Writeup (Aastha Bhatt)

Compile

To run the program, unzip PA1.zip to a folder and open that folder from the command prompt/terminal. Then, run the command: python main.py *or* python3 main.py

Supplementary Methods

Testing TREE_SEARCH()

When searching for '128' in the tree resulting from 'test3.csv' the node <__main__.newNode object at 0x7fbae80618e0> is returned and when searching for -1 and 129 (values that do not exist), None and None is returned respectively.

Testing TREE_MAXIMUM()

The node that is returned when calling the max method in BSTs resulting from 'test1.csv', 'test2.csv', and 'test3.csv' hold keys that are 128, 128, and 128 respectively.

Testing TREE_MINIMUM()

The node that is returned when calling the min method in BSTs resulting from 'test1.csv', 'test2.csv', and 'test3.csv'hold keys that are 0, 0, and 0 respectively.

Deleting from BST

Test Cases: TREE_DELETE(BST3, 6), TREE_DELETE(BST3, 37), and TREE_DELETE(BST3,77)

Original pre-order traversal of BST3:

PREORDER traversal of BST3: 105 94 91 70 36 9 0 7 6 2 1 3 5 4 8 31 29 28 13 10 11 12 23 21 18 14 17 16 15 19 20 22 26 25 24 27 30 32 35 33 34 44 43 40 37 39 38 42 41 66 60 56 53 49 45 46 47 48 52 51 50 55 54 58 57 59 62 61 64 63 65 69 67 68 90 81 78 75 72 71 74 73 77 76 79 80 89 87 82 83 85 84 86 88 92 93 96 95 98 97 100 99 102 101 103 104 115 107 106 114 108 110 109 112 111 113 126 122 120 119 116 118 117 121 125 123 124 127 128

Pre-order traversal of BST3 after calling the above methods:

Preorder Traversal of BST3 after deleting 6: 105 94 91 70 36 9 0 7 2 1 3 5 4 8 31 29 28 13 10 11 12 23 21 18 14 17 16 15 19 20 22 26 25 24 27 30 32 35 33 34 44 43 40 37 39 38 42 41 66 60 56 53 49 45 46 47 48 52 51 50 55 54 58 57 59 62 61 64 63 65 69 67 68 90 81 78 75 72 71 74 73 77 76 79 80 89 87 82 83 85 84 86 88 92 93 96 95 98 97 100 99 102 101 103 104 115 107 106 114 108 110 109 112 111 113 126 122 120 119 116 118 117 121 125 123 124 127 128

Preorder Traversal of BST3 after deleting 37: 105 94 91 70 36 9 0 7 2 1 3 5 4 8 31 29 28 13 10 11 12 23 21 18 14 17 16 15 19 20 22 26 25 24 27 30 32 35 33 34 44 43 40 39 38 42 41 66 60 56 53 49 45 46 47 48 52 51 50 55 54 58 57 59 62 61 64 63 65 69 67 68 90 81 78 75 72 71 74 73 77 76 79 80 89 87 82 83 85 84 86 88 92 93 96 95 98 97 100 99 102 101 103 104 115 107 106 114 108 110 109 112 111 113 126 122 120 119 116 118 117 121 125 123 124 127 128

Preorder Traversal of BST3 after deleting 77: 105 94 91 70 36 9 0 7 2 1 3 5 4 8 31 29 28 13 10 11 12 23 21 18 14 17 16 15 19 20 22 26 25 24 27 30 32 35 33 34 44 43 40 39 38 42 41 66 60 56 53 49 45 46 47 48 52 51 50 55 54 58 57 59 62 61 64 63 65 69 67 68 90 81 78 75 72 71 74 73 76 79 80 89 87 82 83 85 84 86 88 92 93 96 95 98 97 100 99 102 101 103 104 115 107 106 114 108 110 109 112 111 113 126 122 120 119 116 118 117 121 125 123 124 127 128

From the output generated, it can be confirmed that the TREE_DELETE method works.

Preorder, Postorder, and Inorder Traversals

The following is the output generated when calling PREORDER_TREE_WALK(BST3), POSTORDER_TREE_WALK(BST3) AND INODER_TREE_WALK(BST3):

PREORDER traversal of BST3: 105 94 91 70 36 9 0 7 6 2 1 3 5 4 8 31 29 28 13 10 11 12 23 21 18 14 17 16 15 19 20 22 26 25 24 27 30 32 35 33 34 44 43 40 37 39 38 42 41 66 60 56 53 49 45 46 47 48 52 51 50 55 54 58 57 59 62 61 64 63 65 69 67 68 90 81 78 75 72 71 74 73 77 76 79 80 89 87 82 83 85 84 86 88 92 93 96 95 98 97 100 99 102 101 103 104 115 107 106 114 108 110 109 112 111 113 126 122 120 119 116 118 117 121 125 123 124 127 128

POSTORDER traversal of BST3: 1 4 5 3 2 6 8 7 0 12 11 10 15 16 17 14 20 19 18 22 21 24 25 27 26 23 13 28 30 29 34 33 35 32 31 9 38 39 37 41 42 40 43 48 47 46 45 50 51 52 49 54 55 53 57 59 58 56 61 63 65 64 62 60 68 67 69 66 44 36 71 73 74 72 76 77 75 80 79 78 84 86 85 83 82 88 87 89 81 90 70 93 92 91 95 97 99 101 104 103 102 100 98 96 94 106 109 111 113 112 110 108 114 107 117 118 116 119 121 120 124 123 125 122 128 127 126 115 105

INORDER traversal of BST3: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128