cis112-week09: Linked lists

v2025-04-08

Content

Motivation

So far, we considered many data structure such as set, stack, queue. The only tool that we had was arrays. So, we implement our data structures using arrays.

Linked list is another tool to implement data structures.

Some of the data structures that we are familiar with are:

- `Set`
- `List`
- `Stack`
- `Queue`

Introduction

We will develop a queue based on linked list.

Then we will work on a new data structure, called deque.

Web resources

- Generics in Java Tutorials by Oracle.
- Alice and Bob @ wikipedia
- Metasyntactic variable @ wikipedia

Goal

G0. Fill `StudentInfo`

1. Fill your data in `StudentInfo`.

G1. Queue implemented as liked list

In `lab`, `MyNode` is a generic node for doubly linked list. It has `data` field for the content. It also has `next` and `prev` fields to refer to the next node and the previous node in the linked list.

`MyQueue` is a queue implemented as doubly linked list of `MyNode`s. It has 3 fields: `head`, `tail`, `size`. As usual, a new node is added at the `tail` end, and a node is removed from the `head` side. `size` is the number of nodes in the queue.

1. Complete `enqueue` method.
2. Complete `dequeue` method.
3. Test your implementations using `MyQueue_jUnit` in package `week09.lab.ts`.

Challenge

Introduction

A *double-ended queue*, *deque*, is a linear data structure, like stack or queue. There are two ends, called `first` and `last`. Insertion and deletion can be done at both ends by means of methods: `addFirst`, `addLast` and `removeFirst`, `removeLast`.

Note.

- Deque becomes a stack if one uses
  - `addFirst` and `removeFirst` only, or
  - `addLast` and `removeLast` only.
- Deque becomes a queue if one uses
  - only `addLast` and `removeFirst` or
  - only `addFirst` and `removeLast`.

Java API supports deque. `java.util.Deque` is an interface with javadoc. `java.util.ArrayDeque` is an implementation of deque. In `week09.theory`, check `ExampleDeque`, which uses this API.

`java.util.LinkedList` is another implementation of deque. See javadoc.

Steps:

Deque

In `week09.lab`, check `MyDeque`, which is a generic deque. `MyDeque` is implemented using doubly linked list.

1. Complete `enqueueFirst` method.
2. Complete `enqueueLast` method.
3. Complete `dequeueFirst` method.

4. Complete `dequeueLast` method.
5. Test your implementation using `MyDeque_jUnit` in package `week09.lab.ts`.