Data structure project

```
#include <iostream>
#include <windows.h>
#include <stack>
#include <unordered_map>
using namespace std;

void gotoxy(int x, int y)
{
   static HANDLE h = NULL;
   if (!h)
      h = GetStdHandle(STD_OUTPUT_HANDLE);
   COORD c = {x, y};
   SetConsoleCursorPosition(h, c);
}

struct Candidate {
   string name;
   int votesCount;
   Candidate(string n) : name(n), votesCount(0) {}
};

class ElectionSystem {
private:
   stack<int> votesStack;
   unordered_map<string, Candidate*> candidatesMap;
public:
   ElectionSystem() {
      candidatesMap["Anay"] = new Candidate("Anay");
      candidatesMap["Enan"] = new Candidate("Enan");
      candidatesMap["Akram"] = new Candidate("Akram");
      candidatesMap["Shahriyar"] = new Candidate("Shahriyar");
   }

   void castVote() {
      int choice;
      cout << "\n\n ### Please choose your Candidate ####\n\n";
      int index = 1;
      for (auto it = candidatesMap.begin(); it != candidatesMap.end(); ++it) {
         cout << " " << index++ << ". " << it->first << endl;
      }
      cout << " " << index << ". None of These" << endl;
      cout << "\n Input your choice (1 - " << index << ") : ";
      cin >> choice;

      if (choice >= 1 && choice <= index) {
         votesStack.push(choice);
         cout << "\n Thanks for voting!";
```

```
        } else {
            cout << "\n Error: Wrong Choice! Please retry";
        }
    }

    void votesCount() {
        cout << "\n\n --> Voting Statistics <--";
        for (auto it = candidatesMap.begin(); it != candidatesMap.end(); ++it) {
            cout << "\n " << it->second->name << " - " << it->second->votesCount;
        }
        cout << "\n Spoiled Votes - " << votesStack.size();
    }

    void getLeadingCandidate() {
        cout << "\n\n --> Leading Candidate <--\n\n";
        Candidate* leadingCandidate = nullptr;
        for (auto it = candidatesMap.begin(); it != candidatesMap.end(); ++it) {
            if (leadingCandidate == nullptr || it->second->votesCount >
leadingCandidate->votesCount) {
                leadingCandidate = it->second;
            }
        }
        if (leadingCandidate != nullptr) {
            cout << "[" << leadingCandidate->name << "]";
        } else {
            cout << "-----> Warning !!! No-win situation <----";
        }
    }

    void processVotes() {
        while (!votesStack.empty()) {
            int choice = votesStack.top();
            votesStack.pop();
            if (choice >= 1 && choice <= candidatesMap.size()) {
                auto it = candidatesMap.begin();
                advance(it, choice - 1);
                it->second->votesCount++;
            }
        }
    }

    ~ElectionSystem() {
        for (auto it = candidatesMap.begin(); it != candidatesMap.end(); ++it) {
            delete it->second;
        }
    }
};

int main() {

    int i, t = 120;
```

```
cout << "==========> [DATA STRUCTURE PROJECT] <========\n";
cout << "Submitted To:-Safiullah Sir\n ";

cout <<"Submitted By:-\nShakawat Hossain\n ";

for (i = 1; i <= 11; i++)
{
   gotoxy(7, 3 + i);
   cout << "***";
   Sleep(t);
}
for (i = 1; i <= 6; i++)
{
   gotoxy(6, 13);
   cout << "*";
   Sleep(t);
}
for (i = 1; i <= 6; i++)
{
   gotoxy(5, 14);
   cout << "";
   Sleep(t);
}

for (i = 1; i <= 6; i++)
{
   gotoxy(7 + i, 3 + i);
   cout << "***";
   Sleep(t);
}
for (i = 1; i <= 6; i++)
{
   gotoxy(13 + i, 10 - i);
   cout << "***";
   Sleep(t);
}
for (i = 1; i <= 11; i++)
{
   gotoxy(19, 3 + i);
   cout << "***";
   Sleep(t);
}
for (i = 1; i <= 6; i++)
{
   gotoxy(22, 13);
   cout << "*";
   Sleep(t);
}
for (i = 1; i <= 6; i++)
{
   gotoxy(22, 14);
```

```
        cout << "";
        Sleep(t);
    }
for (i = 1; i <= 5; i++)
    {
        gotoxy(30 + i, 4);
        cout << "";
        Sleep(t);
    }
    for (i = 1; i <= 9; i++)
    {
        gotoxy(27 + i, 5);
        cout << "***";
        Sleep(t);
    }
    for (i = 1; i <= 7; i++)
    {
        gotoxy(27, 5 + i);
        cout << "***";
        Sleep(t);
    }
    for (i = 1; i <= 9; i++)
    {
        gotoxy(27 + i, 13);
        cout << "***";
        Sleep(t);
    }
    for (i = 1; i <= 5; i++)
    {
        gotoxy(30 + i, 14);
        cout << "";
        Sleep(t);
    }
    for (i = 1; i <= 7; i++)
    {
        gotoxy(37, 13 - i);
        cout << "***";
        Sleep(t);
    }
    //B
    for (i = 1; i <= 11; i++)
    {
        gotoxy(45, 3 + i);
        cout << "***";
        Sleep(t);
    }
    for (i = 1; i <= 2; i++)
    {
        gotoxy(44, 13);
        cout << "****";
        Sleep(t);
```

```
}
for (i = 1; i <= 2; i++)
{
   gotoxy(43, 14);
   cout << "*";
   Sleep(t);
}
for (i = 1; i <= 6; i++)
{
   gotoxy(45 + i, 4);
   cout << "*";
   Sleep(t);
}

for (i = 1; i <= 6; i++)
{
   gotoxy(45 + i, 5);
   cout << "";
   Sleep(t);
}
for (i = 1; i <= 2; i++)
{
   gotoxy(55, 5 + i);
   cout << "***";
   Sleep(t);
}

for (i = 1; i <= 6; i++)
{
   gotoxy(45 + i, 8);
   cout << "*";
   Sleep(t);
}

for (i = 1; i <= 6; i++)
{
   gotoxy(46 + i, 9);
   cout << "";
   Sleep(t);
}
for (i = 1; i <= 3; i++)
{
   gotoxy(56, 9 + i);
   cout << "***";
   Sleep(t);
}
for (i = 1; i <= 6; i++)
{
   gotoxy(46 + i, 13);
   cout << "";
   Sleep(t);
```

```
    }
    for (i = 1; i <= 6; i++)
    {
        gotoxy(46 + i, 14);
        cout << "*";
        Sleep(t);
    }

    for (i = 1; i <= 11; i++)
    {
        gotoxy(60 + i, 4);
        cout << "***";
        Sleep(t);
    }
    for (i = 1; i <= 11; i++)
    {
        gotoxy(60 + i, 5);
        cout << "***";
        Sleep(t);
    }
    for (i = 1; i <= 11; i++)
    {
        gotoxy(66, 3 + i);
        cout << "***";
        Sleep(t);
    }
    for (i = 1; i <= 11; i++)
    {
        gotoxy(60 + i, 13);
        cout << "***";
        Sleep(t);
    }
    for (i = 1; i <= 11; i++)
    {
        gotoxy(60 + i, 14);
        cout << "***";
        Sleep(t);
    }
///N
    for (i = 1; i <= 11; i++)
    {
        gotoxy(78, 3 + i);
        cout << "***";
        Sleep(t);
    }
    for (i = 1; i <= 2; i++)
    {
        gotoxy(77, 13);
        cout << "*";
        Sleep(t);
    }
```

```
for (i = 1; i <= 2; i++)
{
   gotoxy(76, 14);
   cout << "";
   Sleep(t);
}
for (i = 1; i <= 11; i++)
{
   gotoxy(78 + i, 3 + i);
   cout << "***";
   Sleep(t);
}
for (i = 1; i <= 9; i++)
{
   gotoxy(90, 15 - i);
   cout << "***";
   Sleep(t);
}
for (i = 1; i <= 2; i++)
{
   gotoxy(90, 4);
   cout << "*";
   Sleep(t);
}
for (i = 1; i <= 2; i++)
{
   gotoxy(90, 5);
   cout << "*****";
   Sleep(t);
}

gotoxy(90, 15);
ElectionSystem electionSystem;
int choice;

do {
   cout << "\n\n --> Welcome to IIUC ACR Election 2022 (2EM) <--";
   cout << "\n\n 1. Cast the vote";
   cout << "\n 2. Find vote count";
   cout << "\n 3. Find Leading Candidate";
   cout << "\n 0. Exit";
   cout << "\n\n Please enter your choice : ";
   cin >> choice;

   switch (choice) {
      case 1:
         electionSystem.castVote();
         break;
      case 2:
         electionSystem.processVotes();
         electionSystem.votesCount();
```

```
            break;
        case 3:
            electionSystem.processVotes();
            electionSystem.getLeadingCandidate();
            break;
        default:
            cout << "\n Error: Invalid Choice";
      }
   } while (choice != 0);

   return 0;
}
```