MINI – PROJECT (PPS)

**VOTING – MACHINE**

1. **INTRODUCTION / AIM :**

**In this project, we make a voting machine in which we have 4 different option to vote each user vote, see responses, to see who is winning and after all the user gives their vote.**

**we will conclude the result and we can check the winner and also the number of vote casted to each member.**

1. **ALGORITHM :**

**Step 1: START**

**Step 2: Defining names of candidates in user**

**defined functions**

**Step 3: Declaring Vote counts as variables**

**Step 4: Printing the options to cast the votes**

**Step 5: Taking the choice from the user**

**Step 6: switching the choice of votes of**

**candidates**

**Step 7: Increase 1 vote count for the candidate**

**which the user chose**

**Step 8: If the user didn't choose from the given**

**candidates print "Wrong choice"**

**Step 9: To count the votes of each individual**

**candidates**

**Step 10: If votes go to other than candidates then**

**print the numbers of spoiled votes**

**Step 11: To show the leading candidate use if-else**

**statement to find the candidate with the**

**highest number of votes**

**Step 12: Print the winning candidate's name**

**otherwise Print "No Winning situation" if**

**votes are a tie between candidates**

**Step 13: Declaring the choice as an integer**

**variable**

**Step 14: In the console screen if the user choose**

**1st option to cast the vote take him to**

**choose/vote the candidate**

**Step 15: If the user choose 2nd option to Find the**

**vote count take him to show the vote counts**

**of the candidates**

**Step 16: If the user choose 3rd option to Find the**

**leading candidate take him to show the**

**leading candidate**

**Step 17: If the user chooses the 0th option to Exit**

**to end the program and print Thanks for**

**Voting**

**Step 18: STOP**

1. **PROCEDURE :**
   1. **First we start by looking at the entry screen when we run the program.**
   2. **It starts by showing different options such as “Cast the vote” , “Find Vote Count”, “Finding leading Candidate”, “Exit”. Each of these options are assigned a number 1,2,3,0;**
   3. **The user is required to select one of these numbers as displayed in the options.**
   4. **When the user picks 1 as their choice they will be allowed to cast the vote between four candidates “1.** **Deekshitha”, “2.** **Debashish”, “3.****Jayesh”, “4.Vijeet”, “None of these”.**
   5. **The user is required to input one of these options to vote for the respective candidate.**
   6. **After the user cast’s their vote, the statement “Thanks for voting !!” is printed and the loop will start from start.**
   7. **Here after the vote is casted it will be stored as a vote for that particular candidate and the user is allowed to vote again.**
   8. **As these voting goes on the number of votes casted on one person will increase and the winner will be decided.**
   9. **After reaching to the menu again, if the user selects 2 from the menu which is “Find the vote count” then the code displays all the votes casted in that session.**
   10. **If 3rd option selected then the leading cadidate will be displayed according to the number of votes a person is leadning with compared to his peer cnadidates.**
   11. **If the last option “0” is slected then the code is exited and the statement “Thanks for Voting!!!” is printed.**
   12. **This loop continues as long as the voting is done.**

1. **CODE :**

#include<stdio.h>

#define CANDIDATE\_COUNT

#define CANDIDATE1 " Deekshitha "

#define CANDIDATE2 " Debashish "

#define CANDIDATE3 " Jayesh "

#define CANDIDATE4 " Vijeet "

int votesCount1 = 0, votesCount2 = 0, votesCount3 = 0, votesCount4 = 0, spoiledtvotes = 0;

void

castVote ()

{

int choice;

printf ("\n\n ### Please choose your Candidate ####\n\n");

printf ("\n 1. %s", CANDIDATE1);

printf ("\n 2. %s", CANDIDATE2);

printf ("\n 3. %s", CANDIDATE3);

printf ("\n 4. %s", CANDIDATE4);

printf ("\n 5. %s", "None of These");

printf ("\n\n Input your choice (1 - 4) : ");

scanf ("%d", &choice);

switch (choice)

{

case 1:

votesCount1++;

break;

case 2:

votesCount2++;

break;

case 3:

votesCount3++;

break;

case 4:

votesCount4++;

break;

case 5:

spoiledtvotes++;

break;

default:

printf ("\n Error: Wrong Choice !! Please retry");

getchar ();

}

printf ("\n thanks for vote !!");

}

void

votesCount ()

{

printf ("\n\n ##### Voting Statics ####");

printf ("\n %s - %d ", CANDIDATE1, votesCount1);

printf ("\n %s - %d ", CANDIDATE2, votesCount2);

printf ("\n %s - %d ", CANDIDATE3, votesCount3);

printf ("\n %s - %d ", CANDIDATE4, votesCount4);

printf ("\n %s - %d ", "Spoiled Votes", spoiledtvotes);

getchar ();

}

void

getLeadingCandidate ()

{

printf ("\n\n #### Leading Candiate ####\n\n");

if (votesCount1 > votesCount2 && votesCount1 > votesCount3 && votesCount1 > votesCount4)

printf ("[%s]", CANDIDATE1);

else if (votesCount2 > votesCount3 && votesCount2 > votesCount4 && votesCount2 > votesCount1)

printf ("[%s]", CANDIDATE2);

else if (votesCount3 > votesCount4 && votesCount3 > votesCount2 && votesCount3 > votesCount1)

printf ("[%s]", CANDIDATE3);

else if (votesCount4 > votesCount1 && votesCount4 > votesCount2 && votesCount4 > votesCount3)

printf ("[%s]", CANDIDATE4);

else

printf ("----- Warning !!! No-win situation----");

}

int main ()

{

int i;

int choice;

do

{

printf ("\n\n ###### Welcome to SRM Election/Voting 2022 ######");

printf ("\n\n 1. Cast the Vote");

printf ("\n 2. Find Vote Count");

printf ("\n 3. Find leading Candidate");

printf ("\n 0. Exit");

printf ("\n\n Please enter your choice : ");

scanf ("%d", &choice);

switch (choice)

{

case 1:

castVote ();

break;

case 2:

votesCount ();

break;

case 3:

getLeadingCandidate ();

break;

default:

printf ("\n ##### Thanks for using our voting code #####");

}

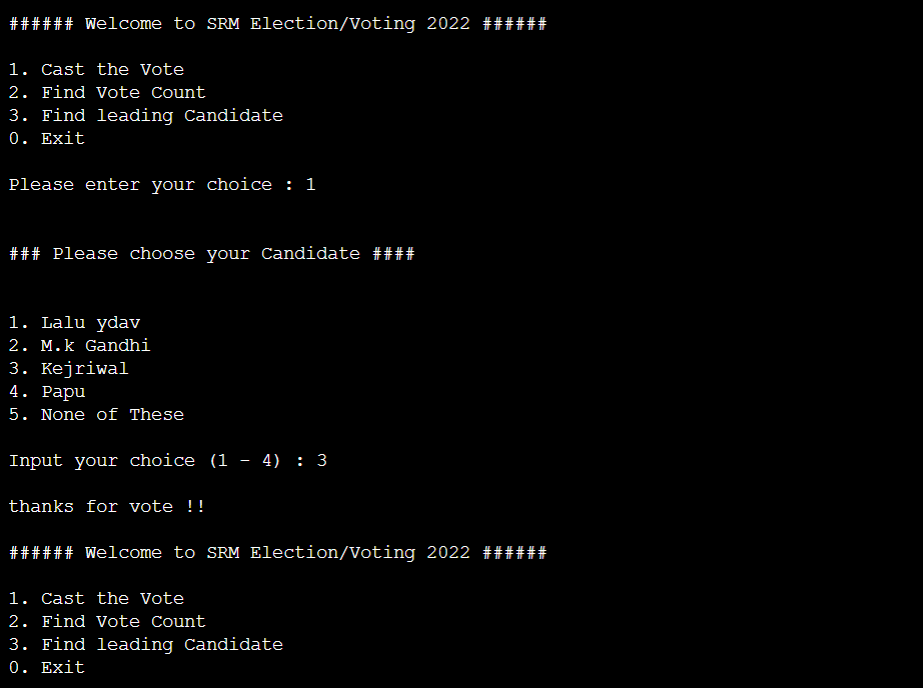
}

while (choice != 0);

return 0;

}

1. **OUTPUT SCREEN :**

****