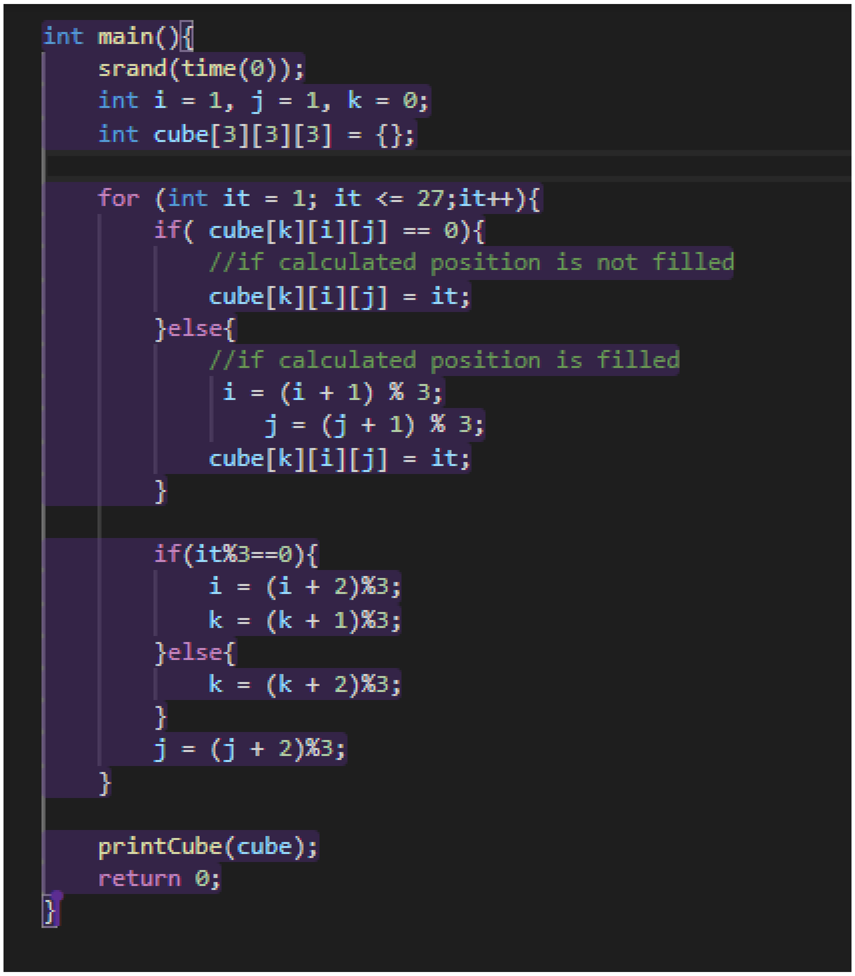
Artificial Intelligence
Assignment-1

Name of Student	Roll Number
Akarsh Tyagi	1910110040
Ayush Jagga	1910110106
Ayush Varma	1910110108

Q1) Write a program to generate a 3-D magic cube

Soln) We researched about the magic square and similarly using observation and analysis we designed an algorithm.



```
int main(){
    srand(time(0));
    int i = 1, j = 1, k = 0;
    int cube[3][3][3] = {};

    for (int it = 1; it <= 27;it++){
        if( cube[k][i][j] == 0){
            //if calculated position is not filled
            cube[k][i][j] = it;
        }else{
            //if calculated position is filled
             i = (i + 1) % 3;
                j = (j + 1) % 3;
            cube[k][i][j] = it;
        }

        if(it%3==0){
            i = (i + 2)%3;
            k = (k + 1)%3;
        }else{
            k = (k + 2)%3;
        }
        j = (j + 2)%3;
    }

    printCube(cube);
    return 0;
}
```

Q2) Write a program for 3-D Tic Tac Toe using the magic square concept.

Soln) Approach towards computer's move

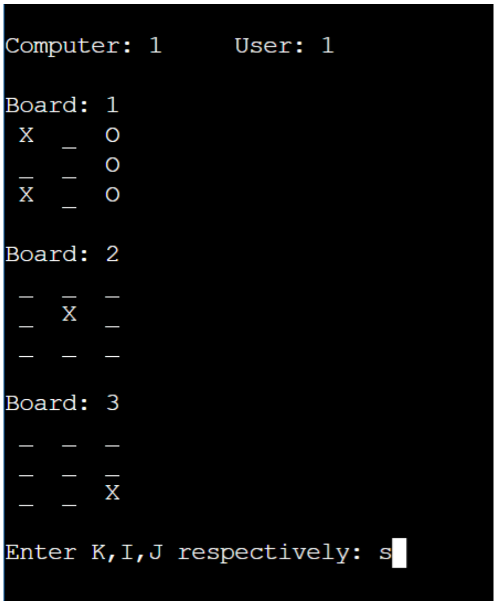
We use two vectors(dynamic arrays) to store the co-ordinate/position of the chances played by both user and computer. For the user, he/she can play at any position desired by him given that it has not been taken already. For computer, it analyses using both the human's and computer's already played positions to plot its next move.

Since it is a competitive game, so the first player to attain the minimum 'N' winning lines wins. Therefore, the computer prioritizes forming its own winning line over blocking the formers probable winning position.

We do this by traversing firstly the computers list to find an eligible co-ordinate according to the MAGIC CUBE rules (sum being 42) and then searches both the list to find if the position has already been occupied, if not, then it moves its cross(pawn) on that position.

Now, for countering the user's possible winning lines, we again find an eligible co-ordinate according to the MAGIC CUBE rules, and if that position is already not taken then occupy hence, blocking it.

There are special edge cases, which we have countered by deriving the logic for it using pseudo code.



```
Computer: 1     User: 1

Board: 1
 X  _  O
 _  _  O
 X  _  O

Board: 2
 _  _  _
 _  X  _
 _  _  _

Board: 3
 _  _  _
 _  _  _
 _  _  X

Enter K,I,J respectively: s
```