Stacks and Queues

Weiss, section 6.6 (pages 258–261)
Linear Structures

• Linear structures are *collections* to which you add an element or from which you remove an element, with one element at a time.
• A to-do list is a like a linear structure. You add a new task to the list. You grab a task that is already on the list.

• Linear structures provide two important operations
  ▪ put( T value): adds an object to the structure.
  ▪ T get(): removes and returns an object from the structure.
• A linear structure has a different policy for implementing *get*.
  ▪ Stacks return an object that is most recently added (last-in, first-out). Example: A set of plates in cafeteria
  ▪ Queues return an object that is least recently added (first-in, first-out). Example: A line at grocery store
Additional Operations

- **isEmpty**: Checks if a structure is empty. This is used as a precondition for *get*.
- **peek**: Returns a value without removing it.
- **size**: Determines the number of values in the structure.
Stack

• Allow access to only the top element

• Operations supported in stack:
  – Add element at top of a stack (push)
  – Extract element from the stack (pop)
  – Inspect the top element of the stack (peek)

• All operations take $O(1)$ time.
An Application for Stack

- Pocket calculator
- Evaluate infix expressions: \( 1 + 2 \times 3 \)
- Calculator needs to know about *operator precedence*.
- Evaluating complex expressions is challenging. Instead of precedence rule:
  - An *infix* expression is converted to a *postfix* expression. Postfix expressions are easily evaluated from left to right.

\[
1 + 2 \times 3 \rightarrow 1 \ 2 \ 3 \ * \ +
\]

- Evaluate postfix expressions using stacks.
  - The top most item becomes the right hand parameter
Queue

- Allow access the least recently (oldest) item
- Model systems when things are placed in line.
- Operations supported in queue:
  - Add element at the back of a queue (enqueue or add)
  - Extract the front element from the queue (dequeue or remove)
  - Inspect the front element of the queue (getFront or element)
- All operations take O(1) time.