

Sample output PR3

```
/*
AVL Tree Implementation
A.Insert an integer to tree and show the balanced tree at each insertion
B.Display the balanced tree and show inorder traversal
C.Exit
=A
Enter integer to be added to tree: 15
Value inserted successfully
```

Root 15<

```
AVL Tree Implementation
A.Insert an integer to tree and show the balanced tree at each insertion
B.Display the balanced tree and show inorder traversal
C.Exit
= A
Enter integer to be added to tree: 18
Value inserted successfully
```

18<

Root 15<

```
AVL Tree Implementation
A.Insert an integer to tree and show the balanced tree at each insertion
B.Display the balanced tree and show inorder traversal
C.Exit
= A
Enter integer to be added to tree: 10
Value inserted successfully
```

18<

Root 15<

10<

```
AVL Tree Implementation
A.Insert an integer to tree and show the balanced tree at each insertion
B.Display the balanced tree and show inorder traversal
C.Exit
= A
Enter integer to be added to tree: 7
Value inserted successfully
```

18<

Root 15<

10<

7<

```
AVL Tree Implementation
A.Insert an integer to tree and show the balanced tree at each insertion
B.Display the balanced tree and show inorder traversal
C.Exit
= A
```

```
Enter integer to be added to tree: 57
Value inserted successfully
```

57<

Root 15<

18<

10<

7<

```
AVL Tree Implementation
A.Insert an integer to tree and show the balanced tree at each insertion
B.Display the balanced tree and show inorder traversal
```

C.Exit

= A

Enter integer to be added to tree: 6

Value inserted successfully

```

                    57<
                18<
Root 15<
```

```

                    10<
                7<
                    6<
```

AVL Tree Implementation

A.Insert an integer to tree and show the balanced tree at each insertion

B.Display the balanced tree and show inorder traversal

C.Exit

= A

Enter integer to be added to tree: 13

Value inserted successfully

```

                    57<
                18<
Root 15<
                    13<
                    10<
                7<
                    6<
```

AVL Tree Implementation

A.Insert an integer to tree and show the balanced tree at each insertion

B.Display the balanced tree and show inorder traversal

C.Exit

= A

Enter integer to be added to tree: 12

Value inserted successfully

```

                    57<
                18<
Root 15<
                    13<
                    12<
                    10<
                7<
                    6<
```

AVL Tree Implementation

A.Insert an integer to tree and show the balanced tree at each insertion

B.Display the balanced tree and show inorder traversal

C.Exit

= A

Enter integer to be added to tree: 9

Value inserted successfully

```

                    57<
                18<
Root 15<
                    13<
                    12<
                10<
                    9<
                    7<
                    6<
```

AVL Tree Implementation

A.Insert an integer to tree and show the balanced tree at each insertion

B.Display the balanced tree and show inorder traversal

C.Exit

= A

Enter integer to be added to tree: 65

Value inserted successfully

```

                    65<
                57<
                18<
Root 15<
                12<
                13<
            10<
                9<
                7<
                6<

```

AVL Tree Implementation

A.Insert an integer to tree and show the balanced tree at each insertion

B.Display the balanced tree and show inorder traversal

C.Exit

= A

Enter integer to be added to tree: 19

Value inserted successfully

```

                    65<
                57<
                19<
                18<
Root 15<
                12<
                13<
            10<
                9<
                7<
                6<

```

AVL Tree Implementation

A.Insert an integer to tree and show the balanced tree at each insertion

B.Display the balanced tree and show inorder traversal

C.Exit

= A

Enter integer to be added to tree: 16

Value inserted successfully

```

                    65<
                57<
                19<
                18<
                16<
Root 15<
                12<
                13<
            10<
                9<
                7<
                6<

```

AVL Tree Implementation

A.Insert an integer to tree and show the balanced tree at each insertion

B.Display the balanced tree and show inorder traversal

C.Exit

= A

Enter integer to be added to tree: 23

Value inserted successfully

```

                                65<
                                57<
                                23<
                                19<
                                18<
                                16<
Root 15<
                                13<
                                12<
                                10<
                                9<
                                7<
                                6<

```

AVL Tree Implementation

A.Insert an integer to tree and show the balanced tree at each insertion

B.Display the balanced tree and show inorder traversal

C.Exit

= B

```

                                65<
                                57<
                                23<
                                19<
                                18<
                                16<
Root 15<
                                13<
                                12<
                                10<
                                9<
                                7<
                                6<

```

Preorder: {15, 10, 7, 6, 9, 12, 13, 19, 18, 16, 57, 23, 65, }

AVL Tree Implementation

A.Insert an integer to tree and show the balanced tree at each insertion

B.Display the balanced tree and show inorder traversal

C.Exit

= C

exit

*/