Anuj Sampat
asampat@bu.edu

CS 566 Programming Assignment 2

Problem

Implement the binary search tree operations for node insert, node delete and post order traversal.

Solution

The node insert, node delete and post order tree traversal operations were implemented using the C language. The following data structure was used to represent each node of the tree:

```
typedef struct node_t
{
        int key;
        struct node_t * parent;
        struct node_t * left;
        struct node_t * right;
} NODE;
```

The 'key' field in the node stores the value input by a user for a node insert operation. The size of this key is limited to the maximum size of the int data type.

The following files have been provided with this report:

Sampat_Anuj_p2.c: The C implementation of the tree operations.

Sampat_Anuj_p2.out: The user executable file for the tree operations.

The C files were compiled using gcc version 3.4.6 on a Red Hat Linux 3.4.6-9 system using the command:

```
$ cc Sampat_Anuj_p2.c -o Sampat_Anuj_p2.out
```

The executable file can be run using the command:

```
$ Sampat_Anuj_p2.out
```

On execution, a main menu will appear that will allow the user to carry out one of the node insert, node delete, post order traversal or exit operations.