MINI PROJECT REPORT

INDIAN PREMIER LEAGUE PLAYERS REPORT USING EXCEL & ONLINE VOTING SYSTEM USING C PROGRAM

Submitted By

Name: Sibix Joy

Reg. No: THAUBVD046

Department of Data Science
St. Thomas' College (Autonomous), Thrissur

Under the Guidance of

Ms. Chinju Paul

Assistant Professor

Department of Data Science, St. Thomas College, Thrissur

Department of Data Science

St. Thomas' College (Autonomous), Thrissur

Affiliated To University of Calicut & Reaccredited By NAAC with "A" Grade & College with Potential Excellence

CERTIFICATE

This is to certify that the Mini Project Report titled **"INDIAN PREMIER LEAGUE PLAYERS REPORT USING EXCEL & ONLINE VOTING SYSTEM USING C** " is a bonafide record of the work carried out by **SIBIX JOY (THAUBVD046)** of St.Thomas' College (Autonomous) Thrissur - 680 005 in partial fulfillment of the requirements for the award of Degree of B.Voc Data Science of University of Calicut, during the academic year 2020-2023. The Mini Project report has been approved as it satisfies the academic requirements in the respect of mini project work prescribed for the said degree.

Head of the Department **Principal** **Internal Guide**

Valued On: …….

Examiners:

1. …….. **2. ……**

Internal Examiner External Examiner

DECLARATION

I hereby declare that the project report entitled **"INDIAN PREMIER LEAGUE PLAYER REPORT USING EXCEL & ONLINE VOTING SYSTEM USING C "** which is being submitted in partial fulfillment of the requirement of the award of the Degree in Bachelor of Vocational Studies in Data Science is the result of the project carried out by me under the guidance and supervision of **Ms. Chinju Paul,** Assistant Professor, Department of Data Science

I further declared that I or any other person has not previously submitted this project report to any other institution/university for any other degree/diploma or any other person.

Place : Thrissur **Sibix Joy**

Date : (Signature)

Indian Premier League players report

CONTENTS

Abstract

Chapter 1

1.1 Introduction
1.2 Aim
1.3 Data Description

Chapter2

2.1 Data Preprocessing
2.2Methodology
2.3Software Specification
2.4Data Analysis Result

Chapter3

3.1Conclusion

ABSTRACT

In this project we are evaluating the dataset of Indian Super League players pursuing the deep analysis of players and their performance.

Thus, through this excel project, our aim is to find best players from the 4 categories such as Batsman, Bowler, Allrounder, Wicket keeper. We can have a detailed analysis of data by using certain excel functions. We have used various excel functions like **MIN(), MAX(), AVERAGE(), COUNTIF()**, **PIVOT TABLE, GRAPHS, CHARTS and CONDITIONAL FORMATTING** to analyse the data.

The dataset used is available on KAGGLE website. The dataset consists of 197 individual data. There are 19 attributes such as **Name, Team, Batting Hand, Bowling Type, Role, Matches, Innings, Runs, Highest, BF, SR, 100, 50, four, six, Runs Given, Wickets, Economy, 5W.**

CHAPTER 1

1.1: INTRODUCTION

The Indian Premier League (IPL) is a professional Twenty20 cricket league, contested by eight teams based out of eight different Indian cities. The league was founded by the Board of Control for Cricket in India (BCCI) in 2007. It is usually held between March and May of every year and has an exclusive window in the ICC Future Tours Programme. The IPL is the most-attended cricket league in the world and ranks sixth among all sports leagues.

In 2010, the IPL became the first sporting event in the world to be broadcast live on YouTube. The brand value of IPL was estimated to be US$3.2 billion in 2014. According to BCCI, the 2015 IPL season contributed 11.5 billion (US$182 million) to the GDP of Indian economy. Until 2014, the top three teams in the tournament qualified for the Champions League Twenty20. However, the Champions League Twenty20 tournament discontinued in 2015 and has been defunct since.

1.2: AIM

i) THE AVERAGE OF MATCHES:-
To find the average of total matches held in IPL.

ii) THE HIGHEST RUNS:-
To find the most runs taken by a player in IPL.

iii) THE LOWEST INNINGS:-
To get the least innings played by a player.

Iv) HOW MANY OF THEM HAVE MORE THAN 4 CENTURY:-
To get the number of players who takes more than 4 Centutry.

v) HOW MANY OF THEM PLAYED MORE THAN 200 MACHES:-
To point out the number of experienced players in IPL

vi) SCATTER DIAGRAM OF STRIKE RATE PF BATSMAN:-
To get the more effective player in scoring runs.

vii) MAXIMUM INNINGS OF WICKET KEEPER:-
To find the most experienced wicket keeper.

viii) HIGHLIGHTS THE BOWLER WHO GIVES MORE THAN 1000 RUNS:-
To find the weak bowlers.

ix) HIGHLIGHTS THE BATSMAN WHO HAS TAKEN MORE THAN 5000 RUNS
To find the players who takes more than 5000 runs.

x) HIGHEST RUNS OF ALLROUNDERS:-
To find the top perfomence by allrounders.

1.3: DATA DESCRIPTION

This project is focused on the performance of IPL players. The original dataset is extracted from Kaggle website.

The dataset contains of 196 players and 19 attributes.

The 19 attributes are :

1) Name :

Displays the name of players.

2) Team :

Displays the team to which he belongs to.

3)Batting_Hand :

Displays the batting style of the batsman,

i)left

ii)right.

4) Bowling_Type :

Displays the bowling style of bowler.

5) Role :

Displays the role that the player plays majority, batsman, bowler, wicket-keeper, all rounder.

6) Matches :

Displays the number of matches played in his IPL career.

7) Innings :

Displays the number of times he batted in those matches.

8) Runs :

Displays the total number of runs scored.

9) Highest :

Displays the players highest score in the IPL career.

10) BF :

Displays the total number of balls faced by the player in IPL.

11) SR :

Displays the batting strike rate of batsman.

12) Century :

Displays the number of 100's the batsman has scored in IPL.

13) Half_century :

Displays the number of 50's the batsman has scored in IPL.

14) Four :

Displays the number of 4's hit by the batsman in IPL.

15) Six :

Displays the number of 6's hit by the batsman in IPL.

16) Runs_Given :

Displays the runs conceded by the bowler in IPL.

17) Wickets :

Displays the number of wickets that the bowler has picked in his IPL career.

18) Economy :

Displays the bowling economy in his IPL career.

19) 5W :

Displays the number of 5 wicket hauls in IPL career.

CHAPTER 2

2.1: DATA PREPROCESING

Data preprocessing is an important step in the data. It is used to transform raw data in a useful and efficient format. Analysing data that has not been carefully screened can produce misleading results. Thus, the representation and quality of data is first and foremost before running any analysis. Data preprocessing includes cleaning, instance selection, normalization, transformation, feature extraction and selection etc.

It is a good dataset with 197 rows and 28 attributes. To decrease the complexity of our dataset we drop 9 attributes named Not outs, Average, 200, Innings_Bowl, Balls, BBI, B_Average, BSR, 10W from our data set. And also we converted the attributes 100 to Century , 50 to Half Century, 4 to Four , 6 to Six .This modification doesn't affect our data set to obtain our aim.

2.2: METHODOLOGY

AIM 1:

To find the average of total matches held in IPL. We used AVERAGE().

AVERAGE ():- Returns the average of a group of number.

AVERAGE ():- =AVERAGE (number 1,[number 2]....)

=AVERAGE(F2:F197)

AIM 2:

To find the most runs taken by a player in IPL. We used MAX().

MIN ():-Returns the smallest value in the data.

MIN():- = MIN(number 1,[number 2].....).

=MAX(H2:H197)

AIM 3:

To get the least innings played by a player. We used MIN().

MAX():-Returns the largest value in the data.

MAX():-=MAX(number 1,[number 2],....).

=MIN(G2:G197)

AIM 4:

To get the number of players who takes more than 4 Centutry. We used COUNTIF().

COUNTIF := To count cells in a range that meet a single condition.

COUNTIF():- =COUNTIF (range, criteria)

=COUNTIF(L2:L197,">4")

AIM 5:

To point out the number of experienced players in IPL. We used COUNTIF().

COUNTIF := To count cells in a range that meet a single condition.

COUNTIF():- =COUNTIF (range, criteria)

=COUNTIF(F2:F197,">200")

AIM 6:

To get the more effective player in scoring runs. We used PIVOT TABLE.

Pivot Table:-A pivot table is a table of grouped values that aggregates the individual items of a more extensive table within one or more discrete categories. This summary might include sums, averages, or other statistics, which the pivot table groups together using a chosen aggregation function applied to the grouped values.

PIVOT TABLE:- Select any cell from the data ,select Insert and then Pivot table. Choose Role and Name and drag to Row Labels and ∑values(Average of SR).

By copying from PIVOT TABLE we use Scatter diagram to observe themost effective player

A scatter plot is a built-in chart type in Excel meant to show the relationship between two variables. A scatter plot works by placing one variable on the vertical axis and a different variable on the horizontal axis. Each piece of data is then plotted as a discrete point on the chart. In a scatter plot, both the X and Y axis display values – an XY chart has no category axis.

AIM 7:

To find the most experienced wicket keeper. We used PIVOT TABLE.

Pivot Table:-A pivot table is a table of grouped values that aggregates the individual items of a more extensive table within one or more discrete categories. This summary might include sums, averages, or other statistics, which the pivot table groups together using a chosen aggregation function applied to the grouped values.

PIVOT TABLE:- Select any cell from the data ,select Insert and then Pivot table. Choose Role and Name and drag to Row Labels and ∑values(Max of Innings).

By copying the PIVOT TABLE we use Line chart to find most experienced player.

A line graph (also called a line chart or run chart) is a simple but powerful tool and is generally used to show changes over time. Line graphs can include a single line for one data set, or multiple lines to compare two or more sets of data.

AIM 8:

To find the weak bowlers. We used PIVOT TABLE

Pivot Table:-A pivot table is a table of grouped values that aggregates the individual items of a more extensive table within one or more discrete categories. This summary might include sums, averages, or other statistics, which the pivot table groups together using a chosen aggregation function applied to the grouped values.

PIVOT TABLE:- Select any cell from the data ,select Insert and then Pivot table. Choose Role and Name and drag to Row Labels and ∑values(Min of Economy).

By copying from the PIVOT TABLE we use conditional formatting to find the weak bowlers

Conditional Formatting:-Conditional formatting allows to automatically apply formatting—such as colors , icons, and data bars—to one or more cells based on the cell value. Conditional formatting provides another way to visualise data and make worksheet easier to understand

. Conditional Formatting:- From the Home tab, click the Conditional Formatting command. A drop-down menu will appear , then select the desired rule from the menu that appears. Enter the desired value(s) into the blank field. Select a formatting style from the drop-down menu, then click O

AIM 9:

To find the players who takes more than 5000 runs.

Pivot Table:-A pivot table is a table of grouped values that aggregates the individual items of a more extensive table within one or more discrete categories. This summary might include sums, averages, or other statistics, which the pivot table groups together using a chosen aggregation function applied to the grouped values.

PIVOT TABLE:- Select any cell from the data ,select Insert and then Pivot table. Choose Role and Name and drag to Row Labels and ∑values(Max of Runs).

By copying from the PIVOT TABLE we use conditional formatting to find the weak bowlers

Conditional Formatting:-Conditional formatting allows to automatically apply formatting—such as colors , icons, and data bars—to one or more cells based on the

cell value. Conditional formatting provides another way to visualise data and make worksheet easier to understand.

Conditional Formatting:- From the Home tab, click the Conditional Formatting command. A drop-down menu will appear , then select the desired rule from the menu that appears. Enter the desired value(s) into the blank field. Select a formatting style from the drop-down menu, then click .

AIM 10:

To find the top perfomence by allroundersWe used PIVOT TABLE.

Pivot Table:-A pivot table is a table of grouped values that aggregates the individual items of a more extensive table within one or more discrete categories. This summary might include sums, averages, or other statistics, which the pivot table groups together using a chosen aggregation function applied to the grouped values.

PIVOT TABLE:- Select any cell from the data ,select Insert and then Pivot table. Choose Role and Name and drag to Row Labels and ∑values(Max of Highest).

By copying from the PIVOT TABLE we use Bar graph tofind top performance.

A bar chart is a graph that shows horizontal bars with the axis values for the bars displayed on the bottom of the graph. It is a graphical object used to represent the data in your Excel spreadsheet.

2.3: Software Specification

MICROSOFT EXCEL

Microsoft Excel is a spreadsheet developed by Microsoft for Windows, macOS, Android and iOS. It features calculation, graphing tools, pivot tables, and a macro programming language called Visual Basic for Applications (VBA).

Microsoft Excel has the basic features of all Spread sheets, using a grid of cells arranged in numbered rows and letter-named columns to organize data manipulations like arithmetic operations. It has a battery of supplied functions to answer statistical, engineering, and financial needs. In addition, it can display data as line graphs, histograms and charts, and with a very limited three-dimensional graphical display. It allows sectioning of data to view its dependencies on various factors for different perspectives (using pivot tables and the scenario manager).

Here, MS excel 2016 version is used. Excel 2016 has 484 functions. Of these, 360 existed prior to Excel 2010. Microsoft classifies these functions in 14 categories. Of the 484 current functions, 386 may be called from VBA as methods of the object "Worksheet Function" and 44 have the same names as VBA functions.

2.4: DATA ANALYSIS RESULTS

AIM 1:

THE AVERAGE OF MATCHES	45.6377551

Using AVERAGE() function of Excel, we will get the average matches that played in Indian Premier League. Here we get 45.6377551. So we conclude that average value for matches is 45.6377551.

AIM 2:

THE HIGHEST RUNS	5878

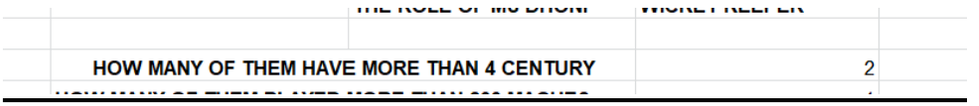
Using MAX() of excel, we will get the largest numeric value in attribute Runs. By doing this we get 5878 as the largest value. Thus we can finalize that most runs taken in attribute Runs is 5878.

AIM 3:

THE LOWEST INNINGS	0

Using MIN () function of excel, we get a detailed idea of the smallest numeric value in the attribute Innings. Thus we get 0 as the smallest value. So we can conclude that least innings played by a player in attribute Innings is 0

AIM 4:



HOW MANY OF THEM HAVE MORE THAN 4 CENTURY	2

Using COUNTIF() function of excel, we get a detailed idea of the numbers of data in the attribute Century under condition. Thus we get 2 as the count value. So we can conclude that there are 2 players who have scored Century more than 2 times

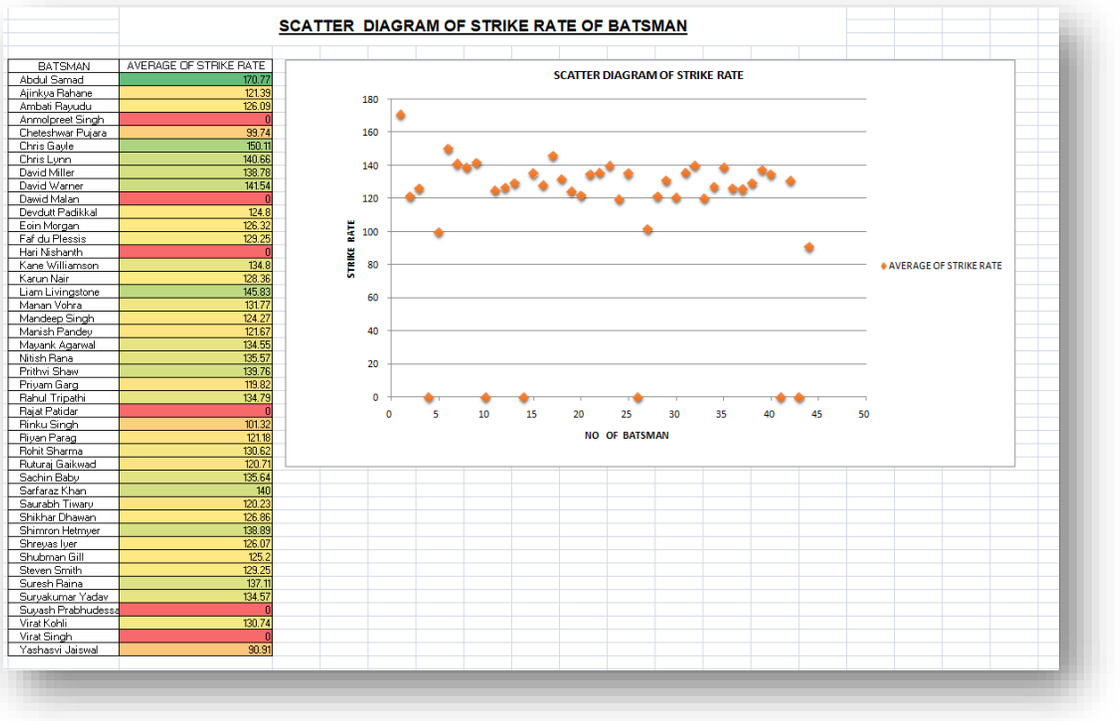
AIM 5:

HOW MANY OF THEM PLAYED MORE THAN 200 MACHES	1

Using COUNTIF() function of excel, we get a detailed idea of the numbers of data in the attribute Matches . Thus we get 1 as the count value. So we can conclude that there is only 1 player who played more than 200 matches.

AIM 6:

Here we use scatter diagram to obtain our aim. Scater diagram is a graph in which the values of two variables are plotted along two axes, the pattern of the resulting points revealing any correlation present.The equation of a scatter plot can be obtained by hand, using either of two main ways: a graphical technique or a technique called linear regression.

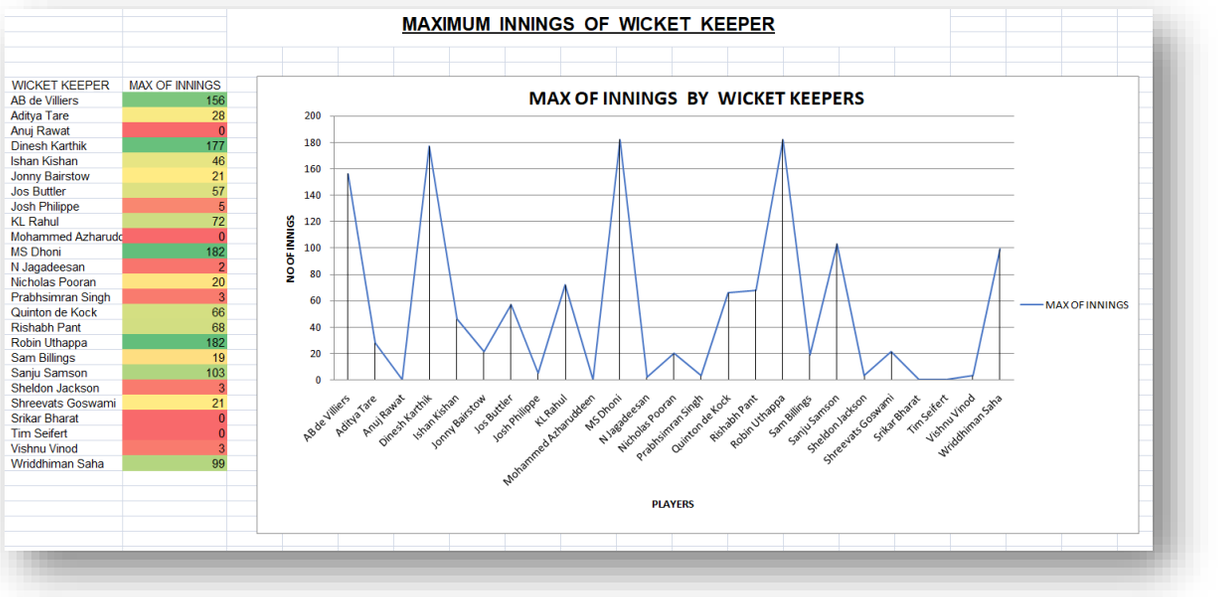


BATSMAN	AVERAGE OF STRIKE RATE
Abdul Samad	170.77
Ajinkya Rahane	121.39
Ambati Rayudu	126.09
Anmolpreet Singh	0
Cheteshwar Pujara	99.74
Chris Gayle	150.11
Chris Lynn	140.66
David Miller	138.78
David Warner	141.54
Dawid Malan	0
Devdutt Padikkal	124.8
Eoin Morgan	126.32
Faf du Plessis	129.25
Hari Nisharth	0
Kane Williamson	134.8
Karun Nair	128.36
Liam Livingstone	145.83
Manan Vohra	131.77
Mandeep Singh	124.27
Manish Pandey	121.67
Mayank Agarwal	134.55
Nitish Rana	135.57
Prithvi Shaw	139.76
Priyam Garg	119.82
Rahul Tripathi	134.79
Rajat Patidar	0
Rinku Singh	101.32
Riyan Parag	121.18
Rohit Sharma	130.62
Ruturaj Gaikwad	120.71
Sachin Baby	135.64
Sarfaraz Khan	140
Saurabh Tiwary	120.23
Shikhar Dhawan	126.86
Shimron Hetmyer	138.89
Shreyas Iyer	126.07
Shubman Gill	125.2
Steven Smith	129.25
Suresh Raina	137.11
Suryakumar Yadav	134.57
Suyash Prabhudessai	0
Virat Kohli	130.74
Virat Singh	0
Yashasvi Jaiswal	90.91

The picture shows the scatter diagram of strike rate of batsman. With the help of the scatter diagram we can pictorially observe the table that the strike rate of batsman where lowest to highest. From the analysis we can conclude that the highest strike rate is 170.77 by Abdul Samad .

AIM 7:

Here we use line chart to obtain our aim.A line chart is a graph that shows a series of data points connected by straight lines. It is a graphical object used to represent the data in your Excel spreadsheet.

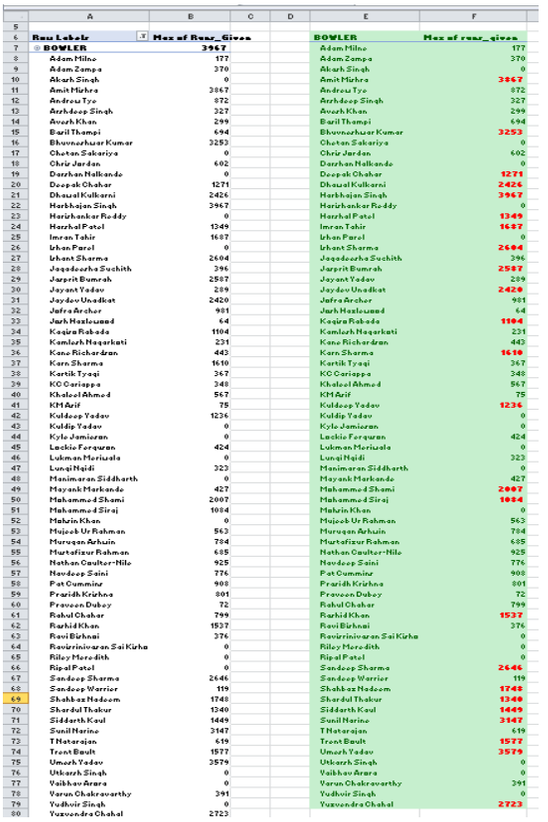


WICKET KEEPER	MAX OF INNINGS
AB de Villiers	156
Aditya Tare	28
Anuj Rawat	0
Dinesh Karthik	177
Ishan Kishan	46
Jonny Bairstow	21
Jos Buttler	57
Josh Philippe	5
KL Rahul	72
Mohammed Azharudd	0
MS Dhoni	182
N Jagadeesan	2
Nicholas Pooran	20
Prabhsimran Singh	3
Quinton de Kock	66
Rishabh Pant	68
Robin Uthappa	182
Sam Billings	19
Sanju Samson	103
Sheldon Jackson	3
Shreevats Goswami	21
Srikar Bharat	0
Tim Seifert	0
Vishnu Vinod	3
Wriddhiman Saha	99

The picture shows the line chart of maximum innings of wicket keeper. Thus we can see that the data set contains 25 players. With the help of line chart we can pictorially observe the maximum Innings of wicket keepers.From the analysis we can conclude that maximum innings is 170.77 by MS Dhoni and Robin Uthappa.

AIM 8:

Here we use conditional formatting to obtain our aim. Conditional formatting is a feature in many spreadsheet applications that allows you to apply specific formatting to cells that meet certain criteria. It is most often used as colour-based formatting to highlight, emphasize, or differentiate among data and information stored in a spreadsheet.



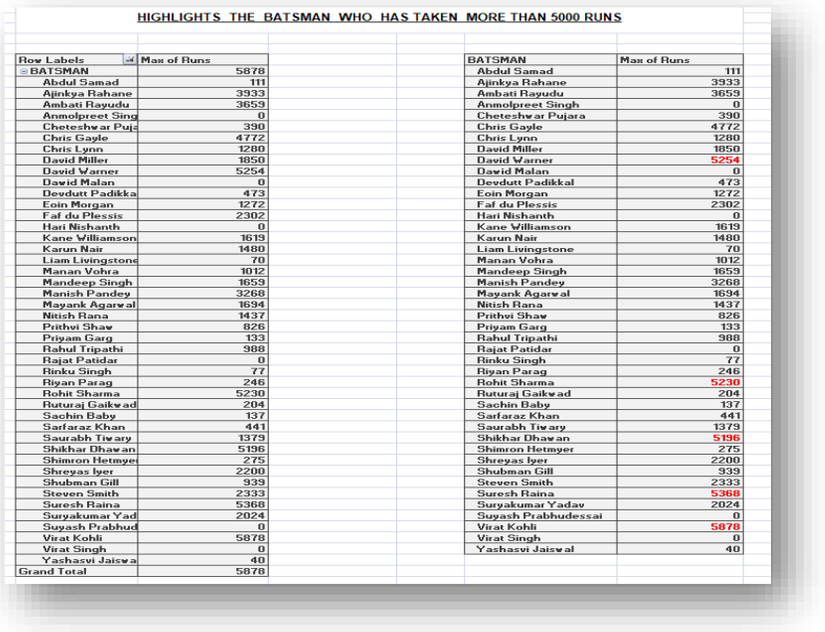
Row Labels	Max of Runs_Given
BOWLER	3967
Adam Milne	177
Adam Zampa	370
Akash Singh	0
Amit Mishra	3867
Andrew Tye	872
Arshdeep Singh	327
Avesh Khan	299
Basil Thampi	694
Bhuvneshwar Kumar	3253
Chetan Sakariya	0
Chris Jordan	602
Darshan Nalkande	0
Deepak Chahar	1271
Dhaval Kulkarni	2426
Harbhajan Singh	3967
Harishankar Reddy	0
Harshal Patel	1349
Imran Tahir	1687
Ishan Porel	0
Ishant Sharma	2694
Jagadeesha Suchith	396
Jasprit Bumrah	2587
Jayant Yadav	299
Jaydev Unadkat	2428
Jofra Archer	981
Jush Hazlewood	64
Kagiso Rabada	1104
Kamlesh Nagarkoti	231
Kane Richardson	443
Karn Sharma	1610
Kartik Tyagi	367
KC Cariappa	348
Khaleel Ahmed	567
KM Asif	75
Kuldeep Yadav	1236
Kuldip Yadav	0
Kyle Jamieson	0
Lockie Ferguson	424
Lukman Meriwala	0
Lungi Ngidi	323
Manimaran Siddharth	0
Mayank Markande	427
Mohammad Shami	2907
Mohammad Siraj	1984
Mohsin Khan	0
Mujeeb Ur Rahman	563
Murugan Ashwin	784
Mustafizur Rahman	685
Nathan Coulter-Nile	925
Navdeep Saini	776
Pat Cummins	908
Prasidh Krishna	801
Praveen Dubey	72
Rahul Chahar	799
Rashid Khan	1537
Ravi Bishnoi	376
Ravisrinivasan Sai Kisho	0
Riley Meredith	0
Rishi Patel	0
Sandeep Sharma	2646
Sandeep Warrier	119
Shahbaz Nadeem	1748
Shardul Thakur	1340
Siddarth Kaul	1449
Sunil Narine	3147
T Natarajan	619
Trent Boult	1577
Umesh Yadav	3579
Utkarsh Singh	0
Vaibhav Arora	0
Varun Chakravarthy	391
Yudhvir Singh	0
Yuzvendra Chahal	2723

BOWLER	Max of runs_given
Adam Milne	177
Adam Zampa	370
Akash Singh	0
Amit Mishra	3867
Andrew Tye	872
Arshdeep Singh	327
Avesh Khan	299
Basil Thampi	694
Bhuvneshwar Kumar	3253
Chetan Sakariya	0
Chris Jordan	602
Darshan Nalkande	0
Deepak Chahar	1271
Dhaval Kulkarni	2426
Harbhajan Singh	3967
Harishankar Reddy	0
Harshal Patel	1349
Imran Tahir	1687
Ishan Porel	0
Ishant Sharma	2694
Jagadeesha Suchith	396
Jasprit Bumrah	2587
Jayant Yadav	299
Jaydev Unadkat	2428
Jofra Archer	981
Jush Hazlewood	64
Kagiso Rabada	1104
Kamlesh Nagarkoti	231
Kane Richardson	443
Karn Sharma	1610
Kartik Tyagi	367
KC Cariappa	348
Khaleel Ahmed	567
KM Asif	75
Kuldeep Yadav	1236
Kuldip Yadav	0
Kyle Jamieson	0
Lockie Ferguson	424
Lukman Meriwala	0
Lungi Ngidi	323
Manimaran Siddharth	0
Mayank Markande	427
Mohammad Shami	2907
Mohammad Siraj	1984
Mohsin Khan	0
Mujeeb Ur Rahman	563
Murugan Ashwin	784
Mustafizur Rahman	685
Nathan Coulter-Nile	925
Navdeep Saini	776
Pat Cummins	908
Prasidh Krishna	801
Praveen Dubey	72
Rahul Chahar	799
Rashid Khan	1537
Ravi Bishnoi	376
Ravisrinivasan Sai Kisho	0
Riley Meredith	0
Rishi Patel	0
Sandeep Sharma	2646
Sandeep Warrier	119
Shahbaz Nadeem	1748
Shardul Thakur	1340
Siddarth Kaul	1449
Sunil Narine	3147
T Natarajan	619
Trent Boult	1577
Umesh Yadav	3579
Utkarsh Singh	0
Vaibhav Arora	0
Varun Chakravarthy	391
Yudhvir Singh	0
Yuzvendra Chahal	2723

The picture shows that the runs given by bowler. Conditional formatting helps us to identify the bowlers who give more than 1000 runs. We find there are 24 players who give more than 1000 runs.

AIM 9:

Here we use conditional formatting to obtain our aim. Conditional formatting is a feature in many spreadsheet applications that allows you to apply specific formatting to cells that meet certain criteria. It is most often used as colour-based formatting to highlight, emphasize, or differentiate among data and information stored in a spreadsheet.



HIGHLIGHTS THE BATSMAN WHO HAS TAKEN MORE THAN 5000 RUNS

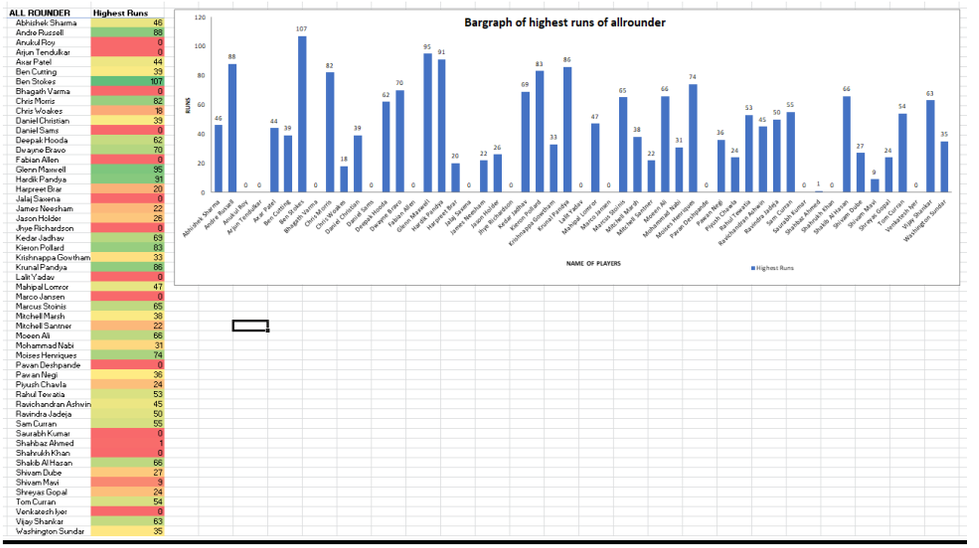
Row Labels	Max of Runs
BATSMAN	5878
Abdul Samad	111
Ajinkya Rahane	3933
Ambati Rayudu	3659
Anmolpreet Sing	0
Cheteshwar Puja	390
Chris Gayle	4772
Chris Lynn	1280
David Miller	1850
David Warner	5254
Dawid Malan	0
Devdutt Padikka	473
Eoin Morgan	1272
Faf du Plessis	2302
Hari Nishanth	0
Kane Williamson	1619
Karun Nair	1480
Liam Livingstone	70
Manan Vohra	1012
Mandeep Singh	1659
Manish Pandey	3268
Mayank Agarwal	1694
Nitish Rana	1437
Prithvi Shaw	826
Priyam Garg	133
Rahul Tripathi	988
Rajat Patidar	0
Rinku Singh	77
Riyan Parag	246
Rohit Sharma	5230
Ruturaj Gaikwad	204
Sachin Baby	137
Sarfaraz Khan	441
Saurabh Tiwary	1379
Shikhar Dhawan	5196
Shimron Hetmyer	275
Shreyas Iyer	2200
Shubman Gill	939
Steven Smith	2333
Suresh Raina	5368
Suryakumar Yad	2024
Suyash Prabhud	0
Virat Kohli	5878
Virat Singh	0
Yashasvi Jaiswa	40
Grand Total	5878

BATSMAN	Max of Runs
Abdul Samad	111
Ajinkya Rahane	3933
Ambati Rayudu	3659
Anmolpreet Singh	0
Cheteshwar Pujara	390
Chris Gayle	4772
Chris Lynn	1280
David Miller	1850
David Warner	5254
Dawid Malan	0
Devdutt Padikkal	473
Eoin Morgan	1272
Faf du Plessis	2302
Hari Nishanth	0
Kane Williamson	1619
Karun Nair	1480
Liam Livingstone	70
Manan Vohra	1012
Mandeep Singh	1659
Manish Pandey	3268
Mayank Agarwal	1694
Nitish Rana	1437
Prithvi Shaw	826
Priyam Garg	133
Rahul Tripathi	988
Rajat Patidar	0
Rinku Singh	77
Riyan Parag	246
Rohit Sharma	5230
Ruturaj Gaikwad	204
Sachin Baby	137
Sarfaraz Khan	441
Saurabh Tiwary	1379
Shikhar Dhawan	5196
Shimron Hetmyer	275
Shreyas Iyer	2200
Shubman Gill	939
Steven Smith	2333
Suresh Raina	5368
Suryakumar Yadav	2024
Suyash Prabhudessai	0
Virat Kohli	5878
Virat Singh	0
Yashasvi Jaiswal	40

The figure shows that the batsman who has taken more than 5000 runs. With the help of conditional formatting we find the players with more than 5000 runs. We can conclude that there are five players who take more than 5000 runs. They are David Warner, Rohith Sharma, Shikhar Dhawan, Suresh raina and Virat Kohli.

AIM 10:

Here we use bar graph to obtain our aim. A bar graph is a graph that shows horizontal bars with the axis values for the bars displayed on the bottom of the graph. It is a graphical object used to represent the data in your Excel spreadsheet. You can use a bar graph when: You want to compare values across categories.



ALL ROUNDER	Highest Runs
Abhishek Sharma	46
Andre Russell	88
Anukul Roy	0
Arjun Tendulkar	0
Axar Patel	44
Ben Cutting	39
Ben Stokes	107
Bhagath Varma	0
Chris Morris	82
Chris Woakes	18
Daniel Christian	39
Daniel Sams	0
Deepak Hooda	62
Dwayne Bravo	70
Fabian Allen	0
Glenn Maxwell	95
Hardik Pandya	91
Harpreet Brar	20
Jalaj Saxena	0
James Neesham	22
Jason Holder	26
Jhye Richardson	0
Kedar Jadhav	69
Kieron Pollard	83
Krishnappa Gowtham	33
Krunal Pandya	86
Lalit Yadav	0
Mahipal Lomror	47
Marco Jansen	0
Marcus Stoinis	65
Mitchell Marsh	38
Mitchell Santner	22
Moeen Ali	66
Mohammad Nabi	31
Moises Henriques	74
Pawan Deshpande	0
Pawan Negi	36
Piyush Chawla	24
Rahul Tewatia	53
Ravichandran Ashwin	45
Ravindra Jadeja	50
Sam Curran	55
Saurabh Kumar	0
Shahbaz Ahmed	1
Shahrukh Khan	0
Shakib Al Hasan	66
Shivam Dube	27
Shivam Mavi	9
Shreyas Gopal	24
Tom Curran	54
Venkatesh Iyer	0
Vijay Shankar	63
Washington Sundar	35

In the above figure we can visually see that the highest runs of allrounders with the help of bar graph.

Through this we can easily find the highest player who take more runs. From the analysis we found the player with highest runs is 107 by Ben Stokes.

CHAPTER 3:

Conclusion

As we all know Cricket is a bat-and-ball game played between two teams of eleven players on a field at the centre of which is a 22-yard (20-metre) pitch with a wicket at each end, each comprising two bails balanced on three stumps. Now a days Indian Premier League is famous T20 cricket tournament. The Indian Premier League is a professional Twenty20 cricket league, contested by eight teams based out of eight different Indian cities. The league was founded by the Board of Control for Cricket in India in 2007

From this project we can predict the players who can get more Bid value in the next auction of Indian Premier League by analysing the data set. In this project we analyse the statistics of 196 players. We useed excel functions to analyse the data. This Project helps to find the best player in all four categories such as Batsman, Bowler, Allrounder, Wicket keeper .

Online voting system

(using turbo c++)

CONTENTS

Abstract

Chapter 1

1.1 Introduction

1.2 Aim

Chapter 2

2.1 Methodology

2.2 Algorithm

2.3 Code

2.4 Output

Chapter 3

3.1 Conclusion

ABSTRACT

The election voting system based on concept of casting the vote for the candidates. The system allows the user or voters to casttheir votes after choosing, the system displays 3 options

1'cast the vote

2*find vote count

3*find leading candidate

Allowing the user to choose one among them.In 1st option the system shows the nameof candidates for the voter to caste the vote. After the votingthe screen will refresh. The board members who conduct the election canonly access the other 2 options with a passcode that we set. Bythe 2nd option we can known how much vote did each candidateget. In 3rd option we can know the leading candidate.

In the program we use #define,switch, break, if-else, do-while.

CHAPTER1

1.1 :INTRODUCTION

An online voting system is a software platform that allows groups to securely conduct votes and elections. High-quality online voting systems balance ballot security, accessibility, and the overall requirements of an organization's voting event At their core, online voting systems protect the integrity of your vote by preventing voters from being able to vote multiple times. As a digital platform, they eliminate the need to gather in-person, cast votes using paper, or by any other means. A secure voting tool that allows your group to collect input from your group and closely scrutinize the results in real time.

In "ONLINE VOTING SYSTEM" a voter can use his/her voting right online without any difficulty. He / She has to be registered first for him/her to vote. Registration is mainly done by the system administrator for security reasons. The system Administrator registers the voters on a special site of the system visited by him only by simply filling a registration form to register voter. After registration, the voter can use their admission number to log into the system and can enjoy services provide by the system such as voting. If invalid/wrong details are submitted, then the student is not registered to vote.

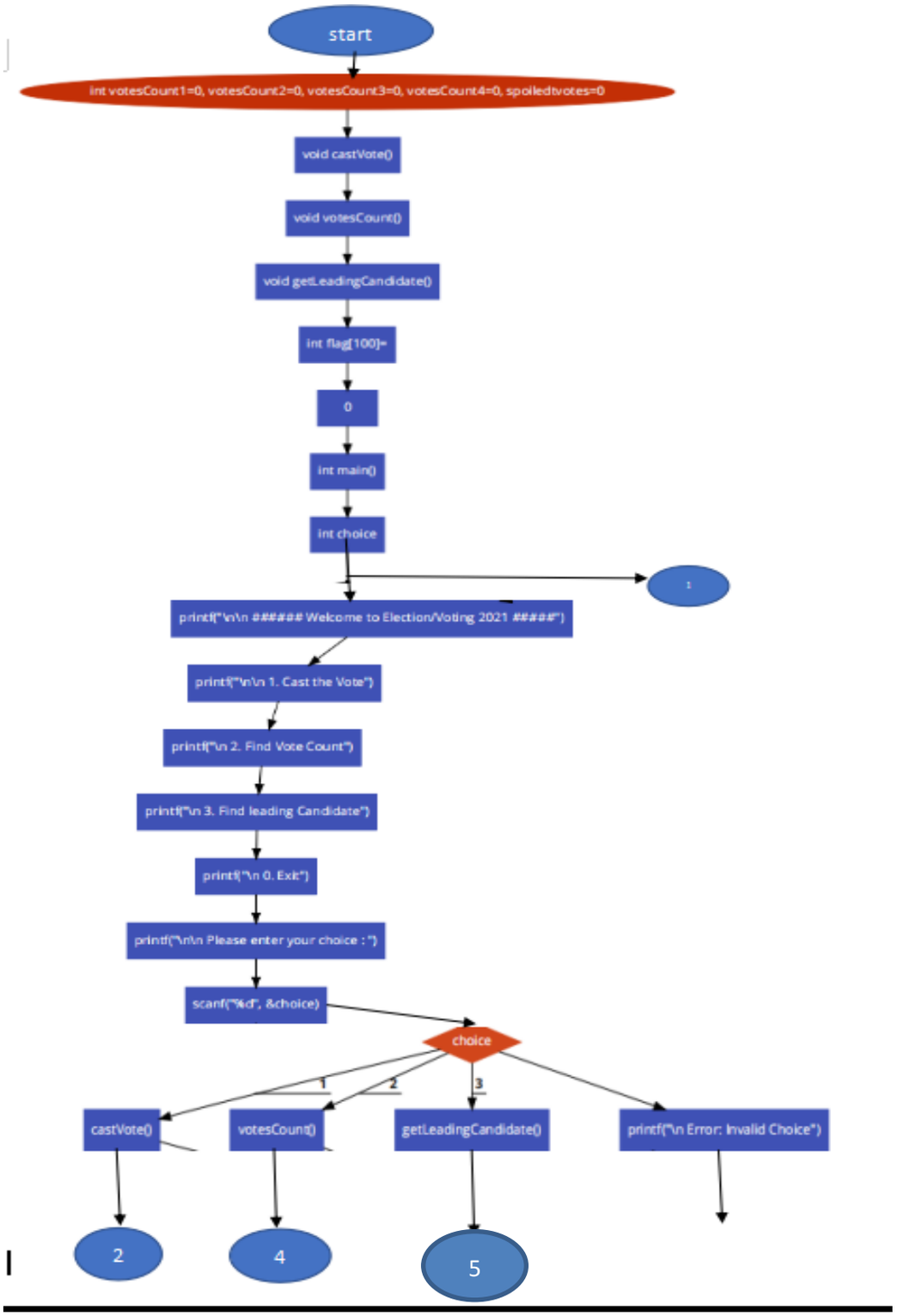
1.2:**AIM**

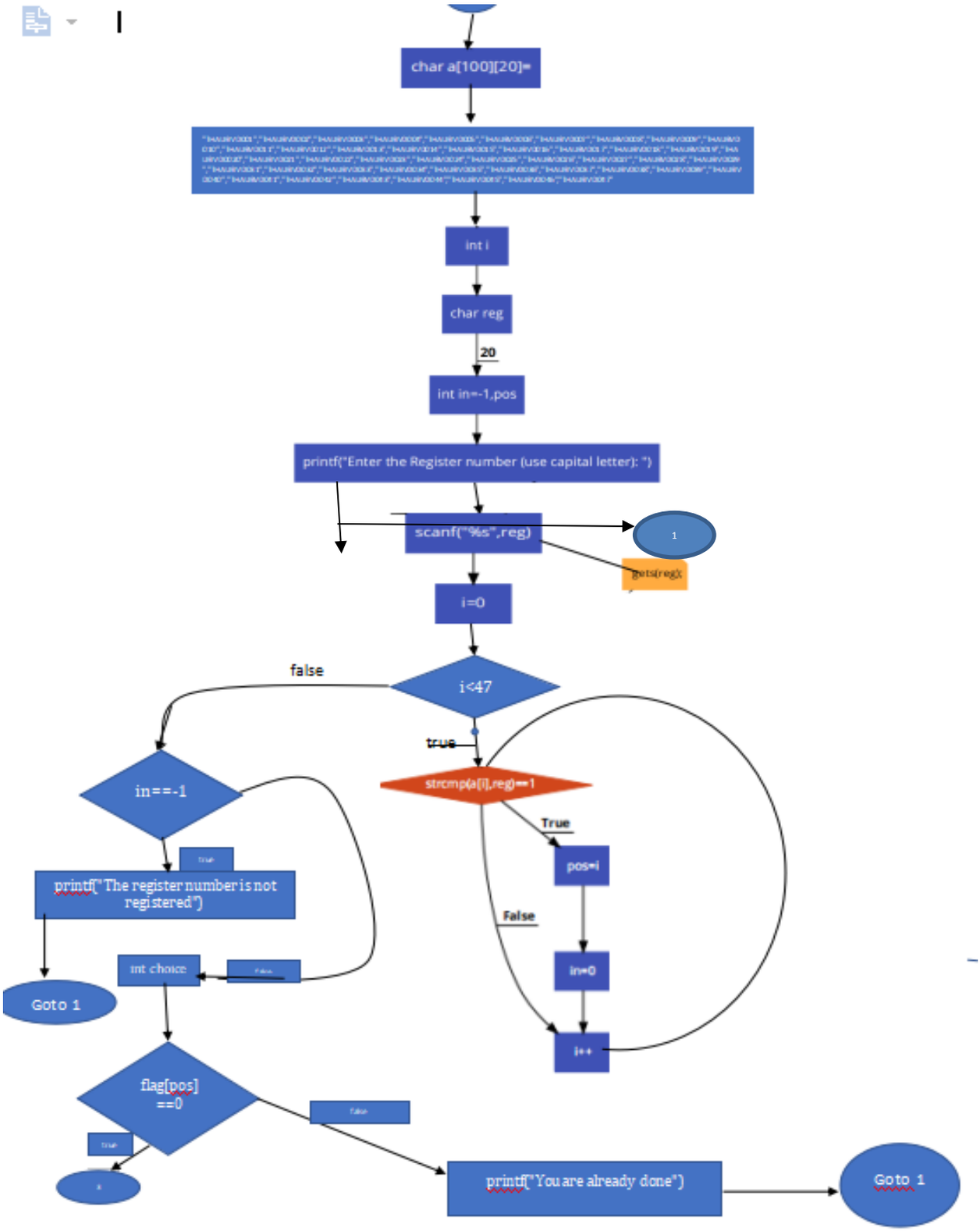
- **Maintaining voter's Identification.**
- **Providing online voting management.**
- **Providing Updation of voter's information.**
- **Provide voter information to COLLEGE ELECTION COMMISION.**
- **COLLEGE ELECTION COMMISION maintains the complete information of voter.**
- **Voter can give his\her vote from any part of India.**

CHAPTER 2

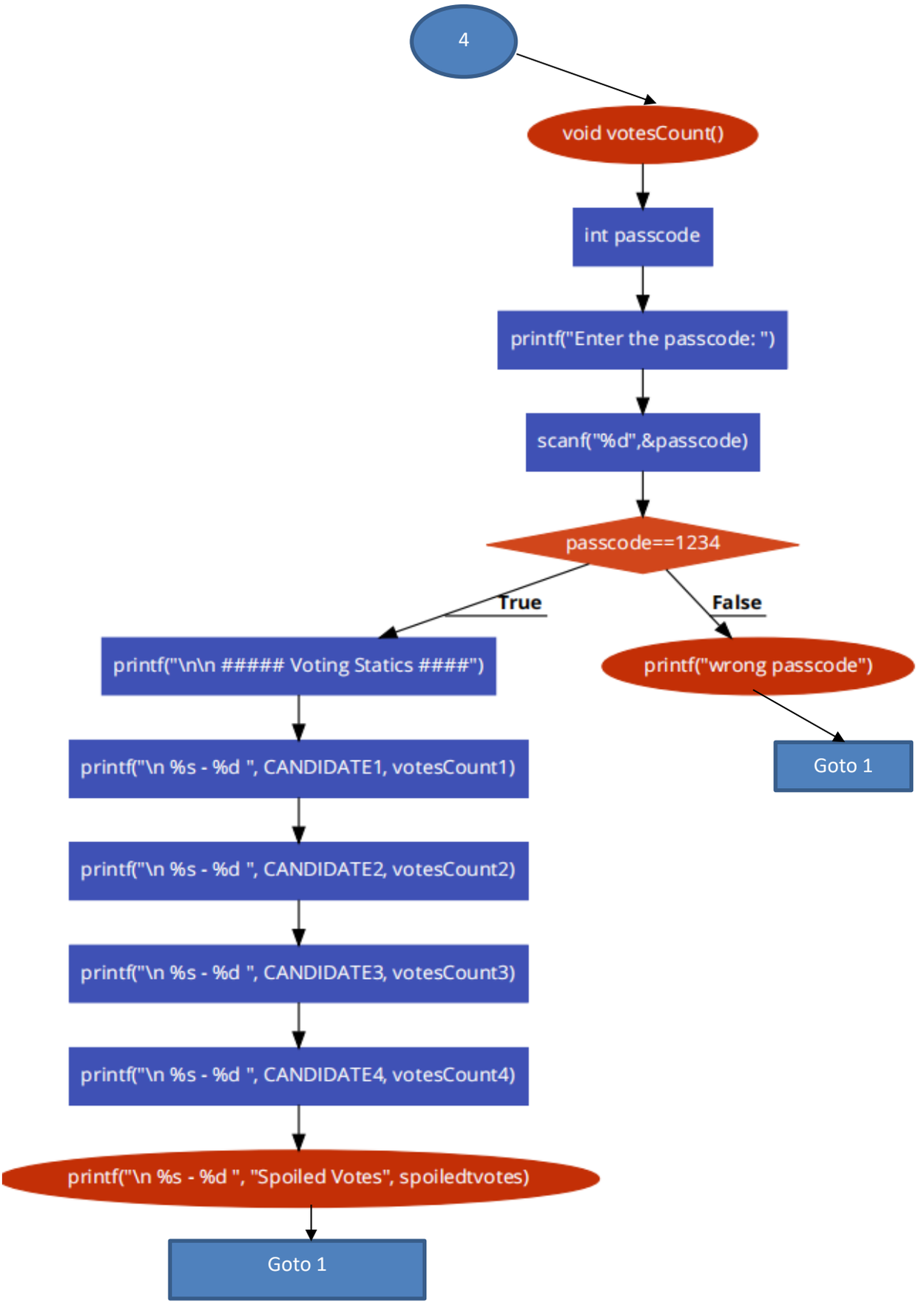
2.1: Methodology

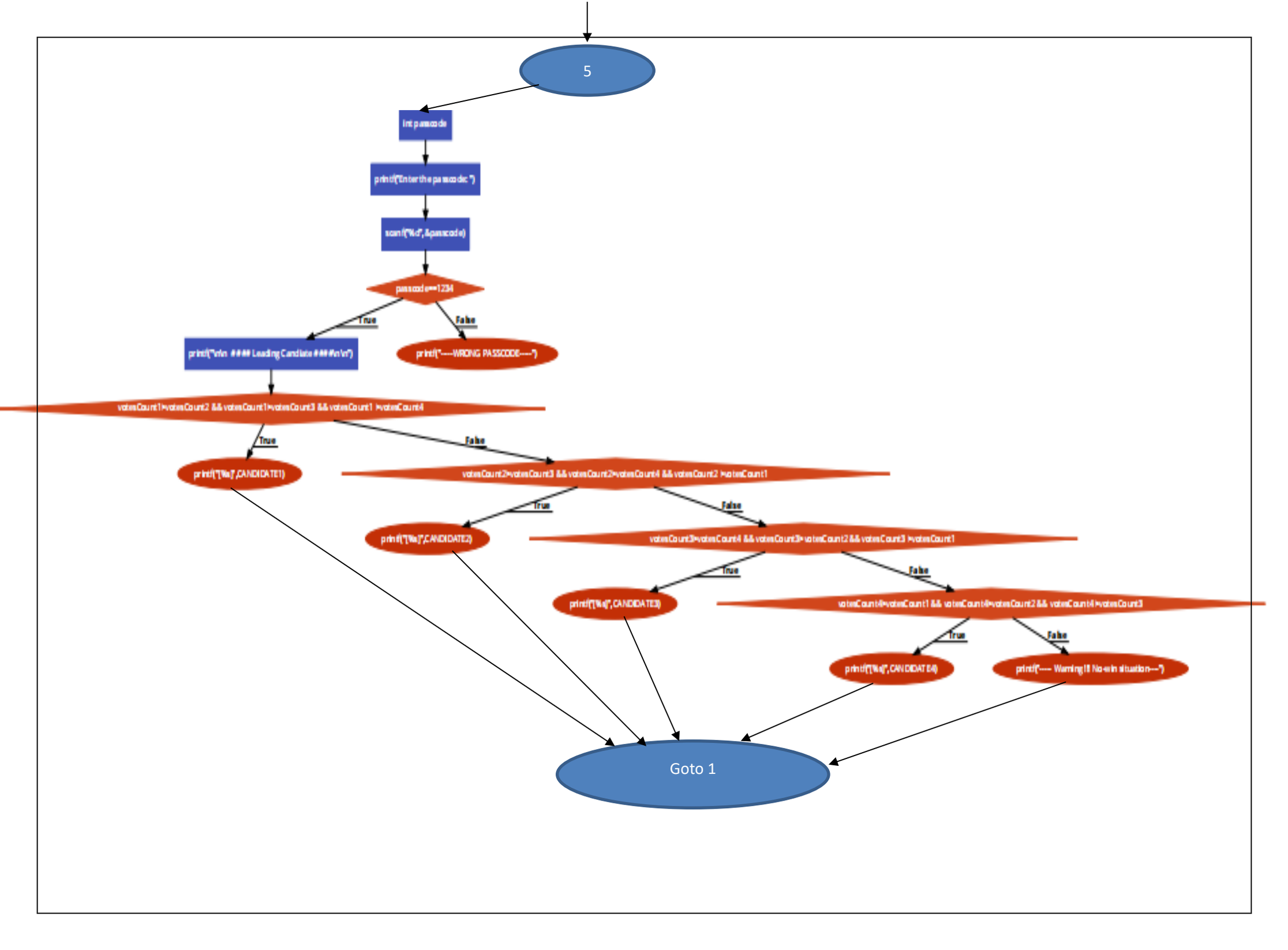
flow chart




char a[100][20]=
int i
char reg
20
int in=-1,pos
printf("Enter the Register number (use capital letter): ")
scanf("%s",reg)
1
gets(reg);
i=0
false
i<47
true
in==-1
strcmp(a[i],reg)==1
True
pos=i
False
in=0
i++
printf(" The register number is not registered")
int choice
Goto 1
flag[pos] ==0
printf("You are already done")
Goto 1


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)
choice
4
5
1
2
3
votesCount4++
spoiledtvotes++
printf("\n Error. Wrong Choice !! Please retry")
votesCount1++
votesCount2++
hold the screen getch();
votesCount3++
i=1
flag[pos]=1
printf("----THANKS FOR THE VOTE----")
Goto 1


4
void votesCount()
int passcode
printf("Enter the passcode: ")
scanf("%d",&passcode)
passcode==1234
True
False
printf("\n\n ##### Voting Statics ####")
printf("wrong passcode")
Goto 1
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)
Goto 1


5
int passcode
printf("Enter the passcode:")
scanf("%d",&passcode)
passcode==1234
True
False
printf("\n\n ### Leading Candiate ###\n\n")
printf("----WRONG PASSCODE----")
voteCount1>voteCount2 && voteCount1>voteCount3 && voteCount1 >voteCount4
True
False
printf("[%s]",CANDIDATE1)
voteCount2>voteCount3 && voteCount2>voteCount4 && voteCount2 >voteCount1
True
False
printf("[%s]",CANDIDATE2)
voteCount3>voteCount4 && voteCount3>voteCount2 && voteCount3 >voteCount1
True
False
printf("[%s]",CANDIDATE3)
voteCount4>voteCount1 && voteCount4>voteCount2 && voteCount4 >voteCount3
True
False
printf("[%s]",CANDIDATE4)
printf("---- Warning!! No-win situation----")
Goto 1

2.2: ALGORITHM

Step 1 :Start
Step 2: Set flag []==0
Step 3 : print
1-caste the vote
2- find vote count
3- find leading candidate
Step 4 : read choice
Step 5 : if choice ==1, enter registration number and assign to reg
Step 5:1 :if condition is ture(strcmp(a[i],reg)==1) , then
Step 5:2:if fix flag[pos]==0 ,print list of candidates.
Step 5:2:1 : read the choice and add to the votecount
Step 5:2:2 :else print “you are already done”
Step 5:3: change flag[pos]==1 for the corresponding reg.no
Step:5:4: print thanks for vote
Step 5:5 :refresh the screen
Step 6 : Else choice==2,Then print “enter the passcode”
Step 7 : if the passcode is correct ,print the votecount of each candidate and spoiled votes
Step 7:1 :else print wrong passcode
Step 7:2 :refresh the screen
Step 8 :else choice==3,then print “enter the passcode “
Step 9 :if the passcode is correct , print the leading candidate
Step 9:1 :else print “wrong passcode”
Step 9:2:refresh the screen
Step 10 :else exit .

2.3: SOURCE CODE

```
#include<stdio.h>
#include<conio.h>
#define CANDIDATE_COUNT

#define CANDIDATE1 "Farzin "
#define CANDIDATE2 "Harris"
#define CANDIDATE3 "Adwin"
#define CANDIDATE4 "Sooraj"

int votesCount1=0, votesCount2=0, votesCount3=0, votesCount4=0, spoiledtvotes=0;
void castVote();
void votesCount();
void getLeadingCandidate();
int flag[100]={0};
int main()
{
clrscr();
int choice;
do{
printf("\n\n ###### Welcome to Election/Voting 2021 #####");
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 Error: Invalid Choice");
}
}while(choice!=0);
```

```
//hold the screen
getch();
return 0;
}
void castVote()
{
   char
a[100][20]={"THAUBVD001","THAUBVD002","THAUBVD003","THAUBVD004","THAUBVD005","THAUBVD006",
"THAUBVD007","THAUBVD008","THAUBVD009","THAUBVD010","THAUBVD011","THAUBVD012","THAUBVD
013","THAUBVD014","THAUBVD015","THAUBVD016","THAUBVD017","THAUBVD018","THAUBVD019","THAU
BVD0020","THAUBVD021","THAUBVD022","THAUBVD023","THAUBVD024","THAUBVD025","THAUBVD026","
THAUBVD027","THAUBVD028","THAUBVD029","THAUBVD031","THAUBVD032","THAUBVD033","THAUBVD0
34","THAUBVD035","THAUBVD036","THAUBVD037","THAUBVD038","THAUBVD039","THAUBVD040","THAU
BVD041","THAUBVD042","THAUBVD043","THAUBVD044","THAUBVD045","THAUBVD046","THAUBVD047"};
   int i;
   char reg[20];
   int in=-1,pos;
printf("Enter the Register number (use capital letter): ");
   scanf("%s",reg); //gets(reg);
   for(i=0;i<47;i++)
   {
      if(strcmp(a[i],reg)==1){
         pos=i;
         in=0;
      }
   }
   if(in==-1)
{   printf("The register number is not registered");




   }
   else
   {
      int choice;
      if(flag[pos]==0)
      {
printf("\n\n ### Please choose your Candidate ####\n\n");
printf("\n 1. %s", CANDIDATE1);
```

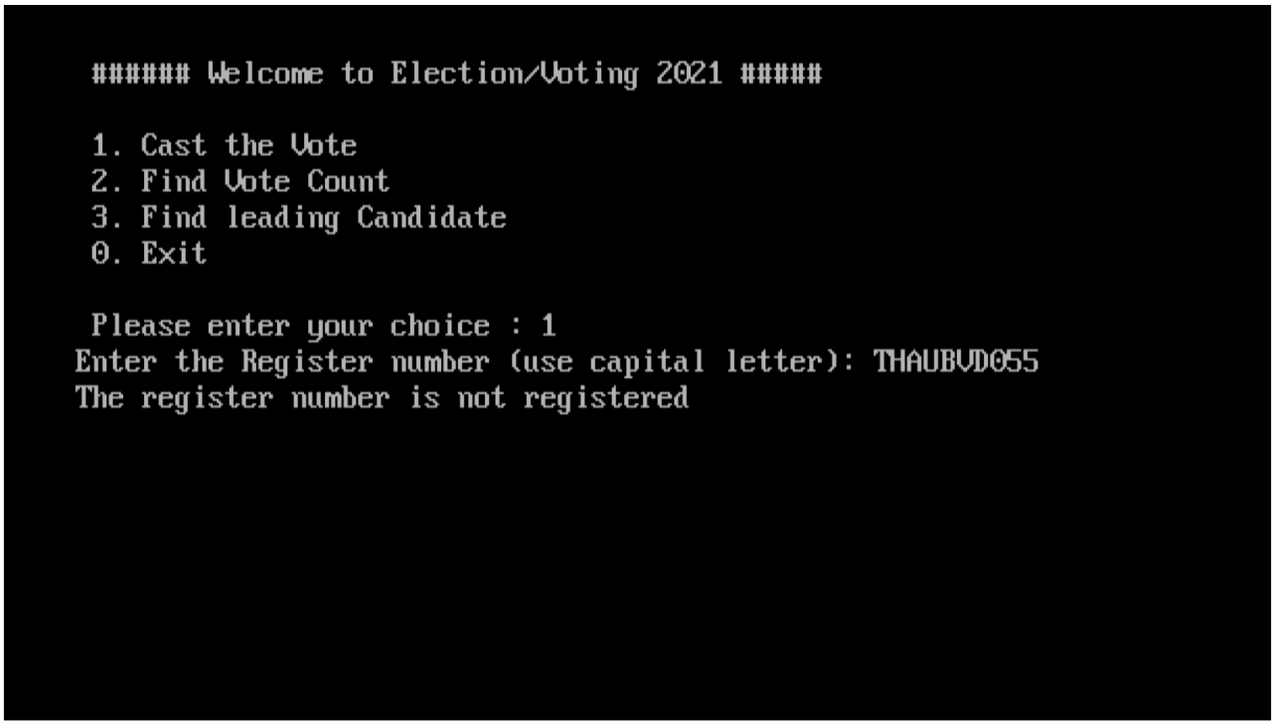
```
printf("\n 2. %s", CANDIDATE2);
printf("\n 3. %s", CANDIDATE3);
printf("\n 4. %s", CANDIDATE4);
printf("\n 5. %s", "None of These");

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");
        }
      //hold the screen
//getch();
        in=-1;
        flag[pos]=1;
printf("------THANKS FOR THE VOTE------");
      }

else{
printf("You are already done");



   }
  }
getch();

}
void votesCount(){int passcode;
printf("Enter the passcode: ");
   scanf("%d",&passcode);
   if(passcode==1234){
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);}
else{
printf("wrong passcode");
}
}

void getLeadingCandidate(){int passcode;
printf("Enter the passcode: ");
   scanf("%d",&passcode);
   if(passcode==1234){
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----"); }
   else
   {printf("-----WRONG PASSCODE-----");}
getch();
}
```

2.4: <u>OUTPUT</u>

1) To cast the vote:-

1) If the entered register number is WRONG:-



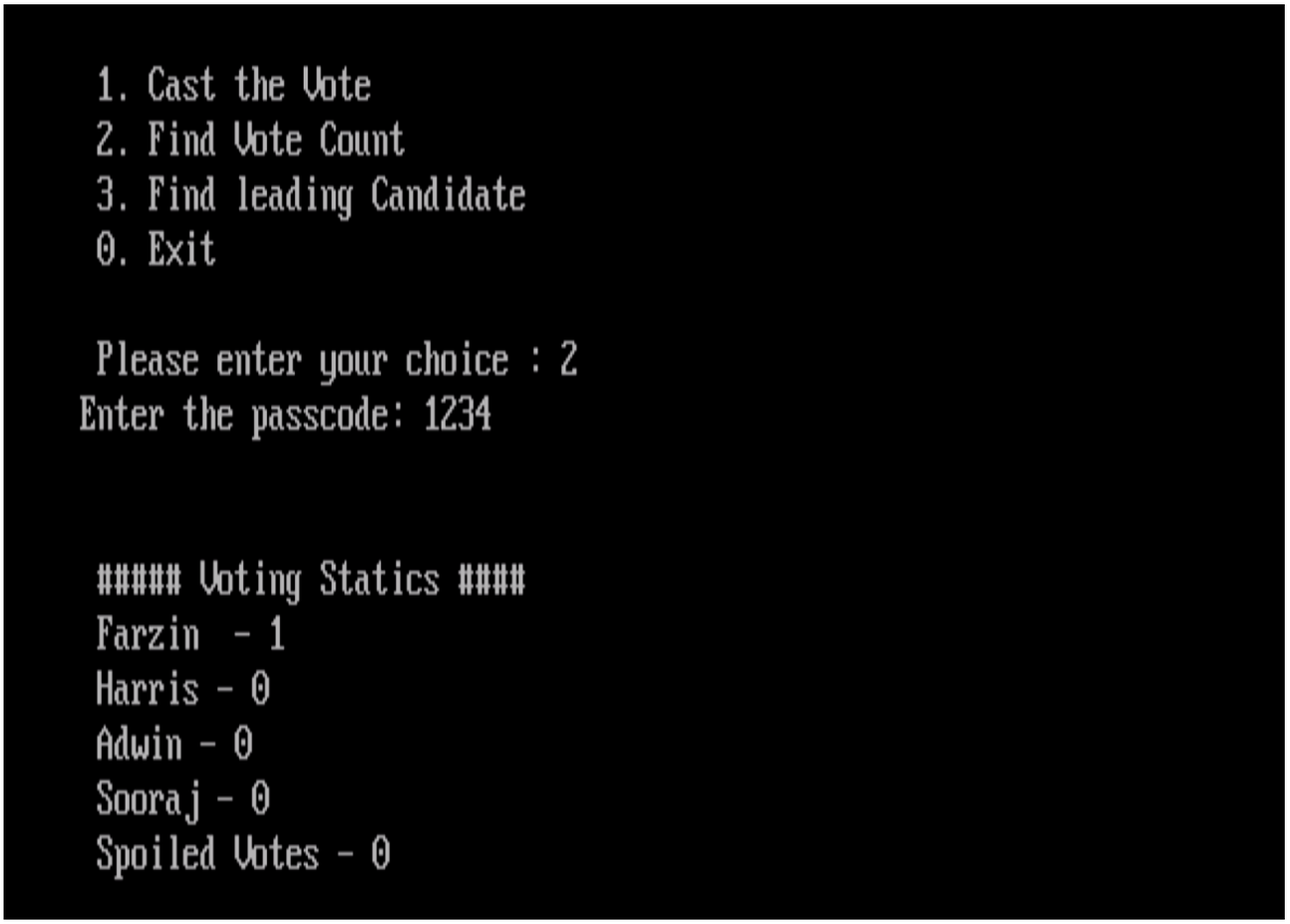
2) If entered register number is CORRECT:-

```
 ###### Welcome to Election/Voting 2021 #####

 1. Cast the Vote
 2. Find Vote Count
 3. Find leading Candidate
 0. Exit

 Please enter your choice : 1
Enter the Register number (use capital letter): THAUBVD003


 ### Please choose your Candidate ####


 1. Farzin
 2. Harris
 3. Adwin
 4. Sooraj
 5. None of These

 Input your choice (1 - 4) : 1
------THANKS FOR THE VOTE------
```

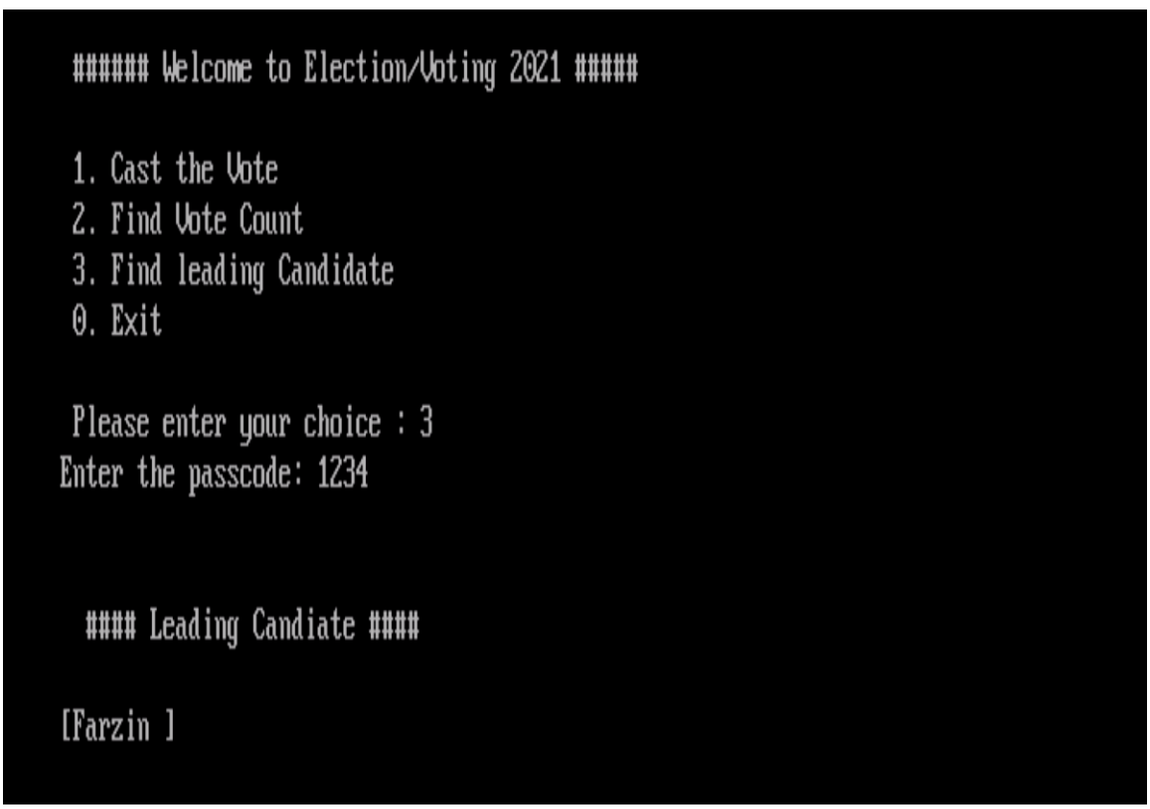
4) If the register number is already voted:-

```
Enter the Register number (use capital letter): THAUBVD003


 ### Please choose your Candidate ####


 1. Farzin
 2. Harris
 3. Adwin
 4. Sooraj
 5. None of These

 Input your choice (1 - 4) : 1
------THANKS FOR THE VOTE------

 ####### Welcome to Election/Voting 2021 #####

 1. Cast the Vote
 2. Find Vote Count
 3. Find leading Candidate
 0. Exit

 Please enter your choice : 1
Enter the Register number (use capital letter): THAUBVD003
You are already done
```

5) To find the vote (only for responsible authority):-



5)To find the leading candidate(only for responsible authority):-



CHAPTER 3

CONCLUSION

By doing this project we were able to bring a new votingsystem.With the advent of technology and Internet in our day to day life, we were able to offer advanced voting system to voters in our college both from the native district and outside through our Online voting system.

- Efficient and Cost Effective
- Intelligent
- Save Time and Resources
- Easy and convenient
- Efficient data storage.
- Accuracy, real-time response and user friendliness.
- Intelligent Management