

CSCI 136:  
Data Structures  
and  
Advanced Programming

Lecture 24

Trees, part 4

Instructor: Dan Barowy

**Williams**

## Announcements

One-on-one: who can't make it?

IntelliJ IDE: download Community

## Outline

Implicit data structures

IntelliJ IDEA

Priority queues

Implicit Data Structures

## Recall: binary search tree

A **binary search tree** is a binary tree that maintains the **binary search property** as elements are added or removed. In other words, the **key** in each node:

- must be  $\geq$  any **key** stored in the left subtree, and
- must be  $\leq$  any **key** stored in the right subtree.

As with other ordered structures, order is maintained **on insertion**.

## BST is an ADT

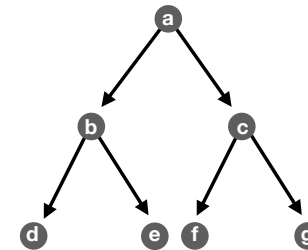
Do we actually need a **tree** to store a **tree**?

No. We can use an **implicit data structure** instead.

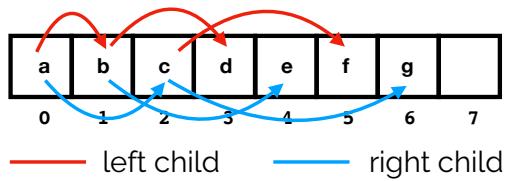
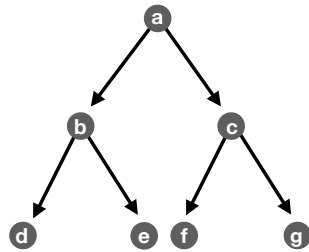
## Implicit data structure

A **implicit data structure** or **space-efficient data structure** is a data structure that stores only **necessary** information. Instead of explicitly representing relationships between elements of the structure using references, an implicit structure **uses the relative positions of elements**.

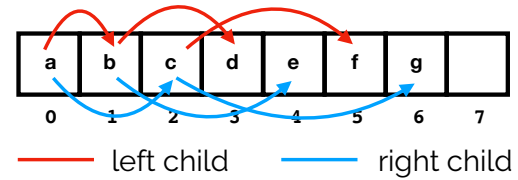
## Implicit binary tree



## Implicit binary tree



## Implicit relationship



$$\text{leftChild}(i) = 2 \times i + 1$$

$$\text{rightChild}(i) = 2 \times i + 2$$

$$\text{parent}(i) = \lfloor (i - 1) / 2 \rfloor$$

## Implicit Binary Search Tree

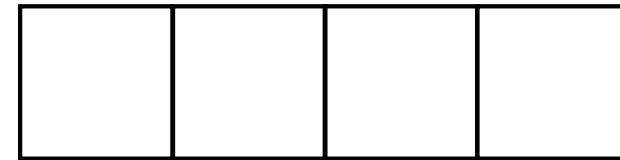
Let's implement an implicit BST.

Priority Queues

## Priority Queue

A **priority queue** is an abstract data type that returns the elements in **priority order**. Under priority ordering, an element **e** with a higher priority (an integer) is returned before all elements **L** having lower priority, even if that **e** was enqueued after all **L**. When any two elements have **equal priority**, they are returned in **first-in, first-out order** (i.e., in the order in which they were enqueued).

## Priority Queue



Ordinary letter



Blue letter

## Priority Queue

insert



0

1

2

3



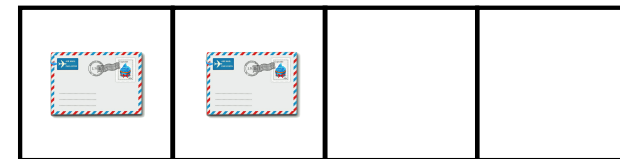
Ordinary letter



Blue letter

## Priority Queue

insert



0

1

2

3



Ordinary letter



Blue letter

# Priority Queue

insert



0

1

2

3



Ordinary letter



Blue letter

# Priority Queue

extract



0

1

2

3



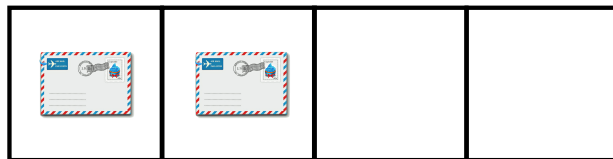
Ordinary letter



Blue letter

# Priority Queue

extract



0

1

2

3



Ordinary letter



Blue letter

# Priority Queue

extract



0

1

2

3



Ordinary letter



Blue letter

## Priority Queue

blue letters: insert



0

1

2

3



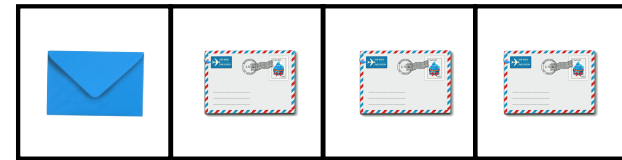
Ordinary letter



Blue letter

## Priority Queue

blue letters: extract



0

1

2

3



Ordinary letter



Blue letter

## Recap & Next Class

### Today we learned:

Implicit data structures

IntelliJ IDEA

Priority queues

### Next class:

Heaps