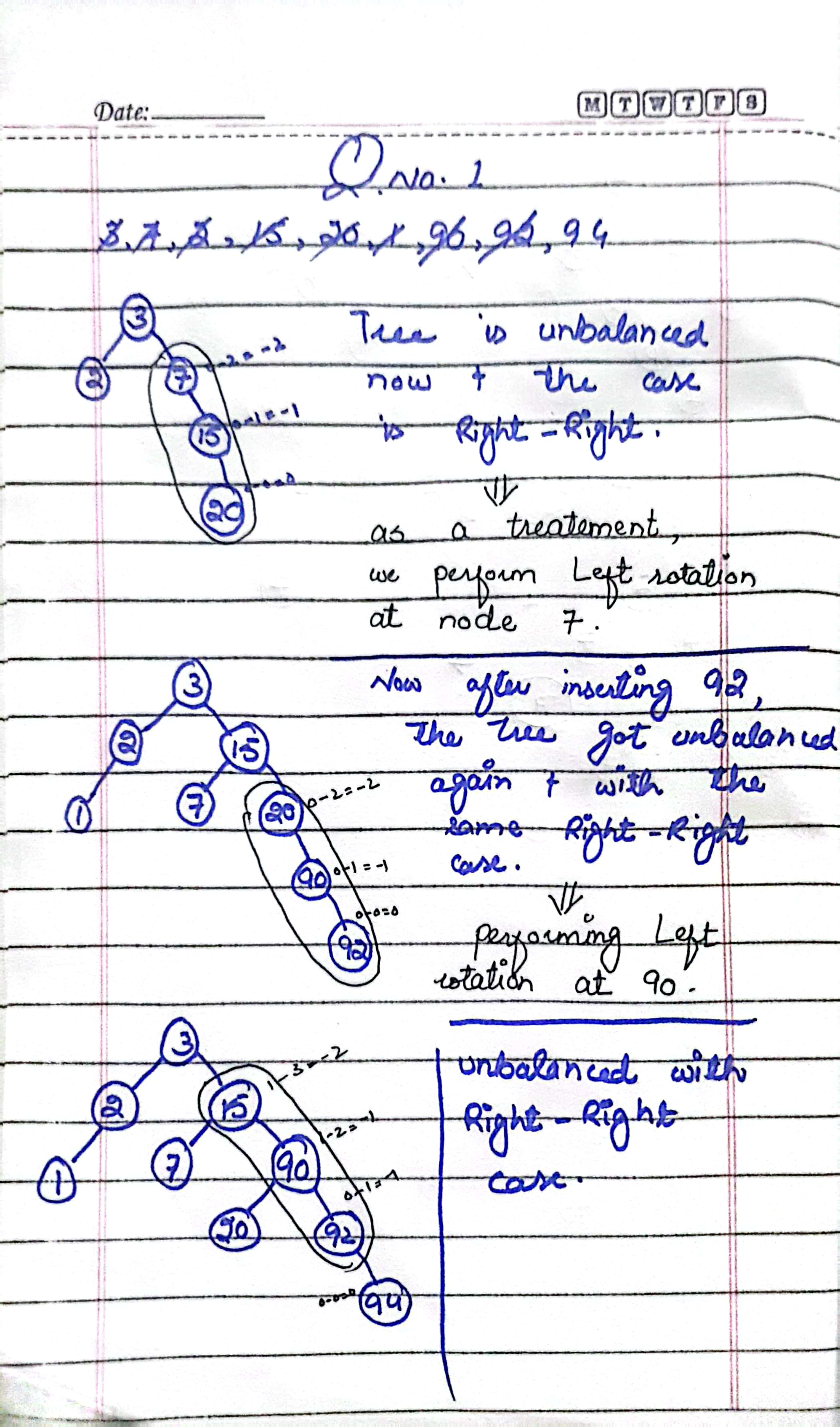
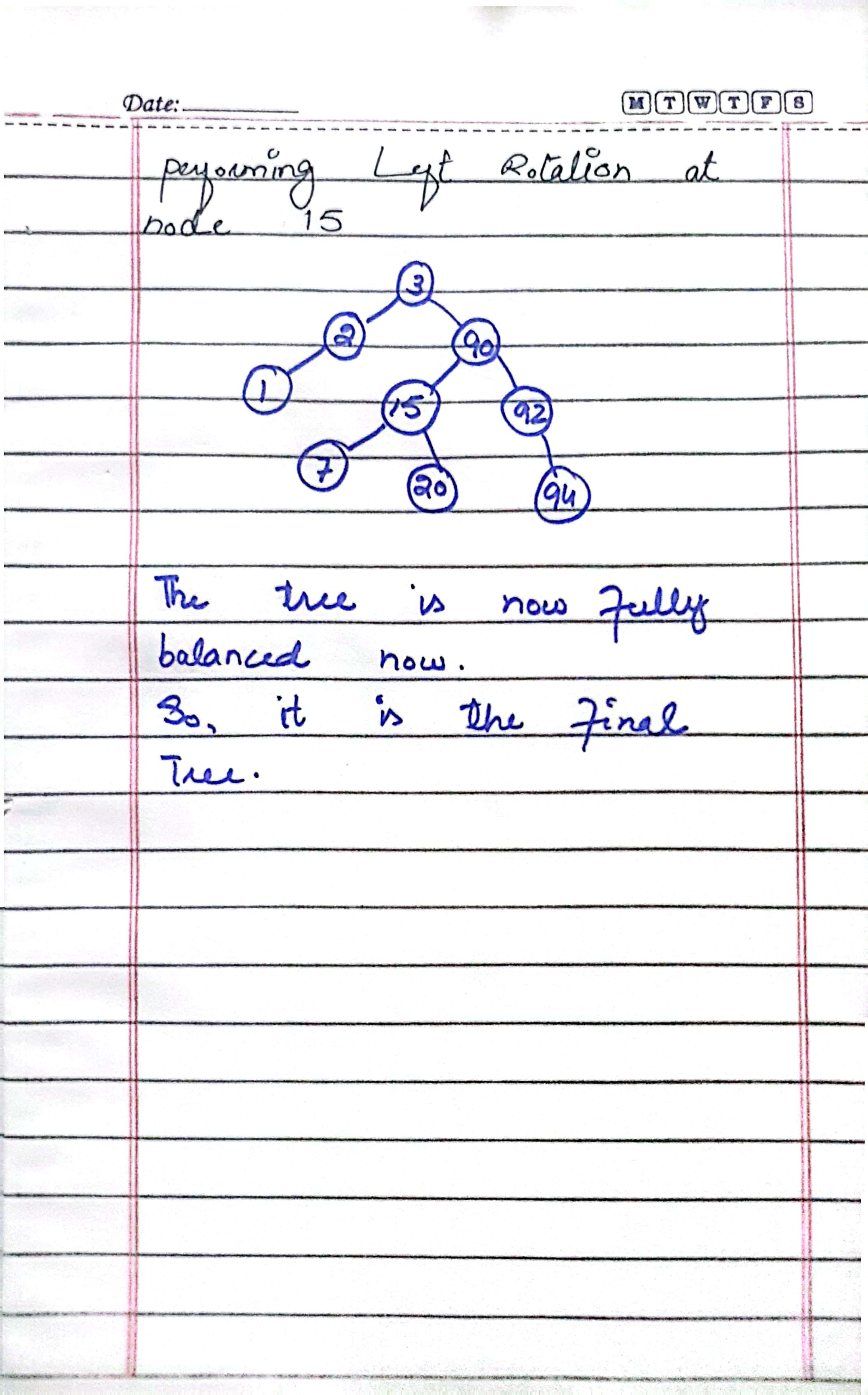
|  |  |
| --- | --- |
| **Name** | **Taaha Hussain Khan** |
| **Roll No** | **L1F21BSCS0917** |
| **Section** | **D12** |

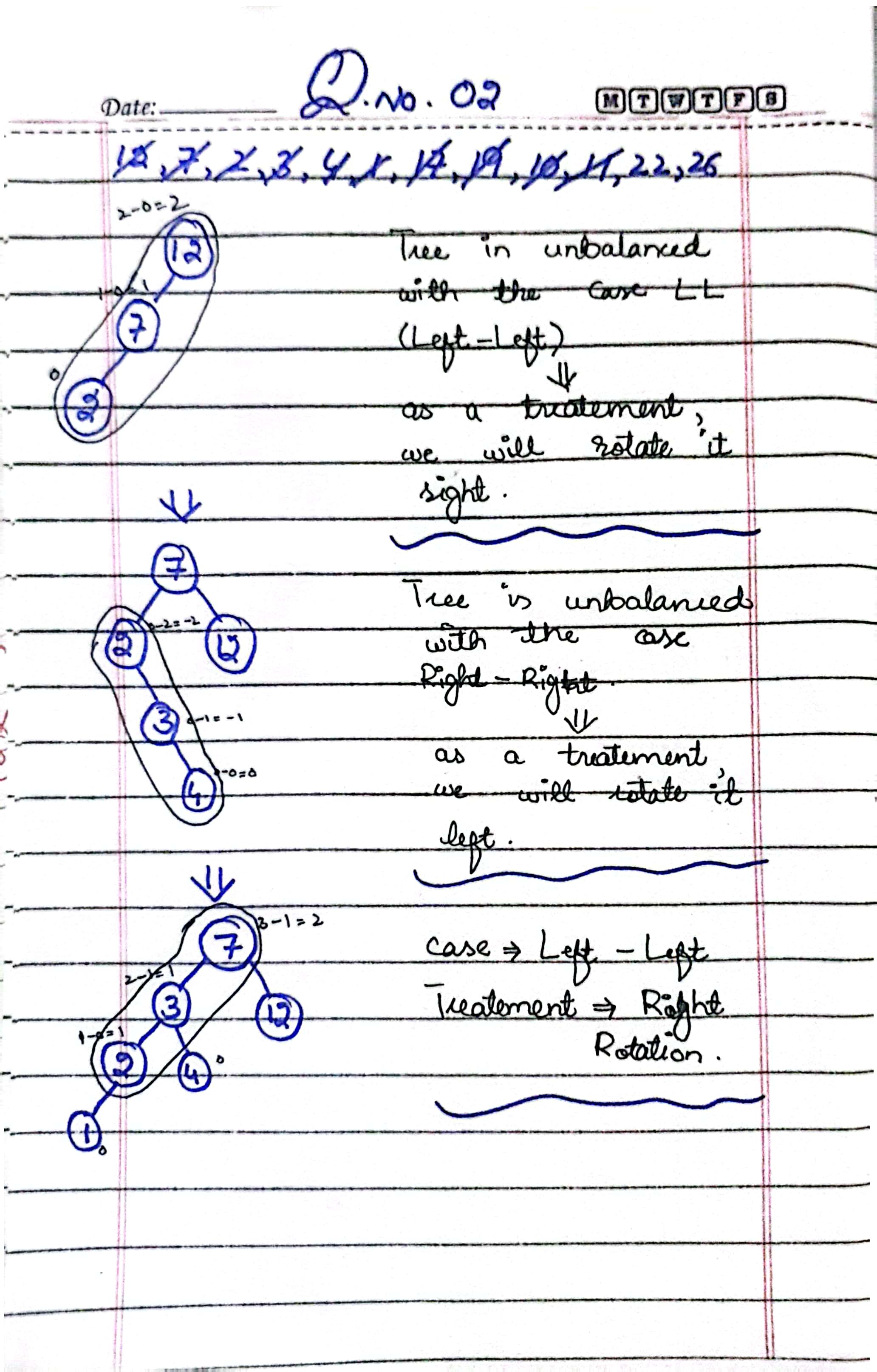
**Assignment 04**

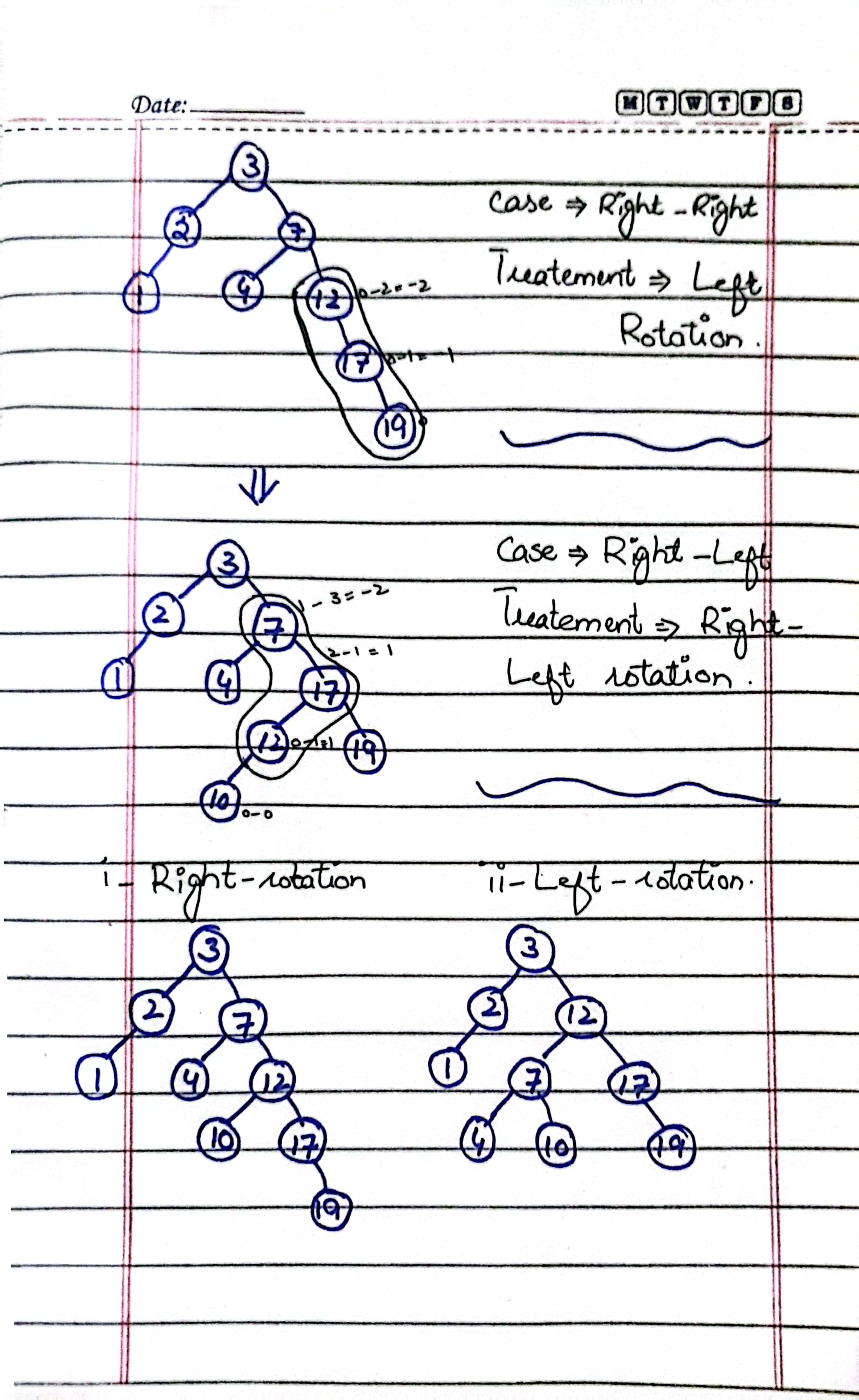
**Task 1**

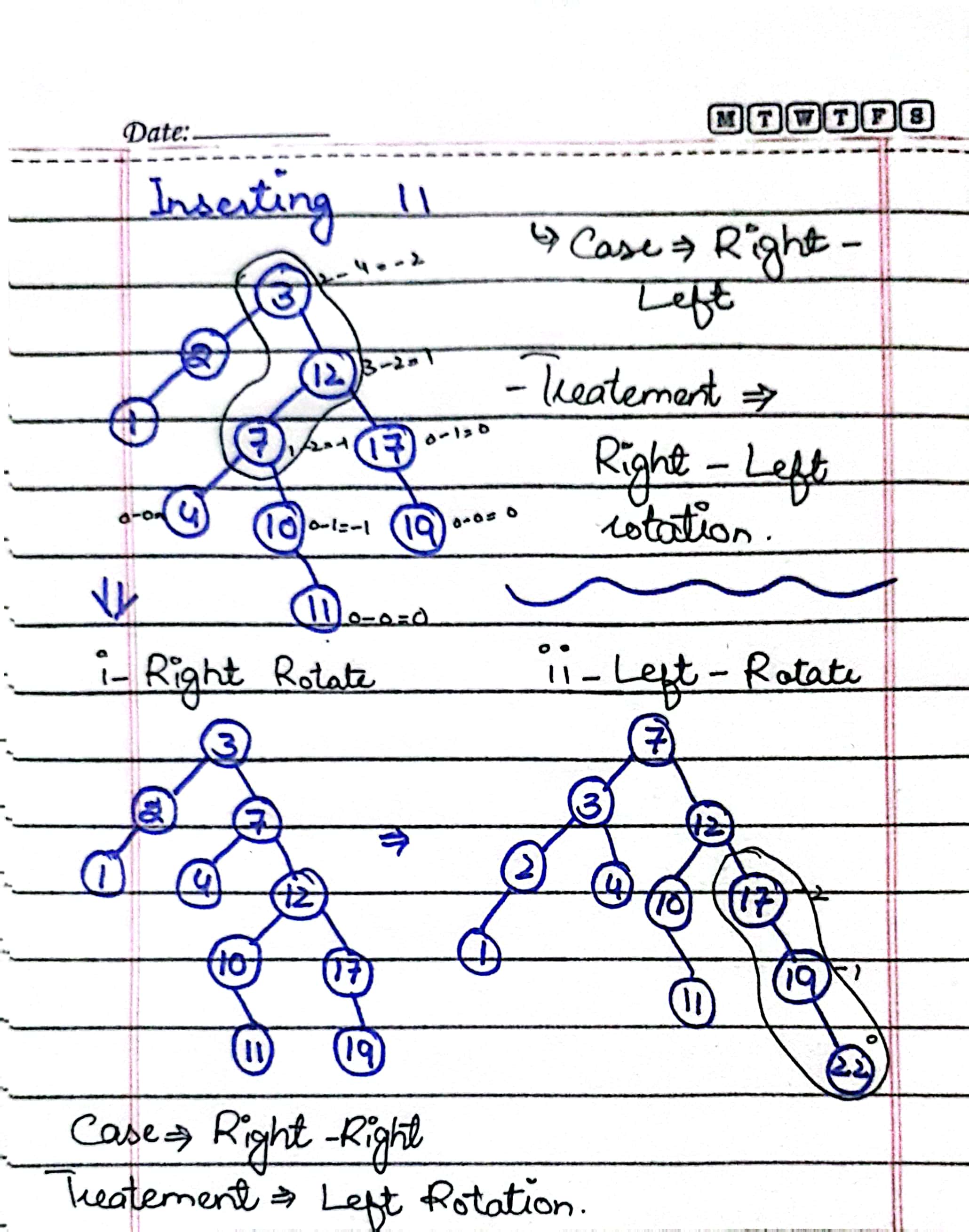
****

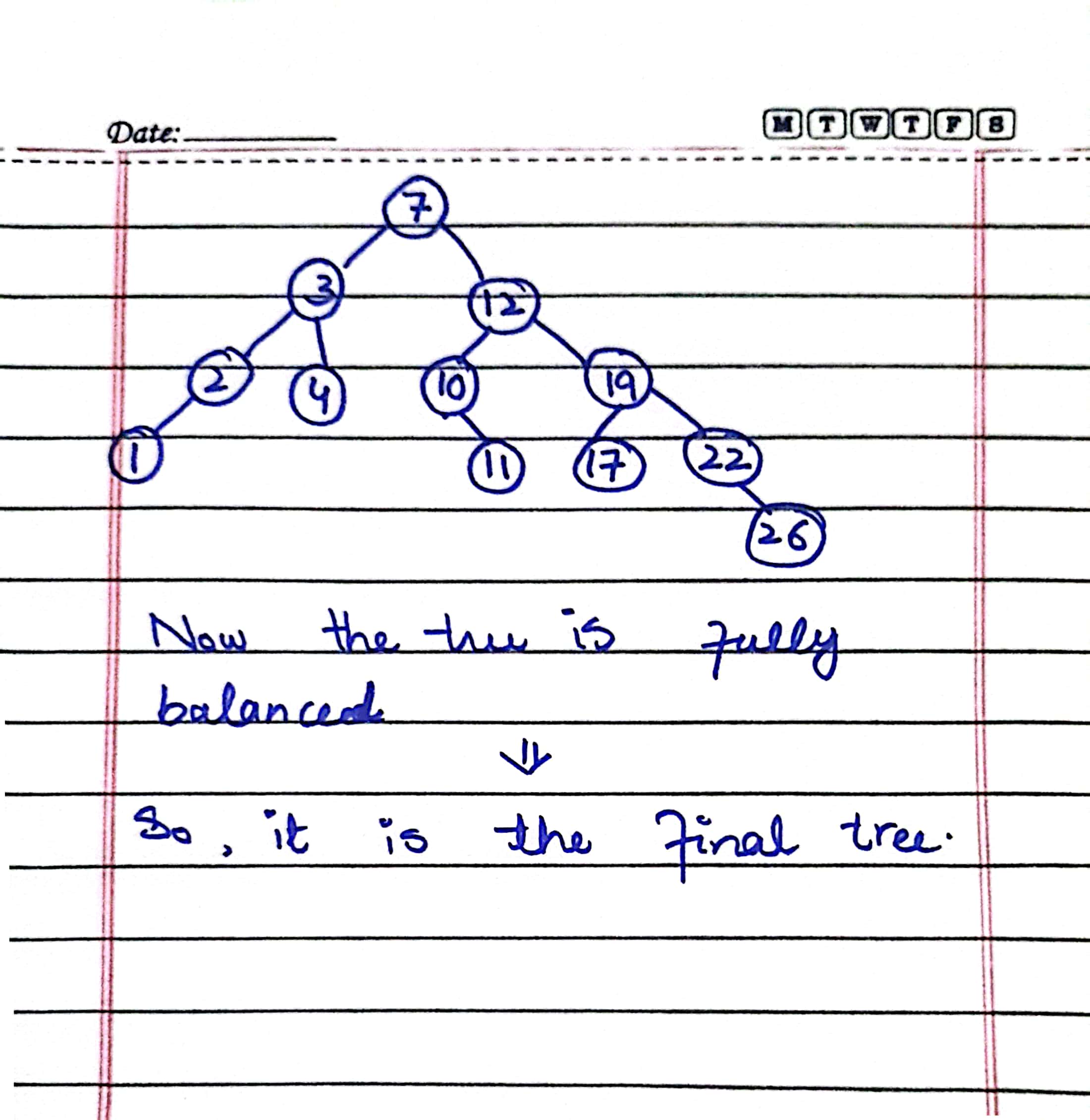
****

**Task 02**

****

****

****

****

**Task 03**

//Taaha Hussain Khan

//L1F21BSCS0917

//D12

//Assignment 04 Task 03

#include <iostream>

using namespace std;

void maxHeapify(int arr[], int size, int i);

void buildHeap(int arr[], int size);

void displayHeap(int arr[], int size);

int main()

{

    int heap\_A[] = {22, 20, 21, 11, 12}, heap\_B[] = {45, 33, 38, 8, 9}, size1 = 5, size2 = 5, resultant;

    cout << "Heap A:\n";

    buildHeap(heap\_A, size1);

    displayHeap(heap\_A, size1);

    cout << "Heap B:\n";

    buildHeap(heap\_B, size2);

    displayHeap(heap\_B, size2);

    resultant = size1 + size2;

    int \*heap\_C = new int[resultant];

    for (int i = 0; i < size1; i++)

        heap\_C[i] = heap\_A[i];

    int j = 0;

    for (int i = size2; i < resultant; i++, j++)

        heap\_C[i] = heap\_B[j];

    cout << "Output: Resulting Merged Heap\n\n";

    cout<<"Heap C:\n"<<endl;

    buildHeap(heap\_C, resultant);

    displayHeap(heap\_C, resultant);

    return 0;

}

void maxHeapify(int arr[], int size, int i)

{

    int largest = i;

    int child1 = 2 \* i + 1;

    int child2 = 2 \* i + 2;

    if (child1 < size && arr[child1] > arr[largest])

        largest = child1;

    if (child2 < size && arr[child2] > arr[largest])

        largest = child2;

    if (largest != i)

    {

        int temp = arr[i];

        arr[i] = arr[largest];

        arr[largest] = temp;

        maxHeapify(arr, size, largest);

    }

    return;

}

void buildHeap(int arr[], int size)

{

    for (int i = size / 2 - 1; i >= 0; i--)

        maxHeapify(arr, size, i);

    return;

}

void displayHeap(int arr[], int size)

{

    for (int i = 0; i < size; i++)

        cout << arr[i] << " ";

    cout << endl;

    return;

}

**Output:**

