```
#include <iostream>
using namespace std;

// Data structure to store a binary tree node
struct Node
{
	int data;
	Node *left, *right;

	Node(int data)
	{
		this->data = data;
		this->left = this->right = nullptr;
	}
};

// Recursive function to perform inorder traversal on the tree
void inorder(Node* root)
{
	// return if the current node is empty
	if (root == nullptr) {
		return;
	}

	// Traverse the left subtree
	inorder(root->left);

	// Display the data part of the root (or current node)
	cout << root->data << " ";

	// Traverse the right subtree
```

```
        inorder(root->right);
}

int main()
{
        /* Construct the following tree
                    1
                  /   \
                 /     \
                2       3
               /      /   \
              /      /     \
             4      5       6
                           /  \
                          /    \
                         7      8
        */

        Node* root = new Node(1);
        root->left = new Node(2);
        root->right = new Node(3);
        root->left->left = new Node(4);
        root->right->left = new Node(5);
        root->right->right = new Node(6);
        root->right->left->left = new Node(7);
        root->right->left->right = new Node(8);

        inorder(root);

        return 0;
}
```

OUTPUT:

4 2 1 7 5 8 3 6