***“TIC TAC TOE”***

A MINI- PROJECT REPORT ON

Submitted in partial fulfillment of the requirements

For the degree of

Bachelor of Engineering

In

Information Technology

by

**Name of the Student-1:SHIVANI D.WAKDE Roll No.-18IT1058**

**Name of the Student-2: SHRUSHTI S. POLEKAR Roll No.-18IT2021**

**Name of the Student-3:MAYURESH R. PANDARE Roll No.-19IT5011**

Supervisor

**Nilima Dongre**



Department of Information Technology

Dr. D. Y. Patil Group’s

**Ramrao Adik Institute of Technology**

Dr. D. Y. Patil Vidyanagar, Sector 7, Nerul, Navi Mumbai 400706.

(Affiliated to University of Mumbai)

( 2020)

# rait logo

Ramrao Adik Institute of Technology

(Affiliated to the University of Mumbai)

Dr. D. Y. Patil Vidyanagar,Sector 7, Nerul, Navi Mumbai 400706.

CERTIFICATE

This is to certify that,Mini Project entitled

“TIC TAC TOE USING BASH SCRIPTING”

is a bonafide work done by

Student Names

1. SHIVANI WAKDE

2. SHRUSHTI POLEKAR

3.MAYURESH PANDARE

and is submitted in the partial fulfillment of the requirement for the

degree of

Bachelorof Engineering

in

Information Technology

to the

University of Mumbai

Supervisor

Prof.Nilima M.Dongre

Project Guide Head of the department

Nilima Dongre Dr. Ashish Jadhav

Certificate of Approval by Examiners

This Mini Project report entitled “ Tic Tac Toe ” is a bonafide work done by Student Names under the supervision of Prof.Nilima Dongre approved for the award of Bacheor Degree in Information Technology, University of Mumbai.

Examiners :

1. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

2. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

Supervisors :

1. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

2. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

Principal :

. . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

Date :

Place :

# **DECLARATION**

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

**Name and Roll No. of Students Signature**

1. SHIVANI D.WAKDE

2. SHRUSHTI S. POLEKAR

3.MAYURESH R. PANDARE

Date:

Place:

# ACKNOWLEDGEMENT

The project “Tic Tac Toe” is creative work of many minds. A proper synchronization between individual is must for any project to be completed successfully. One cannot imagine the power of the force that guides us all and neither can we succeed without acknowledging it.

We take this opportunity to express my profound gratitude and deep regards to our Guide **Nilima Dongre**, Department of the Information Technology Engineering for her or her exemplary guidance, monitoring and constant encouragement throughout the completion of this mini project.

We would like to express our gratitude to **Dr. Ashish Jadhav,** Head of the department, Information Technology Engineering for encouraging and inspiring us to carry out the project in the department lab. We take this privilege to express my sincere thanksare thankful to **Dr. Mukesh D. Patil, Principal RAIT,** for his constant support and motivation.

We also would like to thank all the staff members Department of the Information Technology Engineering for providing us with the required facilities and support towards the completion of the project.

Last but not the least we are thankful to our parents and friends for their constant Inspiration, encouragement and well wishes by which we have made a challenging project.

STUDENT-SHIVANI D. WAKDE(**18IT1058**) Signature

# **PREFACE**

We take great opportunity to present this Mini Project report on “**TIC TAC TOE”** and put before readers some useful information regarding our project.

We have made sincere attempts and taken every care to present this matter in precise and compact form, the language being as simple as possible. We are sure that the information contained in this volume certainly prove useful for better insight in the scope and dimension of this project in it true perspective.

The task of the completion of the project though being difficult was made quite simple, interesting and successful due to deep involvement and complete dedication of our group members.

**TABLE OF CONTENTS**

**Declaration …………….…….………………………………………… I**

**Acknowledgement ……...……………………………………………… II**

**Preface …………….…………………………………………………… III**

**Table of Contents ……………………………………………………… IV**

**Table of figures ...…………………………………………………….… V**

**Abstract…….....…………………………………………………..…….. VI**

**TABLE OF CONTENTS**

**Sr. No. Topic Page No.**

1. **INTRODUCTION………………………………………………………………..8**
   1. **INTRODUCTION TO SCRIPTING LANGUAGES………………………10**
   2. **WHY PARTICULAR SCRIPTING LANGUAGE………………………...11**
   3. **PROBLEM STATEMENT………………………………………………….12**
   4. **OBJECTIVES………………………………………………………………..13**
2. **LITERATURE SURVEY……………………………………………………….14**
   1. **MOTIVATION………………………………………………………………15**
3. **PROPOSED SYSTEM…………………………………………………………..16**
   1. **HARDWARE AND SOFTWARE REQUIREMENTS……………………16**
4. **IMPLEMENTATION……………………………………………………………21**
   1. **SYSTEM BLOCK DIAGRAM……………….……………………………...21**
   2. **MODULE DESCRIPTION………………………………………………...22**
   3. **CODE………………………………………………………………………….26**
5. **RESULT…………………………………………………………………………...36**
   * 1. **OUTPUT SNAPSHOTS**
6. **CONCLUSION AND FUTURE SCOPE………………………………………..37**
   1. **CONCLUSION……………………………………………………………….37**
   2. **FUTURE SCOPE…………………………………………………………….37**
   3. **REFERENCES……………………………………………………………… 37**

**ABSTRACT**

Our project name is Tic-Tac-Toe game. But the question that comes to our mind is what is exactly Tic Tac Toe?

* WHAT IS TIC TAC TOE?

This game is very popular and is fairly simple by itself. It is actually a two player game. In this game, there is a board with n x n squares. In our game, it is 3 x 3 squares. The goal of Tic-Tac-Toe is to be one of the players to get three same

symbols in a row - horizontally, vertically or diagonally - on a 3 x 3 grid .Tic-tac-toe is not a very challenging game for human beings. If you’re an enthusiast ,you’ve probably moved from the basic  game to some variant like three dimensional tic-tac-toe on a larger grid. If you sit down right now to play ordinary three-by-three tic-tac-toe with a friend, what will probably happen is that every game will come out a tie. Both you and your friend can probably

play perfectly, never making a mistake that would allow your opponent to win. But can you describe how you know where to move each turn? Most of the time ,you probably aren’t even aware of alternative possibilities; you just look at the board and instantly know where you want to move. That kind of instant knowledge is great for human beings, because it makes you a fast player. But it isn’t much help in writing a computer program. For that, you have to know very explicitly what your strategy is.

**CHAPTER -1**

**INTRODUCTION**

**INTRODUCTION**

* 1. **INTRODUCTION TO SCRIPTING LANGUAGES**

Usually shells are interactive that mean, they accept command as input from

users and execute them. However some time we want to execute a bunch of

commands routinely, so we have type in all commands each time in terminal.

Shell scripts are similar to the  batch file  in MS-DOS. Each shell script is saved

with  .sh file extension eg . myscript.sh A shell script have syntax just like any other programming language. If you have any prior experience with any programming language like Python, C/C++ etc. it would be very easy to get started with it.

* 1. **WHY PARTICULAR SCRIPTING LANGUAGE**
* There are many reasons to write shell scripts –
* To avoid repetitive work and automation
* System admins use shell scripting for routine backups
* System monitoring
* Adding new functionality to the shell etc.
  1. **PROBLEM STATEMENT**
* Prone to costly errors, a single mistake can change the command which

might be harmful

* Slow execution speed
* Design flaws within the language syntax or implementation
* Not well suited for large and complex task
  1. **OBJECTIVES**
* The command and syntax are exactly the same as those directly entered in
* command line, so programmer do not need to switch to entirely different
* syntax: Writing shell scripts are much quicker
* Quick start

**CHAPTER -2**

**LITERATURE SURVEY**

**2.1** **MOTIVATION**

A player can play a  perfect game  of tic-tac-toe (to win or, at least, draw) if each time it is his turn to play he chooses the first available move from the following list, as used in Newell and Simon&#39;s 1972 tic-tac-toe program. [15]

1. Win: If the player has two in a row, they can place a third to get three in a row.

2. Block: If the opponent has two in a row, the player must play the third themselves to block the opponent.

3. Fork: Create an opportunity where the player has two threats to win (two non-blocked lines of 2).

4. Blocking an opponent&#39;s fork: If there is only one possible fork for the opponent, the player should block it. Otherwise, the player should block any forks in any way that simultaneously allows them to create two in a row. Otherwise, the player should create a two in a row to force the opponent into defending, as long as it doesnt result in them creating a fork.

5. Center: A player marks the center. (If it is the first move of the game, playing on a corner gives the second player more opportunities to make a mistake and may therefore be the better choice; however, it makes no difference between perfect players.)

6. Opposite corner: If the opponent is in the corner, the player plays the

opposite corner.

7. Empty corner: The player plays in a corner square.

8. Empty side: The player plays in a middle square on any of the 4 sides.

**CHAPTER -3**

**PROPOSED SYSTEM**

**3.1 HARDWARE AND SOFTWARE REQUIREMENTS**

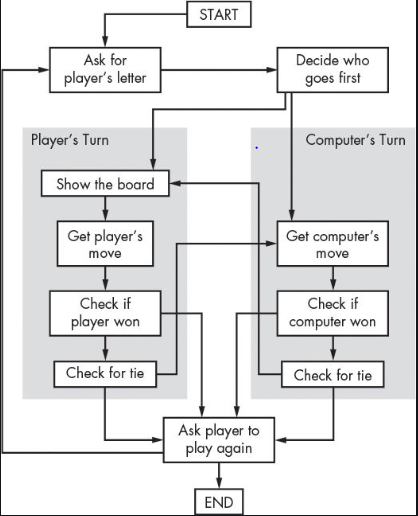
* Shell can be accessed by user using a command line interface. A special program called Terminal in linux/macOS or Command Prompt in Windows OS is provided to type in the human readable commands such as “cat”, “ls” etc. and then it is being execute. The result is then displayed on the terminal to the user. It will list all the files in current working directory in long listing format. Working with command line shell is bit difficult for the beginners because it’s hard to memorize so many commands. It is very powerful, it allows user to store commands in a file and execute them together. This way any repetitive task can be easily automated. These files are usually called batch files in Windows and Shell Scripts in Linux/macOS systems
* Graphical shells provide means for manipulating programs based on graphical user interface (GUI), by allowing for operations such as opening, closing, moving and resizing windows, as well as switching focus between windows. Window OS or Ubuntu OS can be considered as good example which provide GUI to user for interacting with program. User do not need to type in command for every actions. There are several shells are available for Linux systems like – BASH (Bourne Again SHell)  – It is most widely used shell in Linux systems. It is used as default login shell in Linux systems and in macOS. It can also be installed on Windows OS. CSH (C SHell)  – The C shell’s syntax and usage are very similar to the C programming language. KSH (Korn SHell)  – The Korn Shell also was the base for the POSIX Shell standard specifications etc. Each shell does the same job but understand different commands and provide different built in functions.

.

**CHAPTER –4**

**IMPLEMENTATION**

**4.1 SYSTEM BLOCK DIAGRAM**



**4.2 MODULE DESCRIPTION**

If any player is able to draw three X s or three O s in the following combinations then that player wins. The combinations are:

a)  0,1,2

b) 3,4,5

c) 6,7,8

d) 0,4,8

e) 2,4,6

f) 0,3,6

g) 1,4,7

h) 2,5,8

A player can play a  perfect game  of tic-tac-toe (to win or, at least, draw) if each time it is his turn to play he chooses the first available move from the following list, as used in Newell and Simon&#39;s 1972 tic-tac-toe program.

1. Win: If the player has two in a row, they can place a third to get three in a row.

2. Block: If the opponent has two in a row, the player must play the third themselves to block the opponent.

3. Fork: Create an opportunity where the player has two threats to win (two non-blocked lines of 2).

4. Blocking an opponent fork: If there is only one possible fork for the opponent, the player should block it. Otherwise, the player should block any forks in any way that simultaneously allows them to create two in a row. Otherwise, the player should create a two in a row to force the opponent into defending, as long as it doesnt result in them creating a fork.

5. Center: A player marks the center. (If it is the first move of the game, playing on a corner gives the second player more opportunities to make mistake and may therefore be the better choice; however, it makes no difference between perfect players.)

6. Opposite corner: If the opponent is in the corner, the player plays the opposite corner. 7. Empty corner: The player plays in a corner square.8. Empty side: The player plays in a middle square on any of the 4 sides.

**4.3** **CODE**

#! /bin/bash

reset(){

Arr=('.' '.' '.' '.' '.' '.' '.' '.' '.')

player=1

gamestatus=1

echo "------------WELCOME------------ "

echo "Let's start the game"

}

set(){

idx=$(( $1 \* 3 + $2 ))

if [ ${Arr[$idx]} == "." ]; then

Arr[$idx]=$3

player=$((player%2+1))

else

echo "You can't place there!"

fi

}

print(){

echo "r\c 0 1 2"

echo "0 ${Arr[0]} ${Arr[1]} ${Arr[2]}"

echo "1 ${Arr[3]} ${Arr[4]} ${Arr[5]}"

echo "2 ${Arr[6]} ${Arr[7]} ${Arr[8]}"

}

checkmatch(){

if [ ${Arr[$1]} != "." ] && [ ${Arr[$1]} == ${Arr[$2]} ] && [ ${Arr[$2]} == ${Arr[$3]} ]; then

gamestatus=0

fi

}

checkgame(){

checkmatch 0 1 2

checkmatch 3 4 5

checkmatch 6 7 8

checkmatch 0 3 6

checkmatch 1 4 7

checkmatch 2 5 8

checkmatch 0 4 8

checkmatch 2 4 6

}

reset

# infinite game loop

var=0

while [ 1 == 1 ] && [ $var -lt 9 ]; do

echo ""

if [ $player == 1 ]; then sym=O

else sym=X;

fi

echo "Player $player's turn: ($sym)"

print

echo ""

echo " Command:"

echo " 1. set [row] [column]"

echo " 2. restart"

while [ 1 == 1 ]; do

read -r cmd a b

if [ $cmd == "set" ]; then

set $a $b $sym

var=$(($var+1))

break

elif [ $cmd == "restart" ]; then

var=0

reset

break

else

echo "Wrong input"

fi

done

checkgame

if [ $gamestatus != 1 ]; then

echo "Gameover"

player=$((player%2+1))

echo "Player $player ($sym) win!!"

while [ 1 == 1 ]; do

read -r cmd n

if [ $cmd == "restart" ]; then

reset

break

fi

done

fi

done

if [ $var -ge 9 ]; then

echo "Sorry, your match is draw"

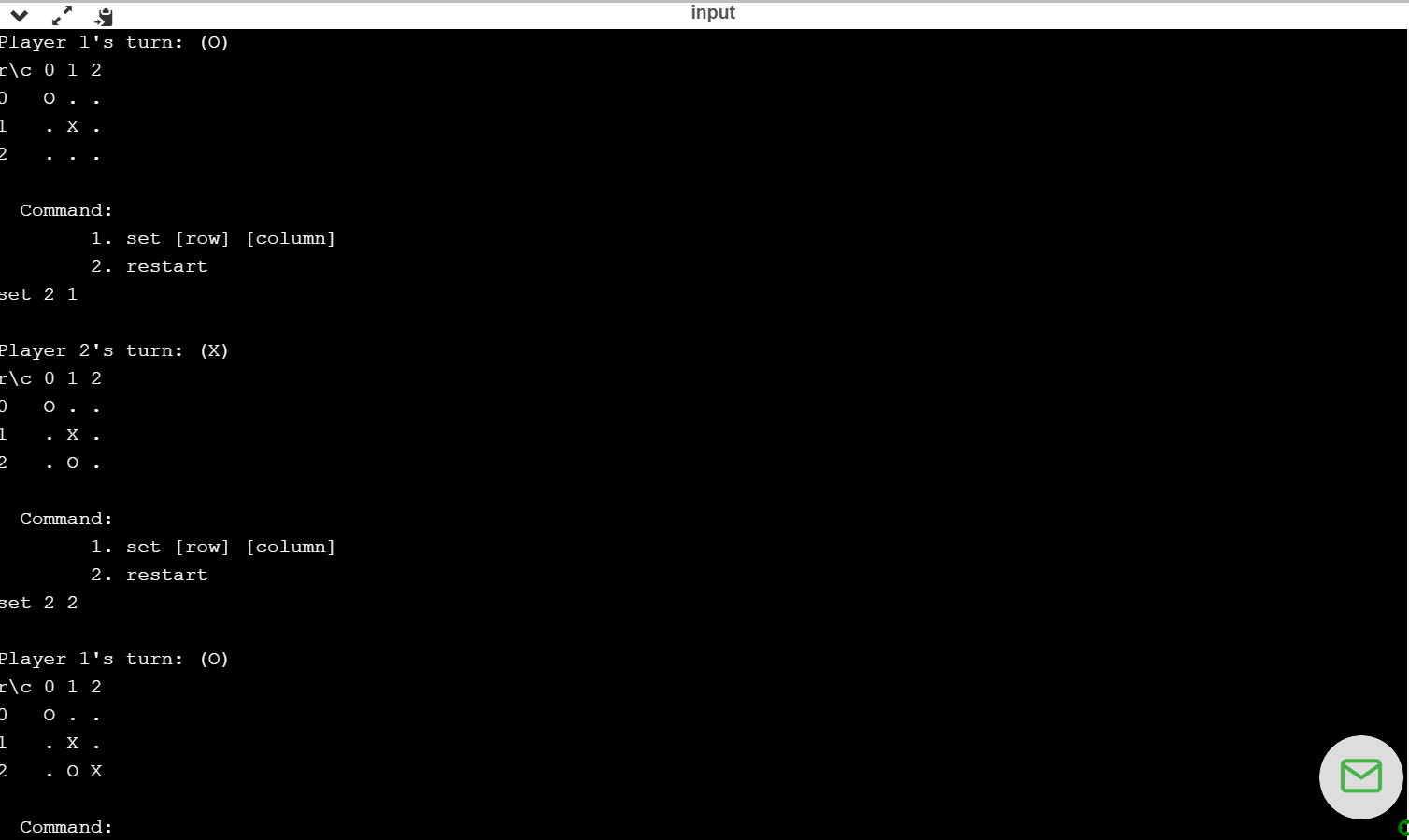
fi

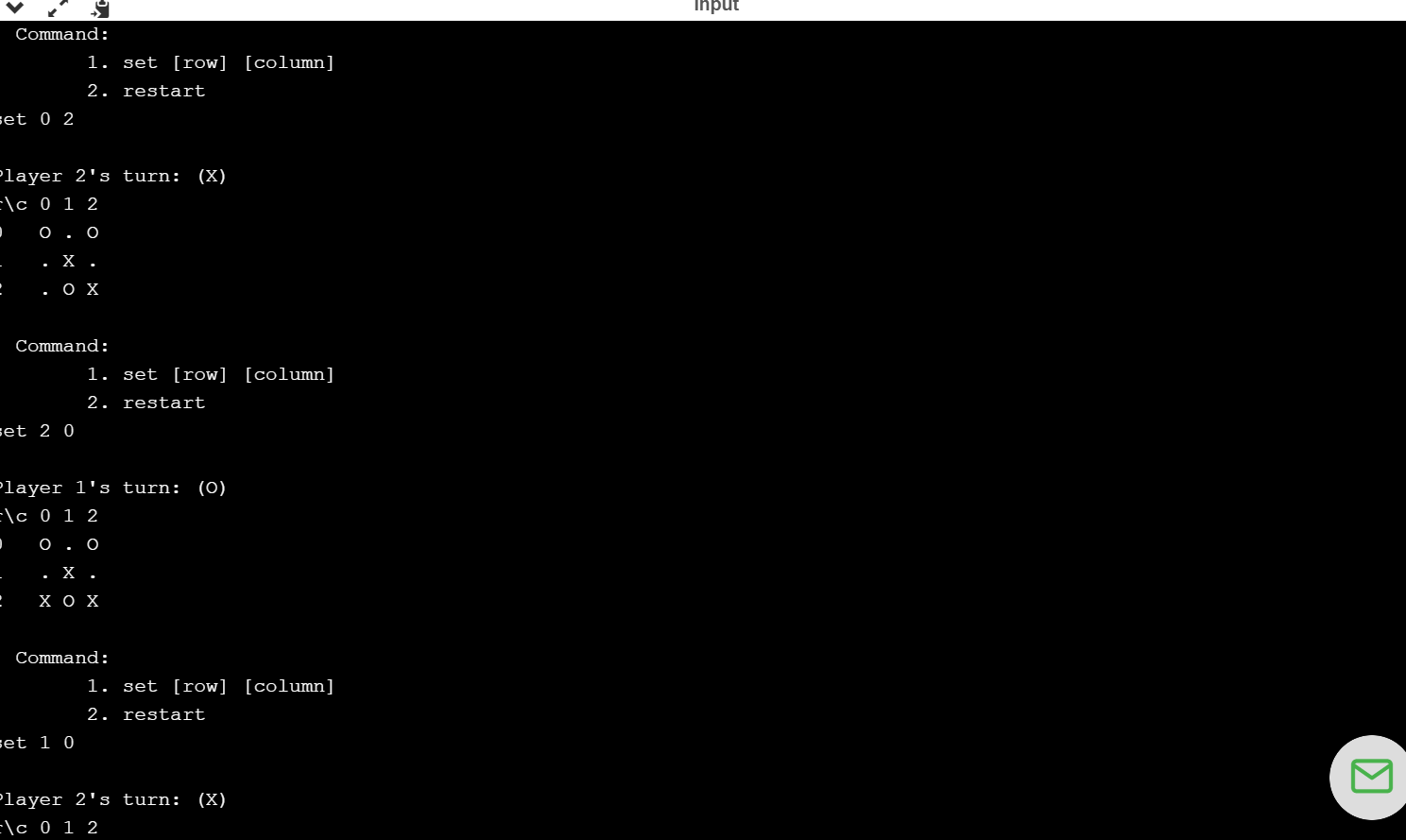
**CHAPTER –5**

**RESULT**

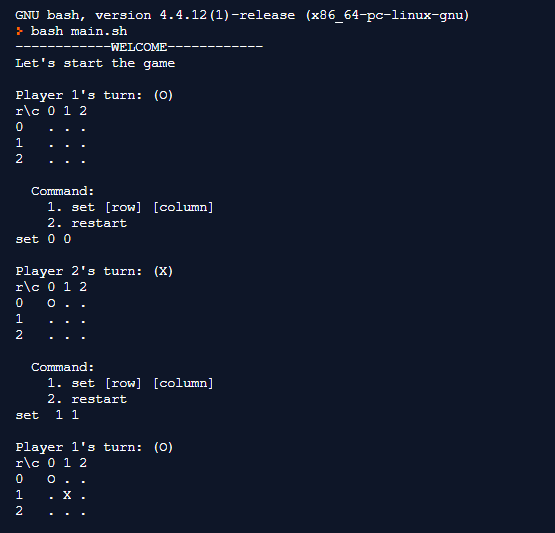
**5.1.1 OUTPUT SNAPSHOTS**

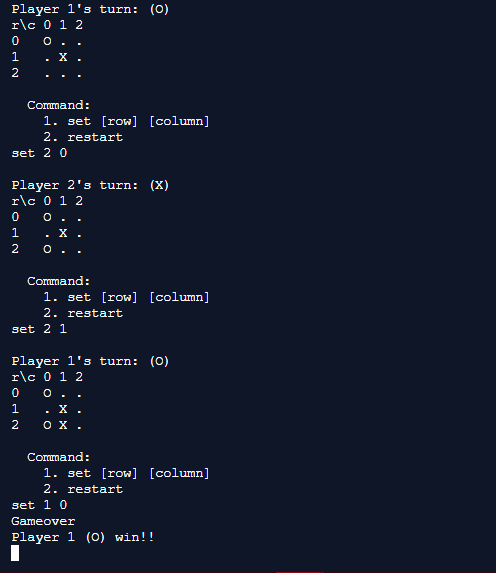


**–**









**CHAPTER –6**

**CONCLUSION AND FUTURE SCOPE**

**6.1 CONCLUSION**

In the end, we would like to conclude that our aim was to make tic tac toe game using bash script. Our code is written for two player version i.e Human vs human. The final outcome (win, lose, draw) was the

only information available regarding the quality of the play.

**6.2 FUTURE SCOPE**

The game is to be played between two people .One of the player chooses ‘O’ and the other ‘X’ to mark their respective cells. The game starts with one of the players and the game ends when

one of the players has one whole row/ column/ diagonal filled with his/her respective character (‘O’ or ‘X’).If no one wins, then the game is said to be draw.

**6.3 REFERENCES**

* [**www.google.com**](http://www.google.com)
* [**www.wikipedia.org**](http://www.wikipedia.org)
* [**www.tutorialpoint.com**](http://www.tutorialpoint.com)
* [**www.w3schools.com**](http://www.w3schools.com)
* [**www.youtube.com**](http://www.youtube.com)