

# Definitions used for the Midterm Exams 132—all sections

UMD CS Department

## 1 Allowed definitions for classes used on this exam.

Tear this page off of the exam packet, if desired, and use it as a handy reference while completing this exam.

### Common Definition(s)

Use the following definition for questions 1 through 3:

```
public class BinaryTree< T > {  
    // public interface:  
    public boolean isEmpty();  
    public boolean isLeaf();  
    public T getValue();  
    public int height();  
    public BinaryTree< T > getLeft();  
    public BinaryTree< T > getRight();  
}
```

### Ordering Relations

Unless told otherwise, assume that any references to Binary Search Trees are ordered such that their left contains elements less than or equal to the root, and the right contains elements greater than the root. All heaps are Max-Heaps, unless told otherwise.

### Provided methods, classes, etc.

You should assume that each of these methods work as described when answering any questions regarding Binary Trees on this exam. Do not assume that any other methods exist unless you defined them, using these methods. Do **not add** any properties to the class definition of `BinaryTree`.

| Method Signature                            | Description  |
|---|--|
| <code>boolean isEmpty()</code>              | Returns <code>true</code> if this <code>BinaryTree</code> is empty.  |
| <code>boolean isLeaf()</code>               | Returns <code>true</code> if this <code>BinaryTree</code> has empty Binary Trees for its left and right siblings.  |
| <code>T getValue()</code>                   | Returns the value on this <code>BinaryTree</code> object; note, an error results if this tree is empty.  |
| <code>int height()</code>                   | Returns the longest path from the root of this <code>BinaryTree</code> to its deepest (most remote) leaf. Note: the <code>height()</code> of an empty tree is 0. |
| <code>BinaryTree&lt;T&gt; getLeft()</code>  | Returns the left child. Note: calling this on an empty tree throws an exception.   |
| <code>BinaryTree&lt;T&gt; getRight()</code> | Returns the right child. Note: calling this on an empty tree throws an exception.  |

Table 1: Binary Tree Operations