```
#include <iostream>
#include <cstdlib>
#include <ctime>
#include <conio.h>
#include <windows.h>
#include <string>
#pragma comment(lib, "winmm.lib")
 #include <graphics.h>
 #include<string.h>
#include<mmsystem.h>
#pragma comment(lib, "winmm.lib")
int userRoll_1, Roll_2, Roll_3;
int playerRoll_2,R_2,R_3;

bool congratulationDisplayed = false;
bool congratulationDisplayed_2 = false;


using namespace std;
 int sum_2=0;
int sum_1 = 0;
 void ladders_2();
 void snakes_2();
 void ladders();
  void snakes();
  void image();
  void image_2();
   void  screen();
   void Secondplayer();
   void menu();
void Firstplayer()
{
            menu();

    char key = ' ';


   while (key != 'q')
    {
      {
         system("color 0A");
      userRoll_1 = (std::rand() % 6) + 1; // Move the r
        cout << "FIRST PLayer_Roll  Dice" << endl;
        key = _getch();
        cout << "FIRST PLAYER Rolled: " << userRoll_1 <
            cout<<"PRESS (ESC) KEY TO CLOSE  GRAPHIC WI
             image();
              screen();

        if ((userRoll_1 == 6 || userRoll_1 == 1) && !co
        {

            cout << "Roll Dice Again As your Token has
            cout<<"************************************
            cout << "CONGRATULATION!! TOKEN OPEN:" << e
            congratulationDisplayed = true;
            sum_1 = 1;
            cout << "DESTINATION:" << sum_1 << endl;
            cout<<"PRESS (ESC) KEY TO CLOSE  GRAPHIC WI
```

```
     screen();


}
else
{
    if (congratulationDisplayed)
    {
        sum_1 += userRoll_1; // Add the roll to
    }
    else
    {
        sum_1 += userRoll_1; // Add the roll to

    }

    if (userRoll_1 != 6 && userRoll_1 != 1)
    {
       cout<<"=================================
    }
}

if (userRoll_1 == 6 || userRoll_1 == 1)
{
    Roll_2 = (std::rand() % 6) + 1; // Move the

    key = _getch();
    cout << "FIRST PLAYER Rolled: " << Roll_2 <

    sum_1 += Roll_2;
    cout << "DESTINATION:" << sum_1 << endl;
        cout<<"PRESS (ESC) KEY TO CLOSE  GRAPHI
                         ladders();
                         snakes();
                          image();
     screen();
    if (Roll_2 != 6 && Roll_2 != 1)
    {
            cout<<"============================
    }
}

while (Roll_2 == 6 || Roll_2 == 1)


{     ladders();
      snakes();
        cout<<"PRESS (ESC) KEY TO CLOSE  GRAPHI
     screen();
    cout << "FIRST PLAYER ROLL Dice:" << endl;
    key = _getche();
    Roll_3 = (std::rand() % 6) + 1;
    cout << "FIRST PLAYER Rolled: " << Roll_3 <

    sum_1 += Roll_3;
    cout << "Destination:" << sum_1 << endl;

    congratulationDisplayed = false;
```

```
           ladders();
             snakes();

            screen();
           if (Roll_3 != 6 && Roll_3 != 1)
           {

              cout<<"*********************************
                  Secondplayer();

           }


             Secondplayer();

       }

             Secondplayer();

       }


   }


  }
///second _player
   //int playerRoll_2,R_2,R_3;
void Secondplayer(){
        char key_2 = ' ';
```

```
playerRoll_2 = (std::rand() % 6) + 1; // Move the
  cout << "SECOND PLayer_Roll  Dice" << endl;
  key_2 = _getch();
  cout << "SECOND PLAYER Rolled: " << playerRoll_
   image_2();
      cout<<"PRESS (ESC) KEY TO CLOSE  GRAPHIC WI
   screen();


  if ((playerRoll_2 == 6 || playerRoll_2 == 1) &&
  {

      cout << "Roll Dice Again As your Token has
      cout<<"************************************
      cout << "CONGRATULATION!! TOKEN OPEN:" << e
      congratulationDisplayed_2 = true;
      sum_2 = 1;
      cout << "DESTINATION:" << sum_2 << endl;
          cout<<"PRESS (ESC) KEY TO CLOSE  GRAPHI
   screen();
  }
  else
  {
      if (congratulationDisplayed_2)
      {
          sum_2 += playerRoll_2; // Add the roll
      }
      else
      {
          sum_2 += playerRoll_2; // Add the roll
```

```
    }

    if ( playerRoll_2!= 6 && playerRoll_2 != 1)
    {
       cout<<"=================================
    }
}

if (playerRoll_2 == 6 || playerRoll_2 == 1)
{
    R_2 = (std::rand() % 6) + 1; // Move the ra

    key_2 = _getch();
    cout << "SECOND PLAYER Rolled: " << R_2 <<
    sum_2 += R_2;


    cout << "DESTINATION:" << sum_2 << endl;
        cout<<"PRESS (ESC) KEY TO CLOSE  GRAPHI
 screen();
            ladders_2();
      snakes_2();
    if (R_2 != 6 && R_2 != 1)
    {
            cout<<"============================
    }
}

while (R_2 == 6 || R_2 == 1)
```

```
        {            ladders_2();
            cout<<"PRESS (ESC) KEY TO CLOSE  GRAPHIC WI
              snakes_2();

         screen();
            cout << "SECOND PLAYER ROLLED Dice:" << end
            key_2 = _getche();
            R_3 = (std::rand() % 6) + 1;
            cout << "SECOND PLAYER Rolled: " << R_3 <<


            sum_2 += R_3;
            cout << "Destination:" << sum_2 << endl;
                 ladders_2();
              snakes_2();
            congratulationDisplayed_2 = false;

 }
}

//LADDER_1 POSITION
void ladders()
{


        if (sum_1 ==2)
        {
```

```
     sum_1=23;

        PlaySound(TEXT("clapping.wav"),NULL,SND
    cout << "LADDER_DESTINATION:" << sum_1 << e
}
else if(sum_1 == 6)    {

     sum_1=45;

            PlaySound(TEXT("clapping.wav"),NULL
    cout << "LADDER_DESTINATION:" << sum_1 << e

}

else if(sum_1==20)
{

     sum_1=42;
            PlaySound(TEXT("clapping.wav"),NULL

    cout << "LADDER_DESTINATION:" << sum_1 << e
}
else if(sum_1==52)
{
    sum_1=72;
        PlaySound(TEXT("clapping.wav"),NULL,SND
    cout<< "LADDER_DESTINITION:" <<sum_1<< endl
}
else if(sum_1==57)
{
```

```
             sum_1=96;

                PlaySound(TEXT("clapping.wav"),NULL,SND
            cout << "LADDER_DESTINATION:" << sum_1 << e
        }
        else if(sum_1==71)   {


             sum_1=92;

                    PlaySound(TEXT("clapping.wav"),NULL
            cout << "LADDER_DESTINATION:" << sum_1 << e
        }
 //SNAKE POSITION
}
 string play_1, play_2;
 void snakes()
 {

         if (sum_1 ==43)
        {


             sum_1=17;

                PlaySound(TEXT("OH.wav"),NULL,SND_SYNC)
                    PlaySound(TEXT("hiss.wav"),NULL,SND
            cout << "SNAKE SLIDE _DESTINATION:" << sum_
```

```
}

else if (sum_1 ==50)
{


     sum_1=5;

            PlaySound(TEXT("OH.wav"),NULL,SND_S
            PlaySound(TEXT("hiss.wav"),NULL,SND
    cout << "SNAKE SLIDE _DESTINATION:" << sum_
}
    else if (sum_1 ==56)
{


     sum_1=8;
            PlaySound(TEXT("OH.wav"),NULL,SND_S
            PlaySound(TEXT("hiss.wav"),NULL,SND

    cout << "SNAKE SLIDE _DESTINATION:" << sum_
}
    else if (sum_1 ==73)
{


     sum_1=15;

                PlaySound(TEXT("OH.wav"),NULL,S
            PlaySound(TEXT("hiss.wav"),NULL,SND
    cout << "SNAKE SLIDE _DESTINATION:" << sum_
}
    else if (sum_1 ==84)
{


     sum_1=58;
                PlaySound(TEXT("OH.wav"),NULL,S
            PlaySound(TEXT("hiss.wav"),NULL,SND

    cout << "SNAKE SLIDE _DESTINATION:" << sum_
}
    else if (sum_1 ==87)
{


     sum_1=49;

                PlaySound(TEXT("OH.wav"),NULL,S
            PlaySound(TEXT("hiss.wav"),NULL,SND
    cout << "SNAKE SLIDE _DESTINATION:" << sum_
}
    else if (sum_1 ==98)
{


     sum_1=40;
                    PlaySound(TEXT("OH.wav"),NU
            PlaySound(TEXT("hiss.wav"),NULL,SND
            cout << "SNAKE SLIDE _DESTINATION:" << sum_
        }
    else if(sum_1==100)
        {
            cout<<"WINNER OF MATCH IS PLAYER_1:"<<play_
            sum_1=0;
            cout<<"DO YOU WANTED TO PLAY AGAIN ! PRESS 
             getch();

        }
        else if(sum_1>100)
        {
            cout<<"OUT OF RANGE !!  ROLL  YOUR DICE  TI
               cout<<"Play till you reach to  limit -->
        }
        }
  //ladder  &snakes

void ladders_2()
{


        if (sum_2 ==2)
        {



             sum_2=23;

                PlaySound(TEXT("clapping.wav"),NULL,SND
            cout << "LADDER_DESTINATION:" << sum_2 << e
```

```
}
else if(sum_2 == 6)    {

     sum_2=45;

            PlaySound(TEXT("clapping.wav"),NULL
    cout << "LADDER_DESTINATION:" << sum_2 << e

}

else if(sum_2==20)
{

     sum_2=42;
            PlaySound(TEXT("clapping.wav"),NULL

    cout << "LADDER_DESTINATION:" << sum_2 << e
}
else if(sum_2==52)
{
    sum_2=72;
        PlaySound(TEXT("clapping.wav"),NULL,SND
    cout<< "LADDER_DESTINITION:" <<sum_2<< endl
}
else if(sum_2==57)
{

     sum_2=96;

        PlaySound(TEXT("clapping.wav"),NULL,SND
            cout << "LADDER_DESTINATION:" << sum_2 << e
        }
        else if(sum_2==71)   {



             sum_2=92;

                    PlaySound(TEXT("clapping.wav"),NULL
            cout << "LADDER_DESTINATION:" << sum_1 << e
        }
 //SNAKE POSITION
}

 void snakes_2()
 {

         if (sum_2 ==43)
        {


             sum_2=17;

                PlaySound(TEXT("OH.wav"),NULL,SND_SYNC)
                    PlaySound(TEXT("hiss.wav"),NULL,SND
            cout << "SNAKE SLIDE _DESTINATION:" << sum_
        }

        else if (sum_2 ==50)
        {
```

```
     sum_2=5;

            PlaySound(TEXT("OH.wav"),NULL,SND_S
            PlaySound(TEXT("hiss.wav"),NULL,SND
    cout << "SNAKE SLIDE _DESTINATION:" << sum_
}
    else if (sum_2 ==56)
{


     sum_2=8;
            PlaySound(TEXT("OH.wav"),NULL,SND_S
            PlaySound(TEXT("hiss.wav"),NULL,SND

    cout << "SNAKE SLIDE _DESTINATION:" << sum_
}
    else if (sum_2 ==73)
{


     sum_2=15;

                PlaySound(TEXT("OH.wav"),NULL,S
            PlaySound(TEXT("hiss.wav"),NULL,SND
    cout << "SNAKE SLIDE _DESTINATION:" << sum_
}
    else if (sum_2 ==84)
{
```

```
         sum_2=58;
                    PlaySound(TEXT("OH.wav"),NULL,S
                PlaySound(TEXT("hiss.wav"),NULL,SND

        cout << "SNAKE SLIDE _DESTINATION:" << sum_
    }
        else if (sum_2 ==87)
    {


         sum_2=49;

                    PlaySound(TEXT("OH.wav"),NULL,S
                PlaySound(TEXT("hiss.wav"),NULL,SND
        cout << "SNAKE SLIDE _DESTINATION:" << sum_
    }
        else if (sum_2 ==98)
    {


         sum_2=40;
                        PlaySound(TEXT("OH.wav"),NU
                PlaySound(TEXT("hiss.wav"),NULL,SND

        cout << "SNAKE SLIDE _DESTINATION:" << sum_
    }
else if(sum_2==100)
    {
            cout<<"WINNER OF MATCH IS Player_2:"<<play_
            sum_2=0;
            cout<<"DO YOU WANTED TO PLAY AGAIN ! PRESS
            getch();


        }
        else if(sum_2>100)
        {
            cout<<"OUT OF RANGE !!  ROLL  YOUR DICE  TI
            cout<<"Play till you reach to  limit -->100

        }
        }

    void  menu()
 {

    cout<<"SNAKES & LADDER  GAME " <<endl;
    cout<<"ENJOY IT!"<<endl;
    cout<<"*-------------------------------------------
    cout<<"********************PRESS(X)****************
    char a;
do{
    cout<<"Enter x to play the Game:"<<endl;
    cin>>a;
    if(a=='x')
    {
    cout<<"PRESS (q) for Quit  if you do not wanted to
    cout<<"TWO-PLAYERS GAME"<<endl;
    cout<<":"<<endl;
    cout<<"********************************************
    cout<<""<<endl;
    cout<<"RED TOKEN IS  FOR  FIRST -PLAYER "<<endl;
    cout<<"BLUE TOKEN IS FOR SECOND -PLAYER"<<endl;

    cout << "(2) PLAYERS GAME:" << endl;
      cout<<"Enter 5-keyword in your  Name(ONly First N
    cout << "ENTER FIRST PLAYER-NAME: " << endl;
    cin >> play_1;
    cout << "ENTER SECOND PLAYER-NAME: " << endl;
    cin >> play_2;
    cout<<"********************************************
    }
    else{
        cout<<"KEY NOT MATCHES!!"<<endl;
    }
    }while(a!='x');
 }


void image()
{

        if(userRoll_1==1){


     initwindow(800, 600, "DICE ROLL_1");
       readimagefile("dice_1.jpg", 0, 0, getmaxx(), get
    delay(500);
     closegraph();
 }
 else if(userRoll_1==2)
 {
     initwindow(800, 600, "DICE ROLL_1");
    readimagefile("dice_2.jpg", 0, 0, getmaxx(), get

    delay(500);
     closegraph();
 }
else if(userRoll_1==3){

 initwindow(800, 600, "DICE ROLL_1");
    readimagefile("dice_3.jpg", 0, 0, getmaxx(), get

    delay(500);
     closegraph();
  }
  else if(userRoll_1==4){

 initwindow(800, 600, "DICE ROLL_1");
    readimagefile("dice_4.jpg", 0, 0, getmaxx(), get

    delay(5000);
     closegraph();
  }
  else   if(userRoll_1==5){
```

```
initwindow(800, 600, "DICE ROLL_1");
   readimagefile("dice_5.jpg", 0, 0, getmaxx(), get

   delay(500);
    closegraph();
 }
 else   if(userRoll_1==6){

initwindow(800, 600, "DICE ROLL_1");
   readimagefile("dice_6.jpg", 0, 0, getmaxx(), get

   delay(500);
    closegraph();
 }
  }
  void image_2()
  {

    if(playerRoll_2==1){

initwindow(800, 600, "DICE ROLL_2");
   readimagefile("dice_1.jpg", 0, 0, getmaxx(), get

   delay(500);
    closegraph();
    }     else  if(playerRoll_2==2){

initwindow(800, 600, "DICE ROLL_2");
   readimagefile("dice_2.jpg", 0, 0, getmaxx(), get
   delay(500);
    closegraph();

 }
      else  if(playerRoll_2==3){

initwindow(800, 600, "DICE ROLL_2");
   readimagefile("dice_3.jpg", 0, 0, getmaxx(), get

   delay(500);
    closegraph();
}
      else  if(playerRoll_2==4){

initwindow(800, 600, "DICE ROLL_2");
   readimagefile("dice_4.jpg", 0, 0, getmaxx(), get

   delay(500);
    closegraph();
}
else    if(playerRoll_2==5){

initwindow(800, 600, "DICE ROLL_2");
   readimagefile("dice_5.jpg", 0, 0, getmaxx(), get

   delay(500);
    closegraph();
}
else    if(playerRoll_2==6){
    initwindow(800, 600, "DICE ROLL_2");
       readimagefile("dice_6.jpg", 0, 0, getmaxx(), get

       delay(500);
        closegraph();
    }
    }

         void screen()
{
    char a;

    initwindow(800, 600, "SNAKES &LADDERS -BOARD");

    int x = 1, y = 539;  // Initial coordinates
    int width = 80, height = 70;
    int j = 1, k = 539;  // Initial coordinates
    int w = 80, h = 70;

    // Load and display background image
    readimagefile("snake.jpg", 0, 0, getmaxx(), getmaxy

    while (true) {
        if (_kbhit()) {
            char key = _getch();
            if (key == 27) {  // 27 is the ASCII code f
                break;
            }

            // Update coordinates based on arrow key
```

```
if (key == 72) {  // Up arrow key
    y -= 10;
} else if (key == 80) {  // Down arrow key
    y += 10;
} else if (key == 75) {  // Left arrow key
    x -= 10;
} else if (key == 77) {  // Right arrow key
    x += 10;
}

if (key == 27) {  // 27 is the ASCII code f
    break;
}

// Update coordinates based on arrow key
if (key == 'a') {  // Up arrow key
    j -= 10;
} else if (key == 'd') {  // Down arrow key
    j += 10;
} else if (key == 'w') {  // Left arrow key
    k -= 10;
} else if (key == 's') {  // Right arrow ke
    k += 10;
}

cleardevice();
// Draw on the off-screen buffer
readimagefile("snake.jpg", 0, 0, getmaxx(),
readimagefile("r.jpg", x, y, x + width, y +
readimagefile("b.jpg", j, k, j + w, k + h);
```

```
            // Swap the off-screen buffer with the on-s
            swapbuffers();

            // Introduce a delay of 100 milliseconds

        }

    }

    // Close the graphics window
    closegraph();
}




int main()
{
    srand(static_cast<unsigned int>(time(0)));
    Firstplayer();
    snakes();
     image();
     image_2();
     screen();
     ladders();
      snakes_2();
     ladders_2();
```

```
          menu();
        Secondplayer();
       return 0;
   }
```