Project 1

<Tic Tac Toe Game>

CSC-5 45276
Name: Choi, Jeannie
Date:7/16/2016

Introduction:

Title: Tic Tac Toe Game

Generally, Tic Tac Toe requires 2 players for the game to be played. In this program, the player will play against the computer. The 3X3 board involving 9 boxes will be displayed, and the numbers from 1 to 9 will be assigned to each box.
The player and the computer will be given a mark (X or O), and the player will be asked to type a number between 1 to 9 to mark a box. Then, the computer will randomly mark its move. The turn will be back to the player, and to the computer, and this will be repeated until one of them's mark occupies 3 consecutive boxes in any one of the rows, columns, or diagonally. The one who achieves this first is the winner, and then the game is over. The player may replay the game as many times as he/she wants, until he/she wants to quit. Every time the player plays, the number of wins, losses, and ties will be accumulated and outputted as a file.

Summary:

Project size: about 240 lines
The number of variables: about 23

In this project, I tried to program a simple game called "Tic Tac Toe," because this is the game I enjoyed to play when I was young. Also, almost everyone is familiar to this game and its rules are easy to be understood. Although this game is simple, it includes complex formulas and lots of programming languages and concepts.
I tried to use most of the concepts that are covered in the class, including menu, a variety types of data, system libraries, operators, formatting, as well as conditionals.
It took me about a week to complete this project. What I struggled the most while programming this game was applying different types of loops.

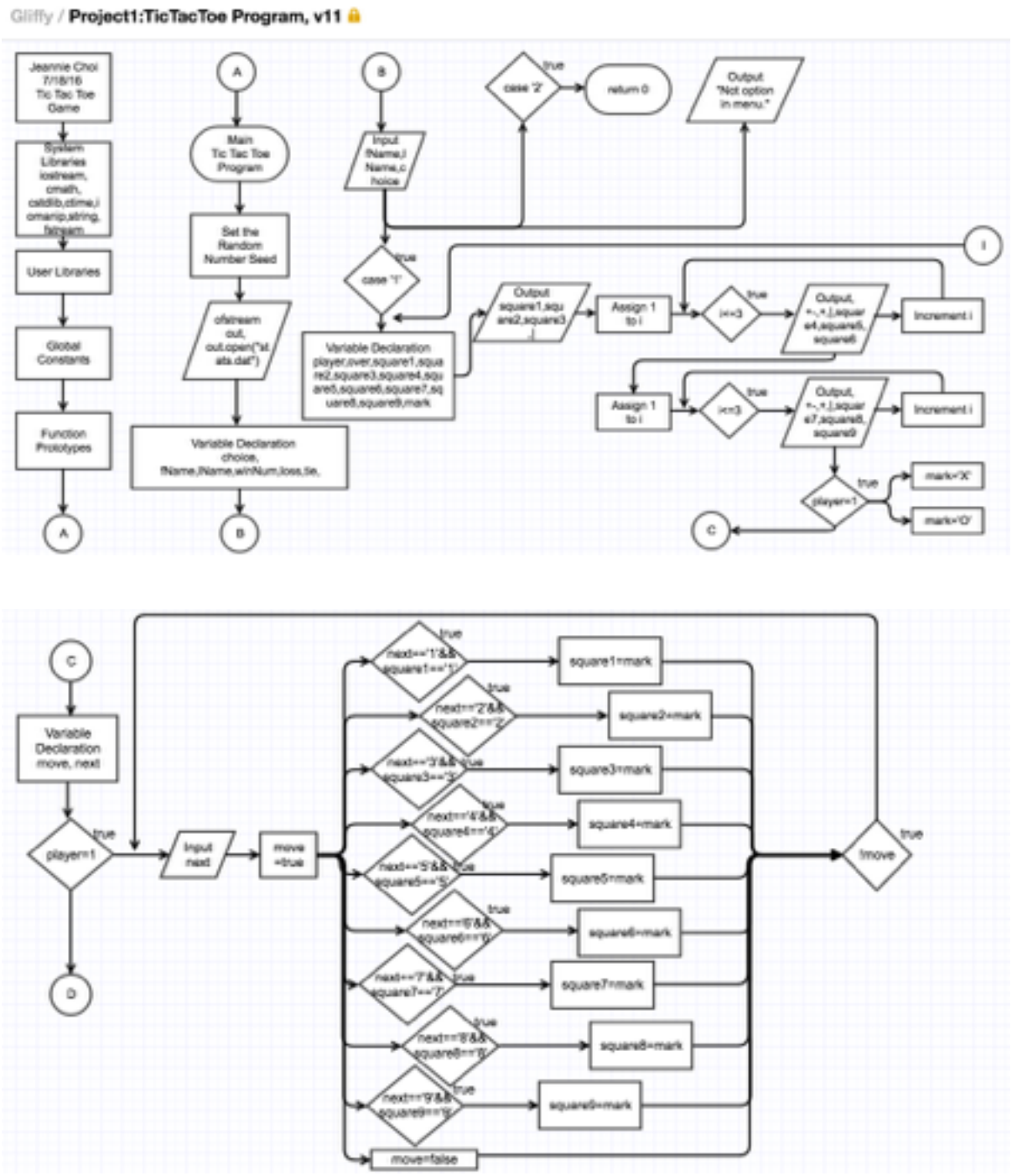
Research:

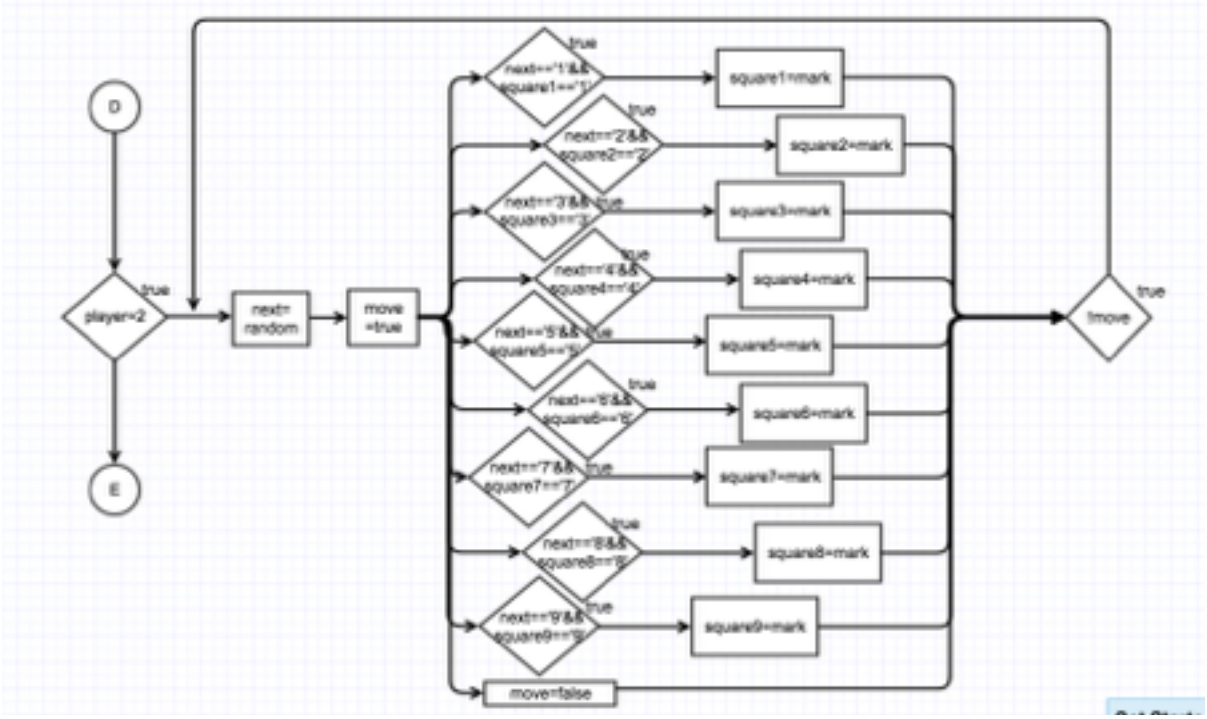
- ASCII Codes for random number from 1-9:
  Since I used character to assign numbers to each square box on the game board, I was required to use ASCII codes. To know what the ASCII codes are for the characters from 1-9, I referred to http://www.asciitable.com/
- Random number formula:
  In order to allow the computer to generate random numbers from 1-9, I used the textbook (Gaddis 8th Edition Chapter 3) as the reference:
  y=(rand()%(maxValue-minValue+1))+minValue
- Boolean expression:
  To understand the use of bool, I referred to the textbook (Chapter 2 and 4). The boolean expression, in this game, is used to determine if the game is over or not and if the move of the player or the computer is valid or not.

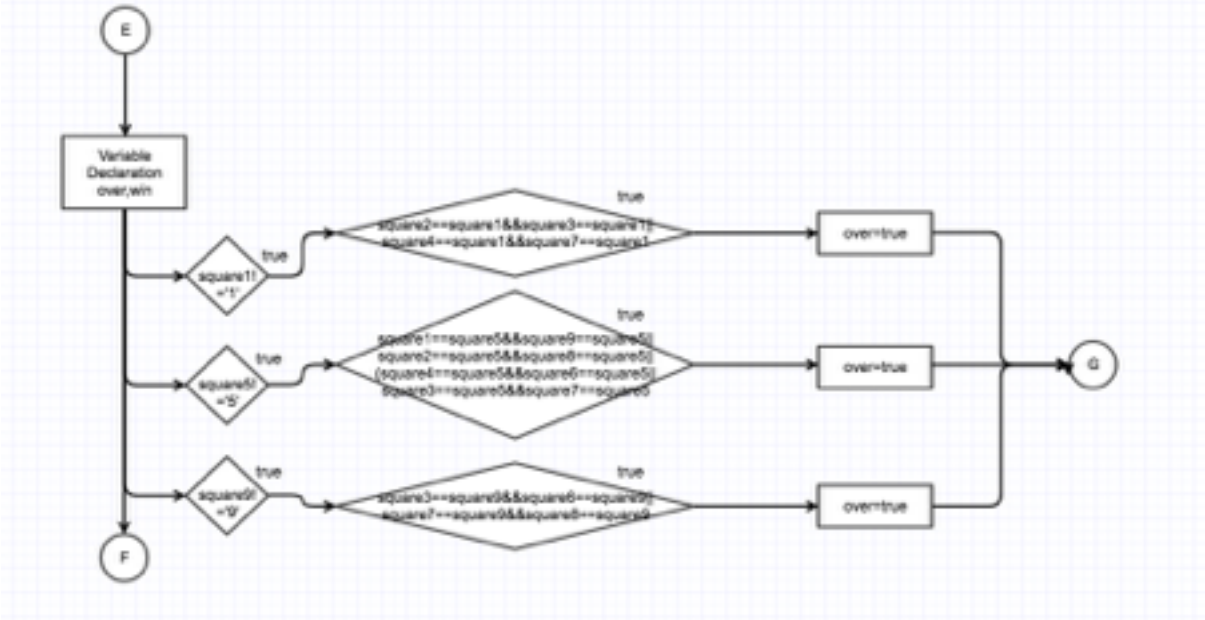
Description:

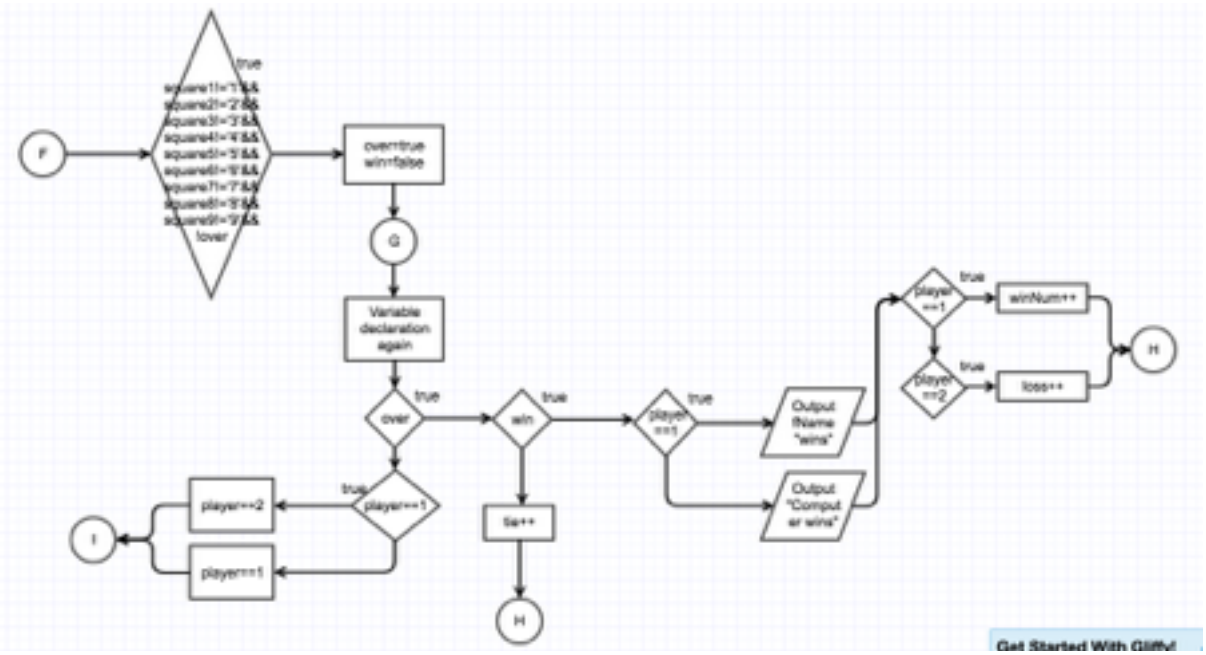
The main point of this program is the use of different types of loops for repetitive and alternating moves between the player and the computer.

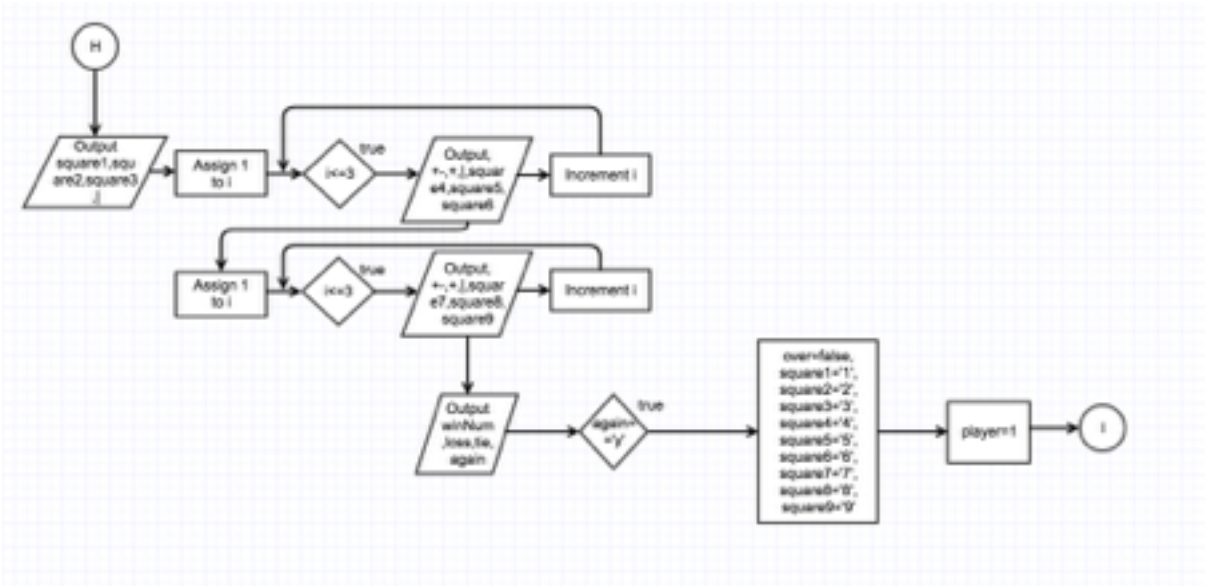
Flowchart:




D
player=2
true
next=
random
move
=true
next=='1'&&
square1==''
true
square1=mark
next=='2'&&
square2==''
true
square2=mark
next=='3'&&
square3==''
true
square3=mark
next=='4'&&
square4==''
true
square4=mark
next=='5'&&
square5==''
true
square5=mark
next=='6'&&
square6==''
true
square6=mark
next=='7'&&
square7==''
true
square7=mark
next=='8'&&
square8==''
true
square8=mark
next=='9'&&
square9==''
true
square9=mark
move=false
!move
true
E


E
Variable
Declaration
over,win
square1!
='1'
true
square2==square1&&square3==square1||
square4==square1&&square7==square1
true
over=true
square5!
='5'
true
square1==square5&&square9==square5||
square2==square5&&square8==square5||
(square4==square5&&square6==square5||
square3==square5&&square7==square5
true
over=true
square9!
='9'
true
square3==square9&&square6==square9||
square7==square9&&square8==square9
true
over=true
G
F


F
true
square1!='1'&&
square2!='2'&&
square3!='3'&&
square4!='4'&&
square5!='5'&&
square6!='6'&&
square7!='7'&&
square8!='8'&&
square9!='9'&&
!over
over=true
win=false
G
Variable declaration again
over
true
win
true
player==1
true
Output fName "wins"
Output "Computer wins"
player==1
true
winNum++
player==2
true
loss++
H
tie++
H
player==1
true
player=2
player=1
I


H
Output square1,square2,square3
Assign 1 to i
i<=3
true
Output, +-,+,|,square4,square5, square6
Increment i
Assign 1 to i
i<=3
true
Output, +-,+,|,square7,square8, square9
Increment i
Output winNum ,loss,tie, again
again== 'y'
true
over=false, square1='1', square2='2', square3='3', square4='4', square5='5', square6='6', square7='7', square8='8', square9='9'
player=1
I

Pseudo Code :

Set the random number seed
Input the name of the player
Input the player's decision, play or quit
If play,
set the game board
Assign mark to the player and the computer
If player's move,
check if the player's move is valid
If the move is valid,
mark the box
Else,
ask to try another move
If computer's move,
set random number
check if the computer's move is valid
If the move is valid,
mark the box
Else,
ask to try another move
Check the game over conditions
If any one of the rows, columns or diagonal lines is all occupied with either the players or the computer's mark only,

game is over.
If the player wins,
increment the number of wins
If the computer wins,
increment the number of losses
Else if the board is entirely filled without making any one line,
the game is over and increment the number of ties
If the game is over,
ask the player to play again
If yes,
the game is not over
set the new board for the new game
Else,
alternate player turns
If quit, return 0

Major Variables:

Type	Variable Name	Description	Location
Int	player	player or computer	if(player==1); else if(player==2);
	i	output the game board	for(int i=1;i<=3;i++);
	random	limiting range of random numbers between 1 and 9	random=(rand()%(57-49+1));
unsigned int	time(0)	random number at different time	srand(static_cast<unsigned int>(time(0)));
unsigned short	winNum	accumulated wins	else if(player==1)winNum++;
	loss	accumulated losses	else if(player==2)loss++;
	tie	accumulated ties	if(win==false)tie++;
string	fName	first name of the player	cin>>fName;
	lName	last name of the player	cin>>lName;
char	choice	option to play or quit	cin>>choice;
	square1('1')	box 1 on the game board	if(square1!='1'; if (next=='1'&&square1=='1') {square1=mark;}

Type	Variable Name	Description	Location

	square1('2')	box 2 on the game board	if(square2!='2';
			else
			if(next=='2'&&square2=='2')
			{square2=mark;}

	square1('3')	box 3 on the game board	if(square3!='3';
			else
			if(next=='3'&&square3=='3')
			{square3=mark;}

	square1('4')	box 4 on the game board	if(square4!='4;
			else
			if(next=='4'&&square4=='4')
			{square4=mark;}

	square1('5')	box 5 on the game board	if(square5!='5';
			else
			if(next=='5'&&square5=='5')
			{square5=mark;}

	square1('6')	box 6 on the game board	if(square6!='6';
			else
			if(next=='6'&&square6=='6')
			{square6=mark;}

	square1('7')	box 7 on the game board	if(square7!='7';
			else
			if(next=='7'&&square7=='7')
			{square7=mark;}

	square1('8')	box 8 on the game board	if(square8!='8';
			else
			if(next=='8'&&square8=='8')
			{square8=mark;}

	square1('9')	box 9 on the game board	if(square9!='9';
			else
			if(next=='9'&&square9=='9')
			{square9=mark;}

	mark	assigning player the mark	if(player==1){mark='X';} else{mark='O';}
	next	next moves of player and computer	cin>>next; next=random;
	again	player's decision to play again or not	cin>>again; if(again=='y'){}

Type	Variable Name	Description	Location

bool	over	game is over	over=true;
			if(over){};
			over=false;

	move	move is valid	move=true;
			else{move=false;}
			while(!move);

	win	there is a win	win=false;
			if(win){};
			if(win==false);

Constructs:

Chapter	Syntax and Keywords	Location
1	using	using namespace std;
	namespace	using namespace std;
2	Arithmetic operators (+, -, *, /,%)	int random=49+(rand()%(57-49+1));
	endl	cout<<"+"<<endl;
	\n	", Choose An Option: \n"
	cout	cout<<"Type 1 to PLAY"<<endl;
	#include	#include <iostream>
	int	int player=1;
	unsigned	srand(static_cast<unsigned int>(time(0)));
	short	unsigned short winNum,loss,tie;
	string	string fName,lName;
	char	char choice;
3	static_cast	srand(static_cast<unsigned int>(time(0)));
	cin	cin>>choice;
	Equality operators(<,>)	for(int i=1;i<=3;i++)
	setw(x)	out<<"Win:"<<setw(3)<<winNum<<endl;

Chapter	Syntax and Keywords	Location
	srand()	srand(static_cast<unsigned int>(time(0)));

	out.open, out.close	ofstream out;
		out.open("stats.dat");
		out.close();

4	Relational operators(==,!=,<=,>=)	for(int i=1;i<=3;i++)
	if	if(next=='1'&&square1=='1')
	else if	else if(next=='2'&&square2=='2')
	else	else{move=false;}
	Logical operators (&&,||,!)	if(next=='1'&&square1=='1')
	switch	switch(choice){ case '1':{
	case	switch(choice){ case '1':{
	default	default:cout<<"Not option in menu."<<endl;
	break	case '2':{return 0; break;
5	Increment (++)	for(int i=1;i<=3;i++)
	for loop	for(int i=1;i<=3;i++)
	do/while loop	do { }while(!move);

Program Code:

```
/*
 * File:   main.cpp
 * Author: Jeannie Choi
 * Created on July 11th, 2016, 7:46 PM
 * Purpose: Project 1: Tic-Tac-Toe Game
 */

//System Libraries
#include <iostream>  //Input/Output Library
#include <ctime>     //Time for random seed
#include <cstdlib>   //Random number seed
#include <iomanip>   //Formatting
#include <cmath>     //Math Library
```

```
#include <fstream>   //File I/O
#include <string>    //String Object
using namespace std; //Namespace of the System Libraries

//User Libraries

//Global Constants

//Function Prototypes

//Execution Begins Here!
int main(int argc, char** argv) {
    //Set the random number seed
    srand(static_cast<unsigned int>(time(0)));

    //Open files and Input Data
    ofstream out;
    out.open("stats.dat");

    //Declare Variables
    char choice;
    string fName,lName;
    unsigned short winNum,loss,tie;

    //Input Data
    cout<<"Welcome to Tic-Tac-Toe Game!"<<endl;
    cout<<"What is your first Name?"<<endl;
    cin>>fName;
    cout<<"What is your last Name?"<<endl;
    cin>>lName;

    cout<<"Hello "<<fName<<" "<<lName<<", Choose An Option: \n"<<endl;
    cout<<"Type 1 to PLAY"<<endl;
    cout<<"Type 2 to QUIT"<<endl;
    cin>>choice;

    //Process the Data
    switch(choice){
        case '1':{
            //Declare Variables
            char
square1('1'),square2('2'),square3('3'),square4('4'),square5('5'),
                 square6('6'),square7('7'),square8('8'),square9('9');
            int  player=1;
            bool over(true);

            //Game loop
            do{
                //Setting Tic-Tac-Toe Game Board
                cout<<"|"<<square1<<"|"<<square2<<"|"<<square3<<"|"<<endl;
                for(int i=1;i<=3;i++)//cout<<"+-+-+-+"<<endl;
                cout<<"+-";
                cout<<"+"<<endl;
                cout<<"|"<<square4<<"|"<<square5<<"|"<<square6<<"|"<<endl;
                for(int i=1;i<=3;i++)//cout<<"+-+-+-+"<<endl;
                cout<<"+-";
                cout<<"+"<<endl;
                cout<<"|"<<square7<<"|"<<square8<<"|"<<square9<<"|"<<endl;
```

```
//Assigning Player The Game Marker
char mark;
if(player==1){
    mark='X';
}else{
    mark='O';
}

//Player's Move
bool move;
char next;
if(player==1){
    do{
        cout<<"Your turn: "<<endl;
        cin>>next;
        move=true;

        //Check if the move is valid
        if     (next=='1'&&square1=='1'){square1=mark;}
        else if(next=='2'&&square2=='2'){square2=mark;}
        else if(next=='3'&&square3=='3'){square3=mark;}
        else if(next=='4'&&square4=='4'){square4=mark;}
        else if(next=='5'&&square5=='5'){square5=mark;}
        else if(next=='6'&&square6=='6'){square6=mark;}
        else if(next=='7'&&square7=='7'){square7=mark;}
        else if(next=='8'&&square8=='8'){square8=mark;}
        else if(next=='9'&&square9=='9'){square9=mark;}
        else{cout<<"Try again."<< endl;
            move=false;}
     }while(!move);
}
//Computer's Move
else if(player==2){
    cout<<"Computer's turn: "<<endl;
    //Check if the move is valid
        do {
            int random=49+(rand()%(57-49+1));
            next=random;
            move=true;
        if     (next=='1'&&square1=='1'){square1=mark;}
        else if(next=='2'&&square2=='2'){square2=mark;}
        else if(next=='3'&&square3=='3'){square3=mark;}
        else if(next=='4'&&square4=='4'){square4=mark;}
        else if(next=='5'&&square5=='5'){square5=mark;}
        else if(next=='6'&&square6=='6'){square6=mark;}
        else if(next=='7'&&square7=='7'){square7=mark;}
        else if(next=='8'&&square8=='8'){square8=mark;}
        else if(next=='9'&&square9=='9'){square9=mark;}
        else{move=false;}
        }while(!move);
}

over    =false;
bool win=true;

//Checking Game Over Conditions
if(square1!='1'){
```

```
                    if(square2==square1&&square3==square1){
                        over=true;
                    }
                    if(square4==square1&&square7==square1){
                        over=true;
                    }
                }
                if(square5!='5'){
                    if(square1==square5&&square9==square5){
                        over=true;
                    }
                    if(square2==square5&&square8==square5){
                        over=true;
                    }
                    if(square4==square5&&square6==square5){
                        over=true;
                    }
                    if(square3==square5&&square7==square5){
                        over=true;
                    }
                }
                if(square9!='9'){
                    if(square3==square9&&square6==square9){
                        over=true;
                    }
                    if(square7==square9&&square8==square9){
                        over=true;
                    }
                }
                //Neither the player nor the computer wins
                if(square1!='1'&&square2!='2'&&square3!='3'&&
                   square4!='4'&&square5!='5'&&square6!='6'&&
                   square7!='7'&&square8!='8'&&square9!='9'&&!over){
                    over=true;
                    win=false;
                }

                //Game Over
                char again;
                if(over){
                    if(win){
                        if(player==1)
                            cout<<fName<<"'s win!"<<endl;
                        else cout<<"The computer wins!"<<endl;
                    }
                    if(win==false)
                        tie++;
                    else if(player==1)
                        winNum++;
                    else if(player==2)
                        loss++;


cout<<"|"<<square1<<"|"<<square2<<"|"<<square3<<"|"<<endl;
                    for(int i=1;i<=3;i++)//cout<<"+-+-+-+"<<endl;
                        cout<<"+-";
                    cout<<"+"<<endl;
```

```
cout<<"|"<<square4<<"|"<<square5<<"|"<<square6<<"|"<<endl;
                    for(int i=1;i<=3;i++)//cout<<"+-+-+-+"<<endl;
                        cout<<"+-";
                    cout<<"+"<<endl;

cout<<"|"<<square7<<"|"<<square8<<"|"<<square9<<"|"<<endl;

                    cout<<"G A M E O V E R"<<endl;
                    cout<<"Win:"<<setw(3)<<winNum<<endl;
                    cout<<"Loss:"<<setw(2)<<loss<<endl;
                    cout<<"Tie:"<<setw(3)<<tie<<endl;
                    cout<<"Would you like to play again? Type (y/n)"<<endl;
                    cin>>again;

                    if(again=='y'){
                        over=false;
                        //Setting new board
                        square1='1';
                        square2='2';
                        square3='3';
                        square4='4';
                        square5='5';
                        square6='6';
                        square7='7';
                        square8='8';
                        square9='9';
                    }
                    player=1;
                }else{
                    //Alternate player turns
                    if(player==1){
                        player=2;
                    }else{
                        player=1;
                    }
                }
            }while(!over);

            //Exit the switch
            break;
        }
        case '2':{
            return 0;
            //Exit the switch
            break;
        }
        default:cout<<"Not option in menu."<<endl;
    }

    //Output the processed Data
    out.close();
    out<<"Win:"<<setw(3)<<winNum<<endl;
    out<<"Loss:"<<setw(2)<<loss<<endl;
    out<<"Tie:"<<setw(3)<<tie<<endl;

    //Exit Stage Right!
    return 0;
```

```
}
```