左括号，入栈
   4. 遇到右括号
      1. 如果栈中有元素 出栈
      2. 如果栈中没有元素 立即停止，并且报错
   5. 当所有字符都扫描完毕，查看栈中内容
      1. 如果是空栈，没有问题
      2. 如果不是空栈，报错
6. 中缀表达式转后缀表达式

遍历中缀表达式中的数字和符号：

* 对于数字：直接输出
* 对于符号：
  + 左括号：进栈
  + 运算符号：与栈顶符号进行优先级比较
* 若栈顶符号优先级低：此符号进栈

（默认栈顶若是左括号，左括号优先级最低）

* 若栈顶符号优先级不低：将栈顶符号弹出并输出，之后进栈
* 右括号：将栈顶符号弹出并输出，直到匹配左括号,将左括号和右括号同时舍弃

遍历结束：将栈中的所有符号弹出并输出

1. 基于后缀表达式运算

遍历后缀表达式中的数字和符号

* + 对于数字：进栈
  + 对于符号：
    - 从栈中弹出右操作数
    - 从栈中弹出左操作数
    - 根据符号进行运算
    - 将运算结果压入栈中

遍历结束：栈中的唯一数字为计算结果