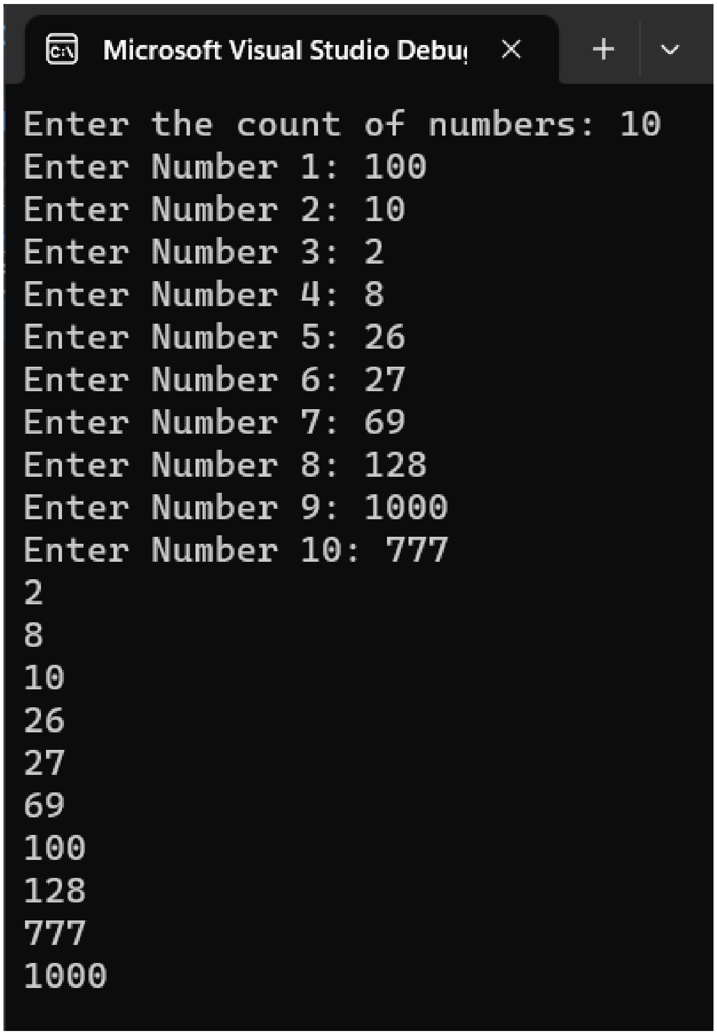
Task 4

Code

```
// BST Implementation
#include <iostream>

struct Node
{
    int data;
    Node* smaller;
    Node* larger;

    Node(int d, Node* s, Node* l)
    {
        this->data = d;
        this->smaller = s;
        this->larger = l;
    }
};

void Push(Node* node, int number)
{
    if (node->data > number)
        if (node->smaller)
            Push(node->smaller, number);

        else
            node->smaller = new Node(number, nullptr, nullptr);

    else if (node->data < number)
        if (node->larger)
            Push(node->larger, number);

        else
            node->larger = new Node(number, nullptr, nullptr);

    else
        std::cout << "Number already exists in the tree;" << std::endl;
}

void PrintTreeData(Node* rootNode)
{
    if (rootNode->smaller)
        PrintTreeData(rootNode->smaller);

    std::cout << rootNode->data << std::endl;

    if (rootNode->larger)
        PrintTreeData(rootNode->larger);
};

int main()
{
    Node rootNode{0, nullptr, nullptr};
    int totalNumbers;

    std::cout << "Enter the count of numbers: ";
    std::cin >> totalNumbers;

    std::cout << "Enter Number 1: ";
    std::cin >> rootNode.data;
    rootNode.smaller = nullptr;
    rootNode.larger = nullptr;

    for (int i = 1; i < totalNumbers; i++)
    {
        int number;
        std::cout << "Enter Number " << i + 1 << ": ";
        std::cin >> number;

        Push(&rootNode, number);
    }

    PrintTreeData(&rootNode);
}
```

Solution



```
Enter the count of numbers: 10
Enter Number 1: 100
Enter Number 2: 10
Enter Number 3: 2
Enter Number 4: 8
Enter Number 5: 26
Enter Number 6: 27
Enter Number 7: 69
Enter Number 8: 128
Enter Number 9: 1000
Enter Number 10: 777
2
8
10
26
27
69
100
128
777
1000
```