Collections I.

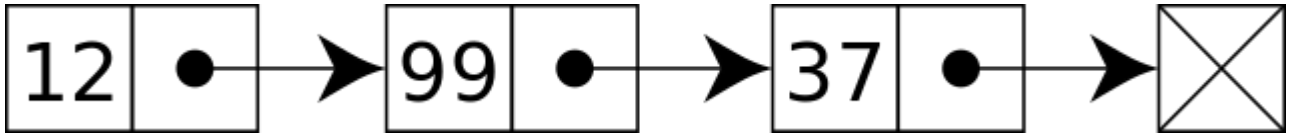
A4B33ALG - ALGORITHMS, SPRING 2017

TOMÁŠ CHAMRA, FEL ČVUT

Basics: Arrays vs. Linked Lists

- **Array**

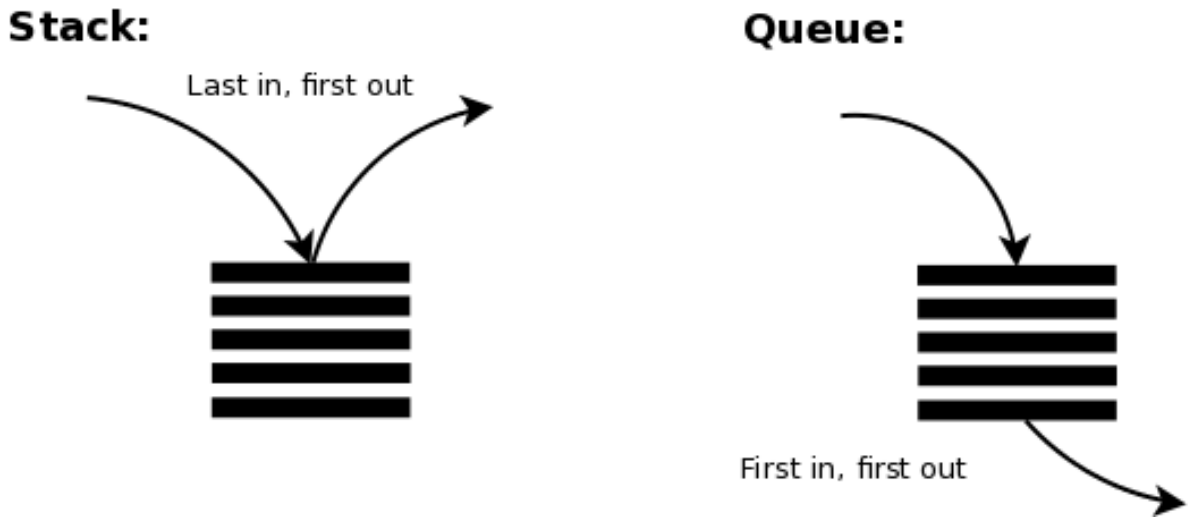
- **Linked List**



Source: https://commons.wikimedia.org/wiki/File:Singly-linked-list.svg

Stacks and Queues

- Abstract data types



Source: http://cs.joensuu.fi/~appro/ics/05-03.php

Collections in Java

Lists

- **List interface**
  - **Array List**
    - Resizable array
  - **Vector**
    - Synchronized list similar to Array List but deprecated
  - **Linked List**
    - Doubly linked list

Stacks and Queues

- **Stack class**
  - Based on Vector – deprecated, do not use
- **Queue interface**
  - **Deque interface** (double ended queue – covers stack as well)
    - **Array Deque**
      - Deque implemented using resizable array
    - **Linked List**
      - Implementation of both List and Deque
  - **Priority Queue**
    - Min Heap based queue (lower goes first)

Containers in C++

Sequence containers

- **Array** (new in C++ 11)
  - Encapsulation of fixed size array
- **Vector**
  - Backed by array, but size can dynamically change
- **Deque**
  - Double ended queue
  - Data is typically split in multiple arrays
- **List**
  - Doubly linked list
- **Forward List** (new in C++ 11)
  - Singly linked list

Container adaptors

- **Stack**
  - LIFO stack as we know it
- **Queue**
  - FIFO queue as we know it
- **Priority Queue**
  - Max Heap based queue (higher goes first)
  - Backed by container with random access (no linked list) and comparator to define the item priority
- *These are not containers themselves, but they need to be backed by some underlying container*

Questions?