**DSA Assignnent 3 Task 1**

In this assignment Task 1, I create 2 Binary Search Tree (BST) structures.

One BST is normal BST which is used for one genetic type . This BST may or may not add duplicate value, depend on the parameter. This BST is in NormalBST package

Another BST is a Tree Map which implement Map Interface and store key and value .The Tree Map only store NodeTree with unique Key. This TreeMap is in TreeMap package.

In order to keep all data elements of the same key, Node of TreeMap contain of unique key and normal BST of value. Different keys can have same values however value Collection of one key does not allow duplicate value.

The main class is Assignment3\_Task1 class which is in Assignment3\_Task1 package.

The program has all functions the assignment requied. If user press 1, user add key, value pair at one time in required format . if user press 2, user can find collection of all values which belong to that key. If that key does not exist , the program will inform user. If user press 3 , the program show treeHeight . If user press 4, the program print out the Key Set of Tree Map. If user press 5, the program print out the Collection of values of Tree Map. If user press 6, the program will print all entry Set (Node Tree) of Tree Map. If user press 7, the program will be terminated.

**Sample output**

Please select an option

1. Add key and value

2. Get value from key

3. Get Tree Height

4. Get Key Set

5. Get Values Collection

6. Get Entry Set

7. Quit Program

1

Please enter a key and value in format: key,value

20,30

The new Key [20] has been mapped to the Value [30]

Please select an option

1. Add key and value

2. Get value from key

3. Get Tree Height

4. Get Key Set

5. Get Values Collection

6. Get Entry Set

7. Quit Program

1

Please enter a key and value in format: key,value

20,40

The existing Key [20] has been mapped to the Value [40]

Please select an option

1. Add key and value

2. Get value from key

3. Get Tree Height

4. Get Key Set

5. Get Values Collection

6. Get Entry Set

7. Quit Program

1

Please enter a key and value in format: key,value

30,50

The new Key [30] has been mapped to the Value [50]

Please select an option

1. Add key and value

2. Get value from key

3. Get Tree Height

4. Get Key Set

5. Get Values Collection

6. Get Entry Set

7. Quit Program

1

Please enter a key and value in format: key,value

70,80

The new Key [70] has been mapped to the Value [80]

Please select an option

1. Add key and value

2. Get value from key

3. Get Tree Height

4. Get Key Set

5. Get Values Collection

6. Get Entry Set

7. Quit Program

1

Please enter a key and value in format: key,value

65,92

The new Key [65] has been mapped to the Value [92]

Please select an option

1. Add key and value

2. Get value from key

3. Get Tree Height

4. Get Key Set

5. Get Values Collection

6. Get Entry Set

7. Quit Program

2

Please enter a key

34

This Key is not existed in the data

Please select an option

1. Add key and value

2. Get value from key

3. Get Tree Height

4. Get Key Set

5. Get Values Collection

6. Get Entry Set

7. Quit Program

2

Please enter a key

20

Collection of values which associated with the Key [20]

30

40

Please select an option

1. Add key and value

2. Get value from key

3. Get Tree Height

4. Get Key Set

5. Get Values Collection

6. Get Entry Set

7. Quit Program

3

Tree Height is 3

Please select an option

1. Add key and value

2. Get value from key

3. Get Tree Height

4. Get Key Set

5. Get Values Collection

6. Get Entry Set

7. Quit Program

4

The Set of All Keys

20

30

65

70

Please select an option

1. Add key and value

2. Get value from key

3. Get Tree Height

4. Get Key Set

5. Get Values Collection

6. Get Entry Set

7. Quit Program

5

The Collection of all Values

30

40

50

80

92

Please select an option

1. Add key and value

2. Get value from key

3. Get Tree Height

4. Get Key Set

5. Get Values Collection

6. Get Entry Set

7. Quit Program

6

All the Entry Set (NodeTree)

For Key: [20] :

30

40

For Key: [30] :

50

For Key: [65] :

92

For Key: [70] :

80