

# Lab: Stacks

### Problem definition:

Most humans prefer to read and write mathematical expressions in infix notation where the operator is written between the two operations, e.g.  $2+3$ . On the other hand, computers deal better with postfix expressions where the operator is written *after* the two operands, e.g.  $23+$ . In order to bridge this gap, you are required to write a function that uses the stack data structure to convert an infix expression into postfix. For example, if the input is “ $2+3$ ”, the output is “ $23+$ ”. If the input is “ $A+B*C$ ”, the output is “ $ABC*+$ ”. For more details, check section 3.6 in the textbook.

# Detailed requirements

- The input is provided as a single Java String
- The output is also a single Java string
- The operators you need to consider
  - +, -, \*, /, %, (, )
- The operands are either single-digit numbers (0-9) or single-letter characters (a-z, A-Z)
- Parentheses ( ) have the highest priority, then multiplication and division, then finally addition and subtraction
- You do NOT need to evaluate the expression; you only need to return the postfix expression