**C++**

PROJECT REPORT

ON

“ELECTION 2020”

**PARTICULARS OF THE STUDENT :**

NAME : MADHUR NARANG

ROLL NO. :

CLASS : XII -A

SCHOOL : DELHI INTERNATIONAL PUBLIC SCHOOL

CERTIFICATE

This is to certify that the project entitled **“ELECTION 2020”** which deals with various processes involved in elections which include addition and alteration of details of candidates, issuing of voter ID Cards and fixing the election dates.

This project is a bonafied piece of work carried out by me with the consultance and able guidance of our supervisor.

ROLL NO. :

Ms. MANJUSHA

(T.G.T COMPUTER SCIENCE)

**ACKNOWLEDGEMENT**

I would like to express a deep sense of thanks and gratitude to various people who helped me a long way in completing this project.

It was privilege to have been guided by **Ms. Manjusha**. She always evinced keen interest in my work .Her constructive advice and constant motivation has been the reason for the successful completion of this project .I would also like to thank classmates for their contribution towards the making of this project .

Thanks to all for development of this project with their constructive criticism and advice.

ROLL NO. :

**SYSTEM DESCRIPTIQN AT A GLANCE**

This project is based on Election Handling Systems. This project help us enter, display and alter the details of different candidates. It also allows us to issue Voter ID Cards and fix a date for Election. Data file handling has been effectively used in the making of this program. It also include various functions which are programmed to perform the above mentioned tasks, Data file handling has been effectively and efficiently used in the program.

This program provides the following services :

* **Issue Voter ID cards**

A person can get his/her Voter ID card issued by simply entering his voter ID Card number. The status of the Voter ID Card is “ ISSUED ” or “NOT ISSUED” .

* **Setting Election Date**

Any official can set new Election Date i.e. the date on which the election will be conducted.

* **Addition of voters**

A person can add new voters to the election system database by entering the name and addresses of the voter.

* **Addition of candidates**

A new candidate can be added to the system database by simply entering the name of the candidate and the symbol of his party.

* **Calculations of votes**

The calculation of votes can also be done with the help of this system .

* **Display of voter details**

A person can also view the details of all the voters with their address and the status of their Voter ID card.

* **Display details of candidates**

A person can view the details of all the candidates who are contesting elections. Their name along with will also be displayed.

* **Deletion of voters**

A person can also delete the details of a voter from the voters list

* **Deletion of candidates**

A person can also delete the details of a candidate from the candidate list.

**DATABASES**

* ***VOTER.DAT***

VOTER No Serial number of voters

NAME Name of the voter

ADDRESS Address of the voter

CARDSTATUS Status of the voter ID card

S.No. Serial number

* ***ELEDATE.DAT***

VOTING DATE Date of election

DA\_DAY Day of election

DA\_MON Month of election

DA\_YEAR Year of election

* ***CAN.DAT***

SERIALNO. Serial number of candidates

NAME Name of the candidate

SYMBOL Symbol of candidate

VOTES Count number of votes

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// PROJECT ELECTION 2020

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// INCLUDED HEADER FILES

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include <fstream.h>

#include <string.h>

#include <stdlib.h>

#include <stdio.h>

#include <ctype.h>

#include <conio.h>

#include <dos.h>

void upper(char ar[])

{ int l=strlen(ar);

for(int i=0;i<l;i++)

{

ar[i]=toupper(ar[i]);

}

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// THIS CLASS CONTROL A|| THE FUNCTIONS RELATED TO VOTER

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class VOTER

{

public :

void ADD\_VOTER(void) ;

void VOTER\_INFORMATION(void) ;

void VOTER\_LIST(void) ;

void ISSUE\_VOTING\_CARD(void) ;

void RESET\_STATUS(void) ;

void DELETE\_VOTER(void) ;

void ELECTION(void);

void VOTE\_CALCULATION(void) ;

void CALCULATE(char []) ;

protected :

int DELETE\_RECORD(char []) ;

char voterno[6], name[26], address[36] ;

int status, cardstatus;

};

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// THIS CLASS CONTROL A|| THE FUNCTIONS RELATED TO CANDIDATE

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class CANDIDATE

{

public:

void ADD\_CANDIDATE(void) ;

void CANDIDATE\_INFORMATION(void) ;

void VOTING\_LIST(void) ;

int ADD\_VOTES(int);

void RESET\_STATUS(void) ;

void DELETE\_CANDIDATE(void) ;

protected :

int DELETE\_RECORD(int) ;

char name[26], symbol[15] ;

int votes , serialno;

};

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// THIS CLASS CONTROL A|| THE FUNCTIONS RELATED ELECTION DATE

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class VOTING\_DATE

{

public:

void SET\_DATE(void) ;

int VALID\_DATE(void);

void RETURN\_DATE(void) ;

private :

int dd, mm, yy;

};

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// THIS CLASS CONTROL A|| THE FUNCTIONS T0 DRAW LINES,

// Boxes, ETC.

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class DRAW

{

public :

void LINE\_HOR(int, int, int, char) ;

void LINE\_VER(int, int, int, char) ;

};

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO DRAW HORIZONTAL LINE

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void DRAW :: LINE\_HOR(int column1, int column2, int row, char c)

{

for ( column1; column1<=column2; column1++)

{

gotoxy(column1,row) ;

cout <<c ;

}

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO DRAW VERTICAL LINE

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void DRAW :: LINE\_VER(int row1, int row2, int column, char c)

{

for ( row1; row1<=row2; row1++)

{

gotoxy(column,row1) ;

cout <<c ;

}

}

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO ADD THE VOTERS IN VOTERS FILE

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void VOTER :: ADD\_VOTER(void)

{

char ch ;

do

{

clrscr() ;

gotoxy(27,1) ;

cout<<"\*\*\* ADDITION OF VOTERS \*\*\*" ;

int found=0 ;

char vno[6] ;

do

{

gotoxy(10,8) ;

clreol() ;

cout <<"Enter voter no. : " ;

gets(vno) ;

if (vno[0]=='0')

return ;

found=0 ;

fstream f ;

f.open("VOTER.DAT", ios::in) ;

if (f.fail())

break ;

while (f.read((char \*) this, sizeof(VOTER)))

if (!strcmpi(voterno,vno))

{

found = 1 ;

break ;

}

f.close() ;

if (found)

{

gotoxy(10,8) ;

clreol() ;

cout <<"Voter no already exist“ ;

getch() ;

}

}while (found) ;

gotoxy(10,10) ;

cout <<"Enter name of the voter : ";

gets(name) ;

upper(name);

gotoxy(10,11) ;

cout <<"Enter address of the voter : " ;

gets(address) ;

upper(address);

status = 0 ;

cardstatus = 0 ;

gotoxy(8,13) ;

cout <<"Save voter record? (y/n) : " ;

ch = getche() ;

if ((ch == 'y' )||( ch == 'Y'))

{

strcpy(voterno,vno) ;

fstream f ;

f.open("VOTER.DAT", ios::app) ;

f.write((char\*)this, sizeof(VOTER)) ;

f.close() ;

}

gotoxy(8,15) ;

cout <<"Add more? (y/n) : " ;

ch = getche() ;

}while((ch == 'y')||(ch == 'Y'));

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION T0 DISPLAY THE VOTER'S LIST

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void VOTER :: VOTER\_LIST(void)

{

clrscr() ;

int row=5, found=0, flag=0 ;

char ch ;

gotoxy(29,1) ;

cout<<"\*\*\* LIST OF VOTERS \*\*\*" ;

gotoxy(1,3) ;

cout <<“VOTE NO. VOTER NAME VOTER ADDRESS CARD ISSUED" ;

gotoxy(1,4) ;

cout <<”~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ “;

fstream f ;

f.open("VOTER.DAT", ios::in) ;

while (f.read((char \*) this, sizeof(VOTER)))

{

flag = 0 ;

found = 1 ;

gotoxy(1,row) ;

cout <<voterno ;

gotoxy(11,row) ;

cout <<name ;

gotoxy(29,row);

cout<<address ;

gotoxy(65,row) ;

if (cardstatus)

cout <<"Yes" ;

else

cout <<"No" ;

if( row == 23 )

{

flag = 1 ;

row=5;

gotoxy(1,25) ;

cout <<"Press any key to continue or Press <ESC> to exit" ;

ch = getch() ;

if (ch == 27)

break ;

clrscr() ;

gotoxy(29,1) ;

cout<<"\*\*\* LIST OF VOTERS \*\*\*" ;

gotoxy(1,3) ;

cout <<"VOTE NO. VOTER NAME VOTER ADDRESS CARD ISSUED" ;

gotoxy(1,4) ;

cout<<” ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ ~ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~”;

}

else

row++ ;

}

if (!found)

{

gotoxy(5,10) ;

cout <<"Records not found" ;

}

if (!flag)

{

gotoxy(1,25) ;

cout <<"Press any key to continue..." ;

getch() ;

}

f.close() ;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO DISPLAY THE VOTER INFORMATION

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void VOTER :: VOTER\_INFORMATION(void)

{

clrscr() ;

gotoxy(27,1) ;

cout<<"\*\*\* VOTER INFORMATION \*\*\*" ;

char vno[6] ;

gotoxy(5,5) ;

cout <<"Enter Voter no. or Press <ENTER> to see list : " ;

gets(vno) ;

if (vno[0] == '0')

return ;

if (strlen(vno) == 0)

{

VOTER\_LIST() ;

return ;

}

int found=0 ;

fstream f ;

f.open("VOTER.DAT", ios::in);

while (f.read((char \*) this, sizeof(VOTER)))

{

if (!strcmpi(vno,voterno))

{

found = 1 ;

gotoxy(10,8) ;

cout <<"Voter no. : " <<voterno ;

gotoxy( 10,10) ;

cout <<"Voter name : " <<name ;

gotoxy(10,11) ;

cout <<"Voter address : " <<address ;

gotoxy(10,13) ;

if (cardstatus)

cout <<"Voting card issued :Yes" ;

else

cout <<"Voting card issued : No" ;

gotoxy(10,14) ;

if (status)

cout <<"Vote given : Yes" ;

else

cout <<"Vote given : No" ;

break ;

}

}

f.close() ;

if (!found)

{

gotoxy(5,20) ;

cout <<"Voting no. not found" ;

}

gotoxy(5,24) ;

cout <<"Press any key to continue..." ;

getch() ;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO RESET A|| THE STATUS IN THE VOTER'S FILE

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void VOTER :: RESET\_STATUS(void)

{

fstream file ;

file.open("VOTER.DAT", ios::in) ;

if (file.fail())

return ;

fstream temp ;

temp.open("temp.dat", ios::out) ;

while (file.read((char \*) this, sizeof(VOTER)))

{

status = 0 ;

temp.write((char \*) this, sizeof(VOTER)) ;

}

file.close() ;

temp.close() ;

file.open("VOTER.DAT", ios::out) ;

temp.open("temp.dat", ios::in) ;

while (temp.read((char \*) this, sizeof(VOTER)))

file.write((char \*) this, sizeof(VOTER)) ;

file.close() ;

temp.close() ;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO ISSUE THE VOTING CARDS TO THE VOTERS

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void VOTER :: ISSUE\_VOTING\_CARD(void)

{

clrscr() ;

gotoxy(27,1) ;

cout<<"\*\*\* ISSUE VOTING CARD \*\*\*";

char vno[6] ;

gotoxy(5,5) ;

cout <<"Enter Vote no. or Press <ENTER> to exit :" ;

gets(vno) ;

if (strlen(vno) == 0)

return ;

int found=0, count=0;

fstream f ;

f.open("VOTER.DAT", ios::in) ;

while (f.read((char \*) this, sizeof(VOTER)))

{

count++ ;

if (!strcmpi(vno,voterno))

{

found = 1 ;

if (cardstatus == 1)

{

gotoxy(5,10) ;

cout <<"Voting card a||ready issued" ;

getch();

return ;

}

break;

}

}

f.close() ;

if (found)

{

int i=0 ;

f.open("VOTER.DAT", ios::in);

fstream temp ;

temp.open("temp.dat", ios::out) ;

while (f.read((char \*) this, sizeof(VOTER)))

{

i++ ;

if (count == i)

{

cardstatus = 1 ;

gotoxy(10,10);

cout<<"Voter card issued";

temp.write((char \*) this, sizeof(VOTER)) ;

}

else

temp.write((char \*) this, sizeof(VOTER)) ;

}

f.close() ;

temp.close();

f.open("VOTER.DAT", ios::out) ;

temp.open("temp.dat", ios::in) ;

while (temp.read((char \*) this, sizeof(VOTER)))

f.write((char \*) this, sizeof(VOTER)) ;

f.close() ;

temp.close() ;

}

else

{

gotoxy(5,20) ;

cout <<"Vote no. not found" ;

}

gotoxy(5,24) ;

cout <<"Press any key to continue..." ;

getch();

}

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO GIVE VOTES BY THE VOTERS

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void VOTER :: ELECTION(void)

{

clrscr() ;

gotoxy(32,1) ;

cout<<"\*\*\* ELECTION \*\*\*";

fstream f ;

f.open("ELEDATE.DAT", ios::in) ;

if (f.fail())

{

gotoxy(5,20) ;

cout <<"Election date is not set." ;

getch();

return ;

}

f.close() ;

VOTING\_DATE voting\_date ;

gotoxy(5,3) ;

cout <<"Election Date : ";

voting\_date.RETURN\_DATE() ;

if (!voting\_date.VALID\_DATE())

{

gotoxy(8,20) ;

cout <<"Sorry! Today is not an election date" ;

getch() ;

return ;

}

char ch, vno[6] ;

do

{

gotoxy(5,8) ;

clreol() ;

cout <<"Enter Voter no. or Press <ENTER> to exit : " ;

gets(vno) ;

if (strlen(vno) == 0)

return ;

int count=0, found=0, valid=1 ;

f.open("VOTER.DAT", ios::in);

while (f.read((char \*) this, sizeof(VOTER)))

{

count++ ;

if (!strcmpi(vno,voterno))

{

found = 1 ;

break ;

}

}

f.close() ;

if (found)

{

if (cardstatus == 0)

{

valid = 0 ;

gotoxy(5,20) ;

cout <<"Voting card not issued" ;

getch();

}

if (status == 1)

{

valid=0;

gotoxy(5,20) ;

cout <<"Vote already given" ;

getch();

}

if (valid)

{

int i=0 ;

f.open("VOTER.DAT", ios::in) ;

fstream temp ;

temp.open("temp.dat", ios::out) ;

while (f.read((char \*) this, sizeof(VOTER)))

{

i++ ;

if (count == i)

{

status = 1 ;

gotoxy(5,20);

cout<<“Vote a||owed";

temp.write((char \*) this, sizeof(VOTER)) ;

}

else

temp.write((char \*) this, sizeof(VOTER)) ;

}

f.close() ;

temp.close() ;

f.open("VOTER.DAT", ios::out) ;

temp.open("temp.dat", ios::in) ;

while (temp.read((char \*) this, sizeof(VOTER)))

f.write((char \*) this, sizeof(VOTER)) ;

f.close() ;

temp.close() ;

gotoxy(5,20) ;

cout <<"Vote given" ;

}

}

else

{

gotoxy(5,20) ;

cout <<"Voter no. not found" ;

getch();

}

gotoxy(5,20) ;

clreol() ;

cout <<“More? (y/n) :" ;

ch = getche();

gotoxy(5,20) ;

clreol();

}while ((ch == 'y')||(ch == 'Y'));

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO DELETE VOTER RECORD FOR THE GIVEN VOTER NAME

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

int VOTER :: DELETE\_RECORD(char vno[6])

{

fstream file ;

file.open("VOTER.DAT", ios::in) ;

fstream temp ;

temp.open(“temp.dat“, ios::out) ;

int found=0 ;

while (file.read((char \*)this, sizeof(VOTER)))

{

if (!strcmpi(voterno,vno))

found = 1 ;

else

temp.write((char \*) this, sizeof(VOTER)) ;

}

file.close() ;

temp.close() ;

if (!found)

return 0 ;

file.open("VOTER.DAT", ios::out) ;

temp.open(“temp.dat“, ios::in) ;

while (temp.read((char \*) this, sizeof(VOTER)))

file.write((char \*) this, sizeof(VOTER)) ;

file.close() ;

temp.close();

return 1;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO GIVE THE VOTER NAME TO DELETE VOTER RECORD

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void VOTER :: DELETE\_VOTER(void)

{

clrscr() ;

gotoxy(27,1) ;

cout<<"\*\*\* DELETION 0F VOTER \*\*\*";

char vno[6] ;

gotoxy(5,5) ;

clreol() ;

cout <<"Enter Voter no. or Press <ENTER> to exit : " ;

gets(vno) ;

if (strlen(vno) == 0)

return ;

gotoxy(5,20) ;

if (!DELETE\_RECORD(vno))

cout <<"Voter no. not found" ;

else

cout <<"Voter record deleted" ;

getch();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO CALCULATE THE VOTES GIVEN BY THE VOTERS

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void VOTER :: VOTE\_CALCULATION(void)

{

clrscr() ;

CANDIDATE candidate;

int sno, valid=0, found=0;

fstream f ;

f.open("VOTER.DAT", ios::in) ;

while (f.read((char \*) this, sizeof(VOTER)))

{

if(status)

{

found = 1 ;

gotoxy(3,21) ;

clreol() ;

cout <<"Voter no. :” <<voterno ;

candidate.VOTlNG\_LIST() ;

do

{

valid = 1 ;

gotoxy(3,23) ;

clreol() ;

cout <<"Vote in the favour of : (Enter S.No.) " ;

cin >>sno;

if (!candidate.ADD\_VOTES(sno))

{

valid = 0 ;

gotoxy(3,23) ;

clreol() ;

cout <<"S.No. not found" ;

getch() ;

}

}while (!valid);

}

}

if (!found)

{

gotoxy(1,23) ;

clreol() ;

cout <<“No voter has given vote.“ ;

getch() ;

return ;

}

candidate.VOTING\_LIST() ;

gotoxy(1,21) ;

clreol() ;

gotoxy(3,23) ;

clreol() ;

cout <<“Vote Calculation Complete." ;

candidate.RESET\_STATUS() ;

getch() ;

}

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO DISPLAY THE CANDIDATE INFORMATION

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void CANDIDATE :: CANDIDATE\_INFORMATION(void)

{

clrscr() ;

int row=5, found=0 ;

char ch;

gotoxy(27,1) ;

cout<<"\*\*\* LIST OF CANDIDATES \*\*\*" ;

gotoxy(1,3) ;

cout <<"S.NO. CANDIDATE NAME SYMBOL“ ;

gotoxy(1,4) ;

cout<<”~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~`~~~~~~~~~” ;

fstream f ;

f.open("CAN.DA ", ios::in);

while (f.read((char \*) this, sizeof(CANDIDATE)))

{

found = 1 ;

gotoxy(1,row);

cout<<serialno;

gotoxy(8,row) ;

cout <<name ;

gotoxy(40,row) ;

cout <<symbol ;

row++;

if (!found)

{

gotoxy(5,10) ;

cout <<"Records not found “;

}

f.close() ;

}

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO ADD CANDIDATE RECORD IN THE CANDIDATE FILE

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void CANDIDATE :: ADD\_CANDIDATE(void)

{

char ch;

int sno;

do

{

int count=0 ;

fstream f ;

f.open("CAN.DAT", ios::in) ;

while (f.read((char \*) this, sizeof(CANDIDATE)))

count++ ;

f.close() ;

sno=count;

if (count >= 15)

{

cout <<"No. of Candidates are 15, no more candidates can be added." ;

getch();

return ;

}

clrscr() ;

sno++;

serialno=sno;

gotoxy(25,1) ;

cout<<"\*\*\* ADDITION OF CANDIDATES \*\*\*" ;

gotoxy(10,8) ;

cout<< "S.No:";

cout<<serialno;

gotoxy(10,12);

cout <<"Enter name of the Candidate : " ;

gets(name) ;

upper(name);

gotoxy(10,14) ;

cout <<"Enter symbol name : " ;

gets(symbol) ;

upper(symbol);

votes = 0 ;

gotoxy(10,16) ;

cout <<"Save Candidate record? (y/n) : " ;

ch = getche() ;

if ((ch == 'y‘) ||( ch == 'Y'))

{

fstream f ;

f.open("CAN.DAT", ios::app) ;

f.write((char \*)this, sizeof(CANDIDATE)) ;

f.close() ;

}

gotoxy(10,18) ;

cout <<"Add more? (y/n) :" ;

ch = getche() ;

}while ((ch== 'y' )|| (ch == 'Y'));

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO RESET VOTES RECEIVED BY CANDIDATE TO ZERO

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void CANDIDATE :: RESET\_STATUS(void)

{

fstream file ;

file.open("CAN.DAT“, ios::in) ;

if (file.fail())

return ;

fstream temp ;

temp.open("temp.dat", ios::out) ;

file.seekg(0,ios::beg) ;

while (!file.eof())

{

file.read((char \*) this, sizeof(CANDIDATE)) ;

if (file.eof())

break ;

votes = 0 ;

temp.write((char \*) this, sizeof(CANDIDATE)) ;

}

file.close() ;

temp.close() ;

file.open("CAN.DAT“, ios::out) ;

temp.open("temp.dat", ios::in) ;

temp.seekg(0,ios::beg) ;

while (!temp.eof())

{

temp.read((char \*) this, sizeof(CANDIDATE)) ;

if (temp.eof())

break ;

file.write(( char \*) this, sizeof(CANDIDATE));

}

file.close() ;

temp.close() ;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO DISPLAY THE LIST OF THE VOTES RECEIVED BY

// CANDIDATE

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void CANDIDATE :: VOTING\_LIST(void)

{

int row=5 ;

gotoxy(28,1) ;

cout<<"\*\*\* VOTE CALCULATION \*\*\*" ;

gotoxy(1,3) ;

cout <<"S.NO CANDIDATE NAME SYMBOL VOTES“;

gotoxy(1,4) ;

cout<<"~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~”;

fstream f;

f.open("CAN.DAT", ios::in) ;

while (f.read((char \*) this, sizeof(CANDIDATE)))

{

gotoxy(2,row) ;

cout <<serialno ;

gotoxy(10,row);

cout<<name;

gotoxy(40,row) ;

cout <<symbol ;

gotoxy(60,row) ;

cout<<votes ;

row++ ;

}

f.close() ;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO ADD VOTES IN THE FAVOUR OF GIVEN CANDIDATE

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

int CANDIDATE :: ADD\_VOTES(int sno)

{

fstream file ;

file.open(“CAN.DAT“, ios::in);

fstream temp ;

temp.open("temp.dat", ios::out);

int i=0, found=0 ;

while (file.read((char \*) this, sizeof(CANDIDATE)))

{

i++ ;

if (file.eof())

break;

if (sno == serialno)

{

found = 1;

votes++;

}

temp.write((char \*) this, sizeof(CANDIDATE)) ;

}

file.close() ;

temp.close() ;

if (!found)

return 0 ;

file.open("CAN.DAT", ios::out);

temp.open("temp.dat", ios::in);

while (!temp.eof()) ;

{

temp.read((char \*) this, sizeof(CANDIDATE)) ;

if (temp.eof())

break ;

file.write((char \*) this, sizeof(CANDIDATE)) ;

}

file.close() ;

temp.close() ;

return 1;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO DELETE RECORD OF THE GIVEN CANDIDATE

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

int CANDIDATE :: DELETE\_RECORD(int sno)

{

fstream file ;

file.open("CAN.DAT", ios::in) ;

fstream temp ;

temp.open("temp.dat", ios::out) ;

int found=0, count=0 ;

while (fIle.read((char\*)this, sizeof(CANDIDATE)))

{

count++ ;

if (sno == count)

found = 1 ;

else

temp.write((char \*) this, sizeof(CANDIDATE)) ;

}

file.close() ;

temp.close() ;

if (!found)

return 0 ;

file.open("CAN.DAT“, ios::out) ;

temp.open("temp.dat", ios::in) ;

while (temp.read((char \*) this, sizeof(CANDlDATE)))

file.write((char \*) this, sizeof(CANDIDATE)) ;

file.close() ;

temp.close() ;

return 1 ;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO GIVE CANDIDATE S.NO. TO DELETE RECORD

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void CANDIDATE :: DELETE\_CANDIDATE(void)

{

clrscr() ;

fstream f ;

f.open("CAN.DAT", ios::in) ;

if (f.fail())

{

gotoxy(5,5) ;

cout <<"Candidates not found" ;

getch() ;

return ;

}

f.close() ;

CANDIDATE\_INFORMATION();

int sno ;

gotoxy(3,23) ;

cout <<"Enter S.No. of the Candidate to be deleted :" ;

cin >>sno ;

if (!DELETE\_RECORD(sno))

{

gotoxy(3,23) ;

clreol() ;

cout<<”S.No not found”;

getch();

return;

}

gotoxy(3,23) ;

clreol() ;

cout <<“Record Deleted" ;

getch() ;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*\*\*\*

// FUNCTION TO RETURN THE ELECTION DATE

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void VOTING\_DATE :: RETURN\_DATE(void)

{

fstream f ;

f.open("ELEDATE.DAT", ios::in) ;

f.read((char\*) this, sizeof(VOTING\_DATE)) ;

f.close() ;

cout <<dd<<'/‘<<mm<<'/'<<yy;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO RETURN THE VALIDITY OF THE TODAY DATE

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

int VOTING\_DATE :: VALID\_DATE(void)

{

fstream f ;

f.open(“ELEDATE.DAT", ios::in) ;

f.read((char \*) this, sizeof(VOTING\_DATE)) ;

f.close() ;

int d1, m1, y1, valid=1;

struct date d;

getdate(&d);

d1 = d.da\_day;

m1 = d.da\_mon ;

y1 = d.da\_year;

if((d1!=dd )||( m1!=mm )||( y1!=yy))

valid = 0 ;

return valid ;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// FUNCTION TO SET THE DATE FOR THE NEW ELECTION

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void VOTING\_DATE :: SET\_DATE(void)

{

clrscr() ;

int d1, m1, y1, d2, m2, y2;

struct date d;

getdate(&d);

d1 = d.da\_day;

m1 = d.da\_mon;

y1 = d.da\_year;

gotoxy(26,1) ;

cout<<"\*\*\* SET NEW VOTING DATE \*\*\*" ;

gotoxy(8,8) ;

cout <<"Enter new Voting date (dd mm yyyy) : " ;

cin >>d2 >>m2 >>y2 ;

if ((y2<y1) ||(y2==y1 && m2<m1) || (y2==y1 && m2==m1 && d2<d1))

{

gotoxy(5,20) ;

cout <<“Invalid Date" ;

getch();

return ;

}

dd = d2 ;

mm=m2;

yy=y2;

fstream f ;

f.open("ELEDATE.DAT", ios::out) ;

f.write((char \*) this, sizeof(VOTING\_DATE)) ;

f.close() ;

f.open("ELEDATE.DAT", ios::in) ;

f.read((char \*) this, sizeof(VOTING\_DATE)) ;

f.close() ;

VOTER voter ;

voter.RESET\_STATUS() ;

CANDIDATE candidate ;

candidate.RESET\_STATUS() ;

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// MAIN FUNCTION DISPLAYING MENU AND CA||S OTHER FUNCTIONS

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void main(void)

{

char ch, ch1, c=22, yes ;

VOTER voter ;

CANDIDATE candidate ;

VOTING\_DATE voting\_date ;

DRAW draw ;

while(1)

{

clrscrO ;

draw.||NE\_HOR(2,79,2,219) ;

draw.LINE\_HOR(2,79,24,219) ;

draw.LINE\_VER(2,24,2,219);

draw.LINE\_VER(2,24,79,219) ;

gotoxy(32,8);

cout <<"ELECTION 2020";

gotoxy(31,9) ;

for (int i=1; i<=15; i++)

cout <<c ;

gotoxy(30,11) ;

cout <<"1: ISSUE VOTING CARDS" ;

gotoxy(30, 12) ;

cout <<"2: ELECTION" ;

gotoxy(30,13) ;

cout <<"3: ADD VOTERS" ;

gotoxy(30,14) ;

cout <<"4: ADD CANDIDATES" ;

gotoxy(30,15) ;

cout<<"5: CALCULATING VOTES“ ;

gotoxy(30,16) ;

cout <<"6: OPTIONS";

gotoxy(30,17) ;

cout <<"0: QUIT" ;

gotoxy(30,19) ;

cout <<"Enter your choice:" ;

ch = getch();

if (c == '0')

{

clrscr();

gotoxy(31,2) ;

cout<<"ELECTION 2020" ;

gotoxy(31,4) ;

cout<<"\n\n\n\n\n\t\t\t\tMADE BY : " ;

cout<<"\n\t\T MADHUR NARANG" ;

cout<<"\n\n\t\t DELHI INTERNATIONAL PUBLIC SCHOOL" ;

break;

}

else

if (ch == '1')

voter.ISSUE\_VOTING\_CARD() ;

else

if (ch == '2')

voter.ELECTION();

else

if (c == '3')

voter.ADD\_VOTER() ;

else

if (ch == '4')

candidate .ADD\_CANDIDATE() ;

else

if (ch == '5')

voter.VOTE\_CALCULATION() ;

else

if (ch == '6')

{

while(1)

{

clrscr() ;

draw.LINE\_HOR(2,79,2,178);

draw.LINE\_HOR(2,79,24,178) ;

draw.LINE\_VER(2,24,2,178) ;

draw.||NE\_\_VER(2,24,79,178) ;

gotoxy(38,9) ;

cout <<"OPTIONS";

gotoxy(37,10) ;

for (int i=1; i<=9; i++)

cout <<c ;

gotoxy(30,12) ;

cout <<"1: SET ELECTION DATE" ;

gotoxy(30,13) ;

cout <<"2: VOTER INFORMATION" ;

gotoxy(30, 14) :

cout <<"3: CANDIDATE INFORMATION" ;

gotoxy(30,15) ;

cout <<"4: DELETE VOTER" ;

gotoxy(30,16) ;

cout <<"5: DELETE CANDIDATE" ;

gotoxy(30,17) ;

cout <<"0: EXIT" ;

gotoxy(30,19) ;

cout <<"Enter your choice:" ;

ch1 = getch() ;

if (ch == '0')

break ;

else

if (ch == '1')

voting\_date.SET\_DATE() ;

else

if (ch1 == '2')

voter.VOTER\_lNFORMATION() ;

else

if (ch1 == '3')

{

candidate.CANDIDATE\_INFORMATION() ;

gotoxy(1,25) ;

cout <<"Press any key to continue..." ;

getch() ;

}

else

if (ch == '4')

voter.DELETE\_VOTER() ;

else

if (ch == '5')

candidate.DELETE \_CANDIDATE();

}

}

}

}