```c
/*
Voting System Simulation using linked list (Project)
This is a custom voting system developed jointly by Nidish(094), Phanish(100) and Hemanth (063).
Do not use this code without permission from the authors ^^
it is meant to be run in stacked mode not side-by-side mode in settings of repl
*/

#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <string.h>
#include <unistd.h>
#include <ctype.h>

int count=0;
//add more pincode then increase range in randrompincodegen
int randrompincode[]={560098,560072,560004,560062,560085,560090,560070,560066,560050,560008};
//does micro sleep for rate higher rate slower typewriter, letter is the char buffer
void typewriter(const char* letter, int rate) {
  for (int i = 0; letter[i] != '\0'; i++) {
    printf("%c", letter[i]);
    fflush(stdout);//flushes the buffer::
    usleep(100 * rate);
  }
}
//The function randomly generates the inputs between the upper and the lower limits(range).
int singlerand(int lower, int upper)
{
 int num = (rand() % (upper - lower + 1)) + lower;
 // printf("%d\n",num); //for debugging:
 return num;
}
//generates random pincode using above function
int randrompincodegen()
{
 int i= singlerand(0, 9);
 return randrompincode[i];
}

//This is the main structure which
//is used to store data

struct node
{
 int voterid,age,pincode;
 enum gender{male=0, female=1}gender;
 int candidate;
 enum missed{notmiss=0, miss=1}missed;
 struct node * next;
}*first = NULL, *last = NULL, *temp = NULL, *temp1 = NULL;


//create a node and adds it to link list
//this is manual insertion
void create()
{
 int voterid,age,pincode,gender;
 temp = (struct node *)malloc(sizeof(struct node));
 voter:
  printf("\nEnter the voter details \n");
  printf("\nVoter id(int), Voter age(int), pincode(int),and gender(0 for male and 1 for female) :");
  scanf("%d %d %d %d", &voterid,&age,&pincode,&gender);
 if(age<18)
 {
  printf("\nAge is %d\n",age);
  printf("\nIneligible\n");
```

```
 printf("\nHow did you get your VoterID???\n");
 goto voter;
 }
 else if(age>110)
 {
 printf("\nAge is %d\n",age);
 printf("\n So Old are you alive\n");
 goto voter;
 }
 else if(gender != 0 && gender != 1)
 {
 printf("\nPlease enter either 0 or 1 for gender, we currently do not support any other inputs for gender \n");
 goto voter;
 }
 else
 {
 temp->voterid = voterid;
 temp->age = age;
 temp->pincode = pincode;
 temp->gender = gender;
 }
 printf("\nEnter the Candidate:  NOTA is any other number (other than 1,2,3,4,5)\n ");// 1:Murthy \n 2:Ramprasad \n
3:GuruPrasad \n 4:Modi \n 5:Rahul Gandhi\n
 printf(".----------------.  .----------------.  .----------------.  .----------------.  .----------------.  \n");
 printf("| .--------------. | | .--------------. | | .--------------. | | .--------------. | | .--------------. | \n");
 printf("| |     __     | | | |  _______   | | | |  _______   | | | _    _ | | | |  _________   | | \n");
 printf("| |    / |    | | | | / ___ `. | | | | / ____ `. | | | | | |  | | | | | |_____|  | | \n");
 printf("| |    `| |    | | | | |_/___) | | | | `'  __) | | | | | |__| |_| | | | | |_____    | | \n");
 printf("| |     | |    | | | |  .'____.' | | | |  _  |__ '. | | | |____   _| | | | '_.____''. | | \n");
 printf("| |    _| |_   | | | | / /_____  | | | | \\____) | | | | |     _| |_  | | | | \\____) | | | \n");
 printf("| |   |_____|  | | | | |_________| | | |  \\_______.' | | |    |_____| | | |  \\_______.' | | \n");
 printf("| |            | | | |           | | | |           | | | |           | | \n");
 printf("| '--------------' | | '--------------' | | '--------------' | | '--------------' | | '--------------' | \n");
 printf(" '----------------'  '----------------'  '----------------'  '----------------'  '----------------'  \n");
 printf("         |                |                |                |                |         \n");
 printf("\n|     Murthy     | |   Ramprasad    | |   GuruPrasad   | |      Modi      | |   Rahul Gandhi   |\n");

 printf("\nEnter Your Choice: ");
 scanf("%d",&temp->candidate);
 temp->next = NULL;
 count++;
}

//uses our random function to insert randomvoters
void createrandom()
{
 int missing= singlerand(0, 10);
 temp = (struct node *)malloc(sizeof(struct node));
 temp->voterid = singlerand(42069, 69420);
 temp->age = singlerand(18, 70);
 temp->pincode= randrompincodegen();
 temp->gender=singlerand(0, 1);
 temp->candidate=singlerand(1, 6);
 if (missing<1)
 {
 temp->missed=1;
 }
 else
 {
 temp->missed=0;
 }
 temp->next = NULL;
 count++;
}
```

```
//inserts into linked list
void insertrandomvoters(int n)
{
 for(int i=0;i<n;i++)
 {
  createrandom();
  if (first == NULL)
  {
   first = temp;
   last = first;
  }
  else
  {
   temp->next = first;
   first = temp;
  }
 }
}

//finds the largest element in array
int largestinArray(int arr[], int n)
{
 int i;
    int max = arr[0];
    for (i = 1; i < n; i++)
        if (arr[i] > max)
            max = arr[i];
    return max;
}

//This function counts the number of voters, missed voters and nota and displays who won the election and candidate specific statistics
void individualStats()
{
 char candidates[5][20]={"Murthy","Ramprasad","GuruPrasad","Modi","Rahul Gandhi"};
 int candidatevotes[5]={0};
 if (first == NULL)
 {
  printf("\n list is empty\n");
  return;
 }
 //do individual candidate stats here
 //1:Murthy \n 2:Ramprasad \n 3:GuruPrasad \n 4:Modi \n 5:Rahul Gandhi
 temp = first;
 int vcount=0;
 int missedv=0;
   while(temp!= NULL)
 {
  if(temp->missed == 1)
   missedv++;
  if(temp->candidate >= 1 && temp->candidate <= 5)
   vcount++;
  if(temp->missed==notmiss)
  {
   switch(temp->candidate)
    {
    case 1:
     candidatevotes[0]++;
     break;
    case 2:
     candidatevotes[1]++;
     break;
    case 3:
     candidatevotes[2]++;
     break;
    case 4:
```

```
    candidatevotes[3]++;
    break;
   case 5:
    candidatevotes[4]++;
    break;
  }
 }
 temp=temp->next;
}
vcount -= missedv;
int n = sizeof(candidatevotes)/sizeof(candidatevotes[0]);
int largest = largestinArray(candidatevotes, n);

char winnercan[20];
if(candidatevotes[0]== largest)
 strcpy(winnercan,"Murthy");
else if(candidatevotes[1]== largest)
 strcpy(winnercan,"Ramprasad");
else if(candidatevotes[2]== largest)
 strcpy(winnercan,"GuruPrasad");
else if(candidatevotes[3]== largest)
 strcpy(winnercan,"Modi");
else if(candidatevotes[4]== largest)
 strcpy(winnercan,"Rahul Gandhi");

printf("\nThere are %d voter(s) \n",count);
printf("\nTotal Votes issued for candidates (votes - nota - missed): %d\n",vcount);
printf("\n Candidate 1 (%s)       got %d votes",candidates[0],candidatevotes[0]);
printf("\n Candidate 2 (%s)    got %d votes",candidates[1],candidatevotes[1]);
printf("\n Candidate 3 (%s)   got %d votes",candidates[2],candidatevotes[2]);
printf("\n Candidate 4 (%s)         got %d votes",candidates[3],candidatevotes[3]);
printf("\n Candidate 5 (%s) got %d votes\n",candidates[4],candidatevotes[4]);
printf("\n                ELECTION WINNER                \n");
printf(" \t\t\t\t%s\n            with Largest Votes-
%d\n                                                                \n\n",winnercan,largest);

}

//This function is used to calculate statistics based on the various parameters that is given in the struct, i.e. voters in specific age
group, gender of voters, total number of voters, nota voters and missed voters
void calculatestats()
{
 if (first == NULL)
 {
  printf("\n list is empty\n");
  return;
 }
 // TODO WORK ON WHY PINCODE DOES NOT WORK
 int vcount = 0;
 int less25 = 0;
 int less50 = 0;
 int less100 = 0;
 int mvoter = 0;
 int fvoter = 0;
 int missedv =0;

 int pin_arr[]={0};
 int pincodes[10]={560098,560072,560004,560062,560085,560090,560070,560066,560050,560008};
 temp = first;
   while(temp!= NULL)
 {
  if(temp->missed == 1)
   missedv++;
  if(temp->candidate >= 1 && temp->candidate <= 5)
   vcount++;
```

```
 if(temp->age <= 25)
  less25++;
 if(temp->age > 25 && temp->age <=50)
  less50++;
 if(temp->age > 50 && temp->age <= 100)
  less100++;
 if(temp->gender == 0)
  mvoter++;
 if(temp->gender == 1)
  fvoter++;
 /* pin_arr[temp->pincode]++; */
 temp=temp->next;
}
vcount-=missedv;

printf("\nThere are %d voter(s) \n",count);
printf("\nTotal Votes issued for candidates (votes - nota - missed): %d\n",vcount);
printf("Total NOTA votes: %d\n",count-vcount-missedv);
printf("\nNumber of voters in age group(18-25):%d\n",less25);
printf("Number of voters in age group(25-50):%d\n",less50);
printf("Number of voters in age group(50-100):%d\n",less100);
printf("\nNumber of male voters : %d\n",mvoter);
printf("Number of female voters : %d\n",fvoter);
printf("\nNumber of Voters Who Missed to vote: %d\n",missedv);
/* for(int i=0;i<10;i++)
{
 int current_pin= pincodes[i];
 printf("No of votes in pincode %d is :%d \n",current_pin,pin_arr[current_pin]);
} */
}

//manual insertion at first
void insertatfirst()
{
 create();
 if (first == NULL)
 {
  first = temp;
  last = first;
 }
 else
 {
  temp->next = first;
  first = temp;
 }
}

//delete at front
void deletefront()
{
 temp = first;
 if (first == NULL)
 {
  printf("\n list is empty\n");
  return;
 }
 if (temp->next == NULL)

 {
  free(temp);
  first = NULL;
 }
 else
 {
  first = temp->next;
  free(temp);
```

```
 }
 count--;
}

//displays the voter list
void display()
{
 if (first == NULL)
 {
  printf("\n list is empty\n");
 }
 else
 {
  temp = first;
  printf("\nThere are %d voter(s) \n",count);
  printf("The voter is \n\n");
  printf("VoterID |  Age  |  Pincode  | gender  |  Voted Candidate  | Status | \n\n");
  while (temp != NULL)
  {
   char missed[20];
   if(temp->missed== miss)
    strcpy(missed,"Missed    ");
   else
    strcpy(missed,"Not Missed ");
   char gender1[20];
   if(temp->gender== male)
    strcpy(gender1," Male  ");
   else
    strcpy(gender1," Female ");
   char candidate[20];
   //1:Murthy \n 2:Ramprasad \n 3:GuruPrasad \n 4:Modi \n 5:Rahul Gandhi
   switch(temp->candidate)
   {
    case 1:
     strcpy(candidate,"Murthy       ");
     break;
    case 2:
     strcpy(candidate,"Ramprasad    ");
     break;
    case 3:
     strcpy(candidate,"GuruPrasad");
     break;
    case 4:
     strcpy(candidate,"Modi         ");
     break;
    case 5:
     strcpy(candidate,"Rahul Gandhi");
     break;
    default:
     strcpy(candidate,"NOTA         ");
   }
   if(temp->missed== miss)
    strcpy(candidate,"------    ");
   printf("%d\t\t %d\t \t%d\t %s \t %s \t   %s\n", temp->voterid,temp->age,temp->pincode,gender1,candidate,missed);
   temp = temp->next;
  }
 }
}

//main function with the menu of the program
int main(void)
{
 srand(time(0));
 typewriter("Hello! Welcome To the program \nThis is a custom voting system developed jointly by Nidish(094), Phanish(100)
and Hemanth (063). \nThis program is a menu driven program which also focuses on ease of use and helping us to understand
```

```
C data structures better. \n\t\t\t\t\t\tThank you.\n", 35);
 int ch, i, n;
 while (1)
 {
  printf("\n ------------Menu------------\n");
  typewriter("\n1.Insert n details of voters manually ",35);
  typewriter("\n2.Insert voter details manually",35);
  typewriter("\n3.Random Generate n voters",35);
  typewriter("\n4.Display Calulated stats(classification based stats)",30);
  typewriter("\n5.Display Individual Candidate stats(classification based candidate)",30);
  typewriter("\n6.Display voters",35);
  typewriter("\n7.Exit",35);
  printf("\n--------------------------------\n\n");
  typewriter("\nEnter your choice : ",35);
  scanf("%d", &ch);
  switch (ch)
  {
   case 1:
    printf("\nEnter the value of n: ");
    scanf("%d", &n);
    for (i = 0; i < n; i++)
     insertatfirst();
    break;
   case 2:
    insertatfirst();
    break;
   case 3:
    printf("\nEnter the value of n: ");
    scanf("%d", &n);
    insertrandomvoters(n);
    break;
   case 4:
    calculatestats();
    break;
   case 5:
    individualStats();
    break;
   case 6:
    display();
    break;
   case 7:
    exit(1);
   default:
    printf("\n Invalid Input, try again");
  }
 }
  return 0;
}
```