

CS35 Data Structures and Algorithms
Practice Quiz 3, Fall 2015

Name: _____

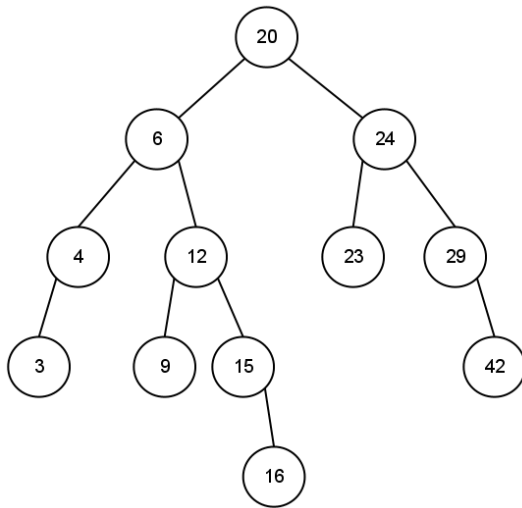
1. Consider this sequence of items for the questions below:

Items (in order of insertion): 27, 29, 3, 11, 7, 1

- (a) Draw the **AVLTree** resulting from the insertion of the items as keys. You may assume the value and key are the same.

- (b) What is the height of the above tree?

2. Consider the following AVL tree:



(a) Write down keys of all nodes visited when searching if the tree contains the key 25.

(b) Draw the AVL tree produced by removing the key 23 from this tree. Clearly show your intermediate work if you want partial credit. Clearly indicate the final tree for your solution.

3. The following is a skeleton for an implementation of a PriorityQueue using a Balanced BST. Part of the declaration for AVLTree is provided for reference, which you should use in your solution. Fill in the methods below to give an implementation of a PriorityQueue that uses an AVLTree to store the values.

```
template <typename K, typename V>
class AVLTree : public BST<K,V> {
public:
    AVLTree();
    ~AVLTree();
    K getMinKey();
    void remove(K key);
    void insert(K key, V value);
};

template <typename P, typename V>
class BBSTPQ : public PriorityQueue<P, V> {
private:
    BST<P,V>* tree;
public:
    BBSTPQ();
    ~BBSTPQ();
    V removeMin();
    V getMin();
    void insert(P priority, V value);
};
```

```
template <typename P, typename V>
BBSTPQ<P,V>::BBSTPQ() {
```

```
}
```

```
template <typename P, typename V>
~BBSTPQ<P,V>::BBSTPQ() {
```

```
}
```

```
template <typename P, typename V>
V BBSTPQ<P,V>::removeMin() {
```

```
}
```

```
template <typename P, typename V>
V BBSTPQ<P,V>::getMin() {
```

```
}
```

```
template <typename P, typename V>
void BBSTPQ<P,V>::insert(P priority, V value) {
```

```
}
```

4. What is the runtime of each method in this implementation?