Name :_____

1. Imagine that you start with an empty min-heap and add the following values one-by-one, in the order shown:
   6, 4, 8, 5, 2

   a. Draw the resulting heap as a TREE:

   b. Show how the tree would look after removeSmallest() is called once:

2. Implement the methods below based on the following Java class definitions. You may not add any instance variables or static variables. **Non-recursive solutions will receive no credit. You may use auxiliary methods.**

```java
public class BinarySearchTree <K extends Comparable<K>, V> {
        private class Node {
            private K key;
            private V data;
            private Node left, right;
            public Node(K key, V data) {
                this.key = key;
                this.data = data;
            }
        }

        private Node root;

        public V get(K key) {      /* YOU MUST IMPLEMENT THIS METHOD */ }
        public void printKeys() {  /* YOU MUST IMPLEMENT THIS METHOD */ }
}
```

a. Define a **recursive** method **get(K key)** that returns the data value associated with the given key. If the key is not present in the tree then return null. Your algorithm must search efficiently, without visiting any unnecessary nodes.

b. Define a **recursive** method *printKeys()* that uses a **post-order traversal** to print all of the keys in the tree.