

PROJECT REPORT

OF

(TIC TAC TOE)

INT213



COMPUTER SCIENCE ENGINEERING

SUBMITTED BY:

1. Name: Aryan Dubey
Reg.No: 11904244
Roll No: RK19KP25
2. Name: Sourabh Anand
Reg.No: 11916273
Roll No: RK19KP41
3. Name: P. Karthikeyan Reddy
Reg.No: 11904255
Roll No: RK19KP24

SUBMITTED TO:

Ankita Wadhawan

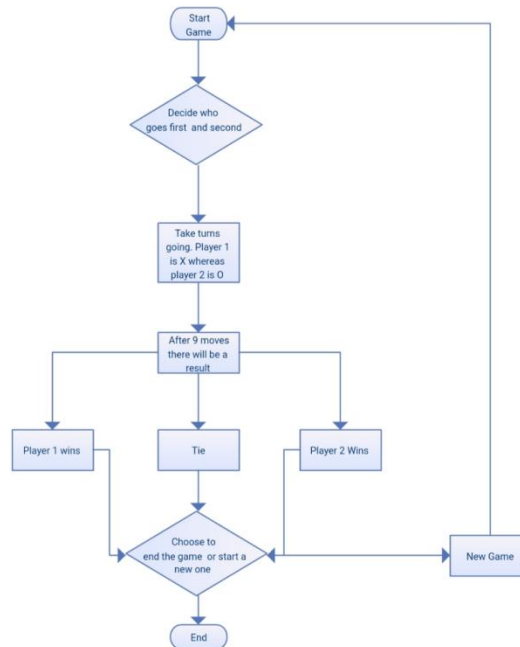
Introduction

In the existing system, most of the records are maintained on paper. It becomes very inconvenient to modify the idea. In the existing system, there is a possibility that the same data in different values which means the entries of the same data do not match. This inconsistent state does not supply the concrete information which poses a problem in the case information related to particular search record.

Our project is very useful. User is no longer required to check his register in search of records, as now it can be searched over the software by choosing some options. The user need not to type in most of the information. He/She is just required to enter the desired options. On the whole it liberates the user from keeping lengthy manual records. In a nutshell, it abates the work load of an organization.

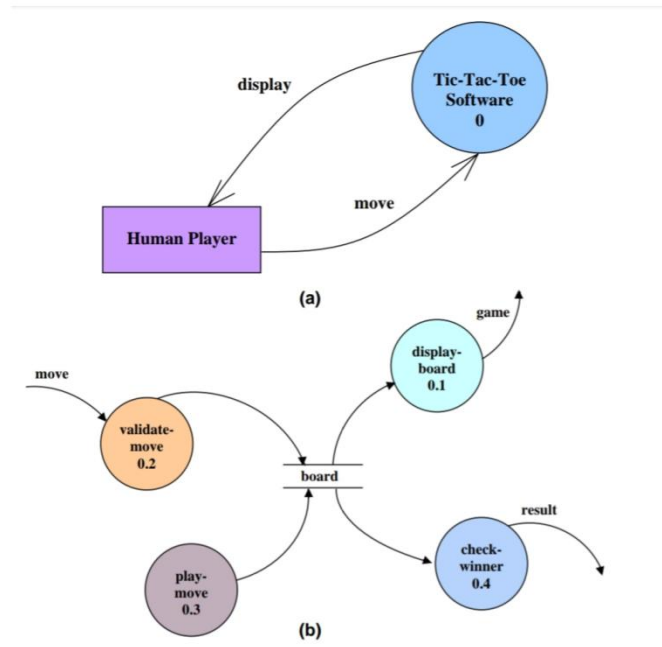
In today's world, no one likes to perform calculations on calculator or manually when computer is there. Every one wants his/her work to be done by computer automatically and displayed the result for further manipulations.

ER DIAGRAM OF A TIC TAC TOE GAME



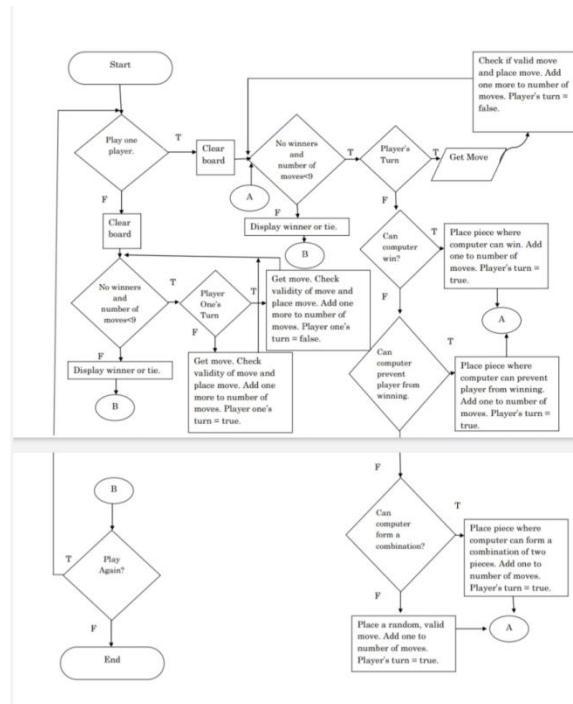
- Step-1 : We have to start the game.
- Step-2 : Now we have to decide who goes first and second.
- Step-3 : Now players will take turns, player 1 is "X", where as player 2 is "O".
- Step-4 : After 9 moves in the game there will be a result.
- Step-5 : Now there will be a three chances are; 1st chance : Player 1 wins, 2nd chance : Player 2 wins, 3rd chance : Other wise the game will be tie.
- Step-6 : Now the players can choose to end the game or start a new game.
- Step-7 : Now the players can press the end button to end the game or they can press the new game button to start a new game.

DFD of Tic Tac Toe Game (Data Flow Diagram)



Tic tac toe is both computer and human game in which a human and the computer makes alternative moves and as well as human and human can make alternative moves on a 3*3 square. A move consists of marking previously unmarked square. The player who first places three consecutive marks along a straight line on the square (i.e.along a row, column, or diagonal) wins the game. As soon as either the human player or the computer wins, a messege congratulating the winner should be displayed. If neither player manages on the board are filled up, then the game is drawn. The computer always tries to win a game.

FLOW CHART



METHODOLOGY

1. GameState

The Game state class contains all the information about the current state of a Tic Tac Toe game. The squares array contains a character value for each square in the Tic Tac Toe board. An empty square is represented by a space character. The player array contains the Players, who each have their own specified symbol which is used to fill squares array. The player Turn integer is an index which corresponds to the player in the player array whose turn it is. The constructor creates an empty Tic Tac Toe board, both players, and initializes the playerTurn index to zero. The researBoard method returns the board to an empty state and resets the playerTurn to zero.

2. Player

The player class is applied to manage the player's information. The number of wins and symbol of the player should be got by the methods in the player class. The variables and methods in player class are described as follows:

Wins: integer variable, represents the number of wins;

Symbol: char variable, represents the symbol that is used in the board by the player;

Player (symbol: char): constructor, creates a new player, defines the new player's symbol and initializes the number of wins is equal to zero;

getWins(): obtains the number of wins;

addWins(): the number of wins is increased 1;

getSymbol(): obtains the symbol of the player.

Move

The Move class is applied to mark the location of the symbol by the row-column method. The variables and methods in Move class are described as follows:

Row: integer variable, represents the row number of the Tic Tac Toe board;

Col: integer variable, represents the column number of the Tic Tac Toe board,

Move: construct, according to the two integers r and c, this method sets the location of the symbol on Tic Tac Toe board.

```
In [1]: from tkinter import *
import tkinter.messagebox
tk = Tk()
tk.title("Tic Tac Toe")

pa = StringVar()
playerb = StringVar()
p1 = StringVar()
p2 = StringVar()

player1_name = Entry(tk, textvariable=p1, bd=3)
player1_name.grid(row=1, column=1, columnspan=3)
player2_name = Entry(tk, textvariable=p2, bd=3)
player2_name.grid(row=2, column=1, columnspan=3)

bclick = True
flag = 0

def disableButton():
    button1.config(state=DISABLED)
    button2.config(state=DISABLED)
    button3.config(state=DISABLED)
    button4.config(state=DISABLED)
    button5.config(state=DISABLED)
    button6.config(state=DISABLED)
    button7.config(state=DISABLED)
    button8.config(state=DISABLED)
    button9.config(state=DISABLED)

def funClick(button):
    global bclick, flag, player2_name, player1_name, playerb, pa
    if button["text"] == " " and bclick == True:
        button["text"] = "X"
        bclick = False
```

Import thinter is the normal standard way of importing things. If you use that and you want to use the frame class from tkinter module, then you use variables is equals to tkinter. If we only need the frame class, we could use from tkinter import frame. **Tk.title("Tic Tac Toe")** it is used to write the title of the game.

Def disableButton() it is used to create buttons in the game like "X", "O"

[illegible]

Def checkForWin(): This is used for to know the winner of the game.

Player 1:

Player 2:

X	O	X
O	O	X
X	X	O

So this is the output of the game.

Result

A player can choose between two symbols with his opponent, usual games use “X” and “O”. If first player choose “X” then the second player have to play with “O” and vice versa.

A player marks any of the 3x3 squares with his symbol (may be “X” or “O”) and his aim is to create a straight line horizontally or vertically or diagonally with two intensions:

- a) Create a straight line before his opponent to win the game.
- b) Restrict his opponent from creating a straight line first.

In case logically no one can create a straight line with his own symbol, the game results a tie.

Hence there are only three possible results – a player wins, his opponent (human or computer) wins or it’s a tie.