CS 32 – UCLA – Homework 5

1.a.)

1.b.) **AFTER DELETING 30**

50

60

20

10

40

70

25

80

65

75

15

35

50

60

20

10

40

70

30

80

65

75

15

35

25

**AFTER DELETING 20**

1.c.)

**PREORDER (NLR):** 50 20 10 15 40 30 25 35 60 70 65 80 75

**INORDER (LNR):** 10 15 20 25 30 35 40 50 60 65 70 75 80

**POSTORDER (LRN):** 15 10 25 35 30 40 20 65 75 80 70 60 50

2.a)

7

5

3

1

2

4

50

60

15

10

40

70

25

80

65

75

35

2.b.)

7

3

5

1

2

4

Count: 6

2.c.)

5

3

4

1

2

Count: 5

3.a.)

struct BSTNode

{

Node(int value, BSTNode\* parent)

: m\_value(value), l\_child(nullptr), r\_child(nullptr), m\_parent(parent)

{}

int m\_value;

BSTNode\* l\_child;

BSTNode\* r\_child;

BSTNode\* m\_parent;

};

3.b.)

void insert(BSTNode\* curr, int value, BSTNode\* parent)

{

insertHelper(curr, value, nullptr);

}

void insertHelper(BSTNode\* curr, int value, BSTNode\* parent)

{

if(curr is nullptr)

create a new BSTNode with the data as the value, the parent pointer being set to parent

and the left and right children to be set as nullptr

else if(curr's value is less than the value passed in)

insertHelper(curr's left child, value, curr)

else if(curr's value is greater than the value passed in)

insertHelper(curr's right child, value, curr)

}

4.a.) O(C+S)

4.b.) O(log(C) + S)

4.c.) O(log(C) + log(S))

4.d.) O(1 + log(S)) = O(log(S))

4.e.) O(1+1) = O(1)

4.f.) O(log(C) + S)

4.g.) O(1+S) = O(S)

4.h.) O(Clog(S))