```
#include <iostream>
 using namespace std;
class Node {
 public:
 int data;
 Node* prev;
 Node* next;
 Node(int val) {
 data = val;
 prev = NULL;
 next = NULL;
 }
 };
  class Binary {
 public:
 Node* head;
 Node* last;
 Binary() {
 head = NULL;
 last = NULL;
 }
 void CreateList(int n) {
 while (n != 0) {
 int rem = n % 2;
 Node* temp = new Node(rem);
 if (head == NULL) {
 head = last = temp;
 } else {
 last->next = temp;
 temp->prev = last;
 last = temp;
 }
 n =n/2;
 }
 }
void DisplayList() {
 Node* temp = last;
 while (temp != NULL) {
 cout << temp->data;
 temp = temp->prev;
 }
 cout << endl;
 }
  void Ones_complement() {
 Node* temp = head;
 while (temp != NULL) {
 temp->data = 1- temp->data;
 temp = temp->next;
 }
 }
void Twos_complement() {
 Ones_complement();
 Node* temp = last;
 int carry = 1;
 while (temp != NULL && carry != 0) {
 int sum = temp->data + carry;
 temp->data = sum % 2;
 carry = sum / 2;
 temp = temp->prev;
 }
 }
 };
  int main() {
 Binary b1, b2, b3;
 int choice, num1, num2;
 while (1) {
 cout << "\nMenu:\n";
 cout << "1. Create Binary List\n";
 cout << "2. Display Binary List\n";
 cout << "3. 1's Complement\n";
 cout << "4. 2's Complement\n";
 cout << "5. Add Two Binary Numbers\n";
 cout << "6. Exit\n";
cout << "Enter your choice: ";
 cin >> choice;
 switch (choice) {
 case 1:
 cout << "Enter Decimal Number: ";
 cin >> num1;
 b1.CreateList(num1);
 break;
 case 2:
 cout << "Binary Representation: ";
 b1.DisplayList();
 break;
 case 3:
 b1.Ones_complement();
 cout << "1's Complement: ";
 b1.DisplayList();
 break;
 case 4:
 b1.Twos_complement();
 cout << "2's Complement: ";
 b1.DisplayList();
 break;
 case 5:
 cout << "Enter first decimal number: ";
 cin >> num1;
 cout << "Enter second decimal number: ";
 cin >> num2;
 int n3;
 n3 = num1 +num2;
 cout << "Addition of Binary Numbers: ";

 b3.CreateList(n3);

 b3.DisplayList();

 break;

 case 6:

 return 0;

 default:

 cout << "Invalid choice!";

 }

 }

 return 0;

 }
```