## **Explanation of the program:**

The program has been written to develop a **Tic Tac Toe**, a two-player game. We will start the program by using #include, a preprocessor directive that tells the preprocessor to include header files in the program. < > indicate the start and end of the file name to be included. iostream is a header file that contains functions for input and output operations e.g., cin and cout.

Next, we will include three more header files; #include <conio.h> provides console input/output and the function \_getche() is used, #include <cmath> declares a set of functions to compute common mathematical operations and transformations throughout the program. #include <cstdlib> is the function prototype of rand produces an integer at random each time rand is called.

Now, we require the ‘using’ directive,

using namespace std;

char aray[10] declares an array with the  *size = 10*, named  *aray*, of type *char* with the elements = { 'o','1','2','3','4','5','6','7','8','9' };

void humanvscomputer() indicates the function. Here we will use {} brackets to start and end the function. srand(time(0)); will be used to randomly take the number while playing against the user. int check(); is the function used for finding out the winner as it has all the possible chances to win the game. void Display(); displays the matrix and the layout of the game. int player = 1, i, choice; are the integer variables that are declared and initialized. char mark; a character variable is declared as mark. In a do while loop, we will call the Display() function and let the game started by using an if else condition. The enter number by the use will be marked as ‘X’ whereas the computer’s randomly entered number will be marked as ‘0’. An if else statement nested in a while loop will be used for both the players.

if (choice == 1 && aray[1] == '1') { aray[1] = mark; placed = 1;} if choice is equal to 1 and(logical operator &&) the element stored in aray 1 is equal to 1 then ‘O’ is marked. Same steps will be done till if (choice == 9 && aray[9] == '9') { aray[9] = mark; placed = 1;}. In player’s turn, the player is asked to input a number on which their symbol (X) will be marked using If – Else Structure, if (choice == 1 && aray[1] == '1') aray[1] = mark; If the user enters any number other than 1-9 or repeats any number, Invalid move will be shown on screen. Once all the numbers are done, the result is shown on the screen.

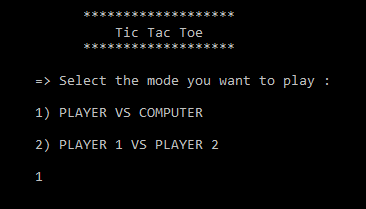
void Display() indicates the display function for humanvscomputer() function in which the layout as well as the design is programmed. For loops are used to print \* in the horizontal line.

void humanvshuman() is the function, it is called if the player 1 (X) selects to play with player 2 (0). It will have the same program as it was for the player in the humanvscomputer() function. The wining player will be announced at the end or Game Draw will be shown on the screen.

void Display() indicates the display function for humanvshuman() function in which the layout as well as the design is programmed. For loops are used to print \* in the horizontal line.

int main() indicates that the main() function. Here we will use {} brackets to start and end the function. Name of the game in between the \* is printed on the screen as well as the user is asked to select the mode, they want to play; 1)PLAYER VS COMPUTER or 2)PLAYER 1 VS PLAYER 2. If the user enters 1, humanvscomputer() function is called else humanvshuman() is called.

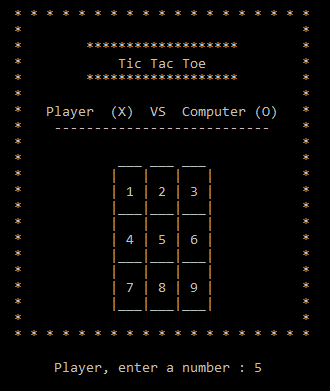
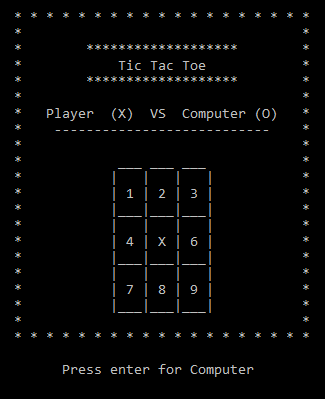
## **Output of the program:**



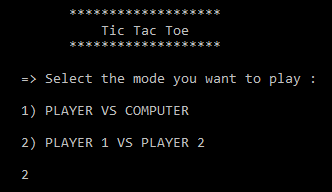
Player

VS

Computer





Player 1

VS

Player 2

