LINKEDLIST | C++

CODE

```
// HEADER FILES
#include <iostream>
#include <conio.h>
using namespace std;
// CLASS FOR LINKEDLIST
class List {
private:
	// STRUCTURE
	typedef struct node {
		int data;
		node *next;
	} *nodePtr;

	nodePtr head;
	nodePtr curr;
	nodePtr temp;

public:
	// CONSTRUCTOR
	List() {
		head = NULL;
		curr = NULL;
		temp = NULL;
	}
	// FUNCTION FOR ADDING NODE
	void AddNode(int addData) {
		nodePtr n = new node;
		n->data = addData;
		n->next = NULL;

		if (head != NULL) {
			curr = head;
			while (curr->next != NULL) {
				curr = curr->next;
			}
			curr->next = n;
		}
		else {
			head = n;
		}
	}
	// FUNCTION FOR DELETING NODE
	void DeleteNode(int delData) {
		nodePtr delPtr = NULL;
		temp = head;
		curr = head;
		while (curr != NULL && curr->data!=delData) {
```

CODE

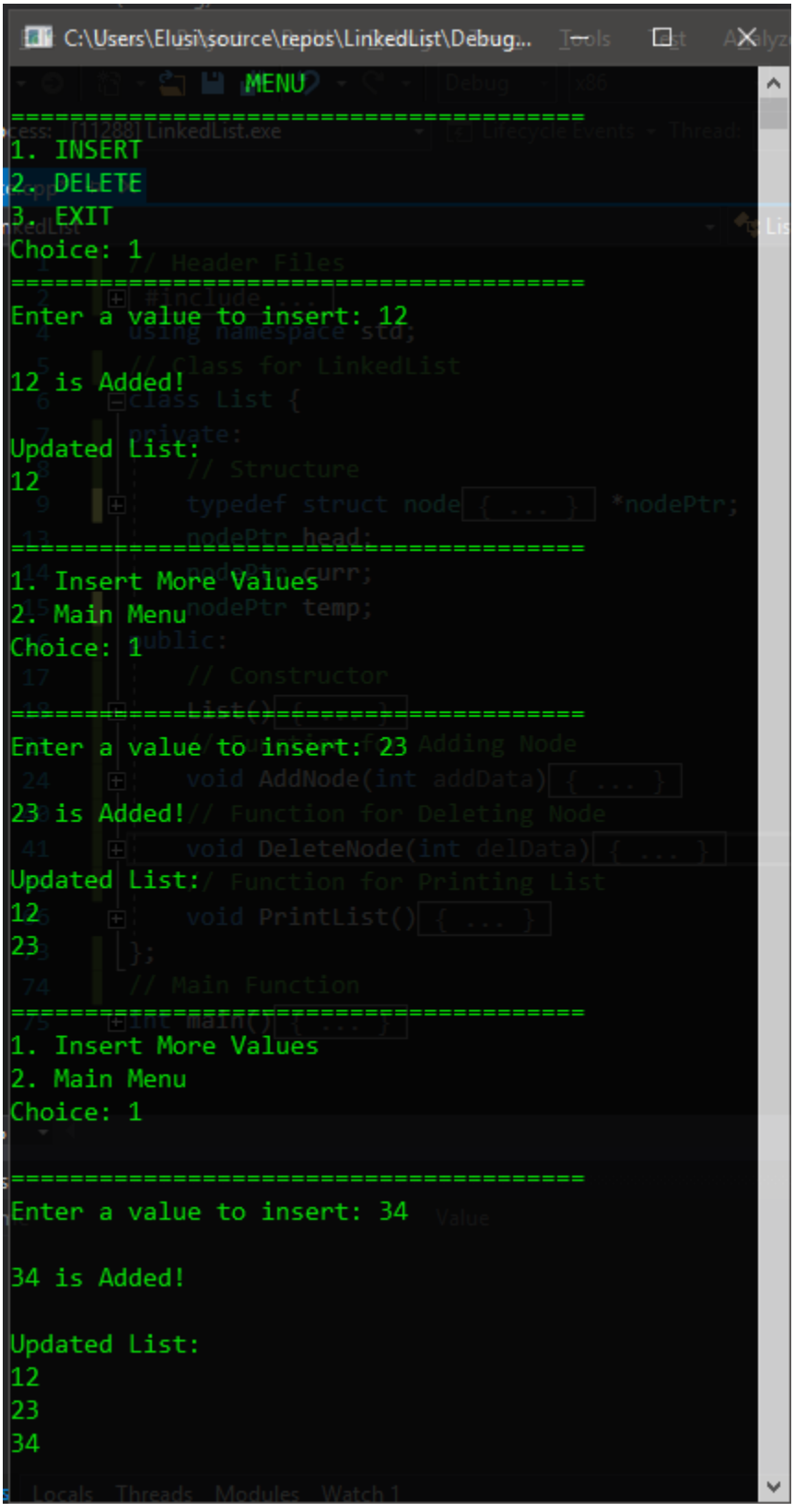
```
                        temp = curr;
                        curr = curr->next;
                }
                if (curr == NULL) {
                        cout << delData << " is not in the List.\n";
                        delete delPtr;
                }
                else {
                        delPtr = curr;
                        curr = curr->next;
                        temp->next = curr;
                        if (delPtr == head) {
                                head = head->next;
                                temp = NULL;
                        }
                        delete delPtr;
                        cout << " The value " << delData << " is deleted.\n";
                }
        }
        // FUNCTION FOR PRINTING LIST
        void PrintList() {
                curr = head;
                while (curr != NULL) {
                        cout << curr->data<<endl;
                        curr = curr->next;
                }
        }
};
// MAIN FUNCTION
int main() {
        List Usama;
mmenu:
        cout << "\n=======================================\n";
        cout << "\t\tMENU";
        cout << "\n=======================================\n";
        cout<<"1. INSERT\n2. DELETE\n3. EXIT\n";
        char choice;
        cout << "Choice: ";
        cin >> choice;
        if (choice == '1') {
                cout << "=======================================\n";
                int num1;
        start:
                cout << "Enter a value to insert: ";
                cin >> num1;
                Usama.AddNode(num1);
                cout << "\n"<<num1<<" is Added!\n\nUpdated List: \n";
                Usama.PrintList();
                cout << "\n=======================================\n";
                cout << "1. Insert More Values\n2. Main Menu\n";
                        cout<<"Choice: ";
                cin >> choice;
```

CODE

```
		cout << "\n=======================================\n";
		if (choice == '1')
			goto start;
		else if (choice == '2')
			goto mmenu;
	}
	else if (choice=='2') {
		cout << "\nList Values\n";
		Usama.PrintList();
		int num1;
		cout << "\nEnter a value to delete: ";
		cin >> num1;
		Usama.DeleteNode(num1);
		cout << "\nUpdated List: \n";
		Usama.PrintList();
		goto mmenu;
	}
	else if (choice == '3') {
		cout << "\n=======================================\n";
		cout << "   Thanks for Using our Software :)";
		cout << "\n=======================================\n";
	}
	else {
		cout << "\nInvalid!\n";
		goto mmenu;
	}
	_getch();
}
```

LINKEDLIST | C++

OUTPUT



LINKEDLIST | C++

OUTPUT