

## Solutions to Test 2

**Question 1.** (18 PTS.) State the (worst-case) running time of:

1. Insert in a queue implemented using an array.

$O(1)$

2. Evaluating a fully-bracketed expression using the stack-based algorithm.

$O(n)$

3. Inorder traversal of a binary search tree.

$O(n)$

4. Delete in a red-black tree.

$O(\log n)$

**Question 2.** (18 PTS.) Waseme is implementing a stack that stores chars using a linked list. The code starts:

```
struct CNode { char val; CNode * next; };
class CharStack {
private:
    CNode * top;
public:
    const static char UNDERFLOW = '?';
...

```

Give a suitable implementation of the pop function.

```
char CharStack::pop( ) {
```

```
    if ( top==nullptr ) {
        return UNDERFLOW;
    }
    else {
        char answer = top -> val;
        CNode * hold = top;
        top = top -> next;
        delete hold;
        return answer;
    }
}
```

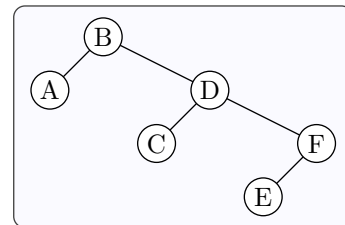
**Question 3.** (16 PTS.) Wayne has a binary tree with 6 nodes. When he does an inorder traversal, the order of the nodes is

A, B, C, D, E, F

When he does a postorder traversal, the order of the nodes is

A, C, E, F, D, B

Draw the tree.



**Question 4.** (18 PTS.) Wojtek created a binary tree class and a method called `conundrum`.

```
struct BTNode { BTNode *left, *right};
class BinaryTree {
    BTNode *root;

    bool BinaryTree::conundrum( ) {
        return puzzling(root);
    }
    bool BinaryTree::puzzling( BTNode *curr) {
        if( curr==nullptr )
            return false;
        else if( (curr->left == nullptr) && (curr->right == nullptr) )
            return true;
        else
            return (puzzling( curr->left) != puzzling( curr->right));
    }
}
```

(a) Draw one tree where `conundrum` returns true, and one where `conundrum` returns false.

True: A tree with just a root node

False: A tree with root node and two children

(b) When does `conundrum` return true? That is, what property of the tree does it test for?

Returns true if and only if has an odd number of leaves

**Question 5.** (14 PTS.) Draw a red–black tree containing the five data values 1 through 5 and which has as many red nodes as possible.

Several answers. Each has Black root with two Black children and then two Red grandchildren.

**Question 6.** (16 PTS.) The following **array** represents a heap.

1	3	5	6	11	8	13
---	---	---	---	----	---	----

Draw the final **tree** after the following two operations **in succession** (using standard algorithm):

(a) INSERT(4)

(b) EXTRACTMIN()

