Stacks and Queues

Wednesday, January 31, 2018 7:14 PM

Stacks

- LIFO (last-in-first-out)
- Push and pop
- Applications: Calling functions (state of computation saved on stack), recursion, bracket matching, expression parsing (postfix, reverse Polish notation), maze, matching
- Companion Exception: Underflow

Implementation

- LinkedList: Top of stack = front of BLL
    - Composition/ inheritance method
- Array (top pointer)
- Java API
    - boolean empty(), E peek(), E pop(), E push(E item), int search (Object o) [1-based]
    - Searching uses 2 stacks to take out items and put them back afterwards

Postfix Notation

- Precedence rules: assign level to operator (not unique)
- Conversion of Infix to Postfix
- Write algorithm for Infix -> Prefix

Application: Tarzan swinging tree problem

Queues

- FIFO (first-in-first-out)
- Enqueue and deque
- Applications: BFS of trees, palindromes, simulations

Implementation

- TLL composition/ inheritance
- Circular Array (front and back pointer)
    - front == back can imply full/ empty
    - Keep track of size or leave an empty space
- Java API
    - boolean offer(E o), E peek(), E poll()

Palindromes

- Stack = reverse order
- Queue = preserve order
- Check sequences