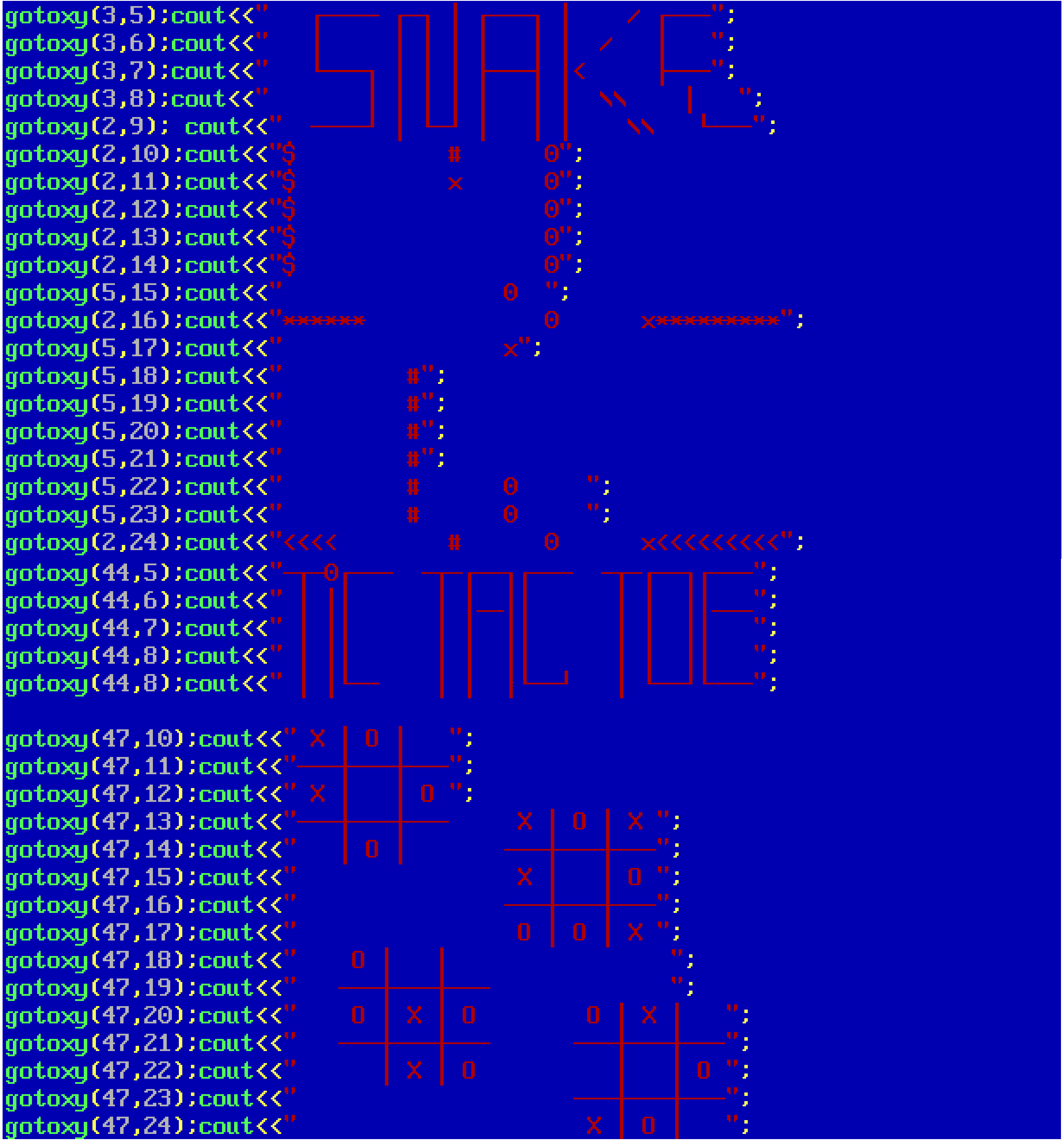
```
#include<fstream.h>
#include<conio.h>
#include<string.h>
#include<ctype.h>
#include<dos.h>
#include<iomanip.h>
#include<stdlib.h>

///////////////////////////////////////////////////////////////////////////////////////////////
//////////////////////////////////////TIC TAC TOE GAME/////////////////////////////////////////
///////////////////////////////////////////////////////////////////////////////////////////////


struct game
{
       int pos[4];
       char tictoe[10];
};

void snake();
void tictactoe();

void game_choose()
{
     clrscr();
     int i,j;
     char ch='a';  //any value except 1,2,0
     for(i=1,j=4;j<=25;j++)
    {
      gotoxy(i,j);
      cout<<(char)186;
    }
    for(i=38,j=4;j<=25;j++)
    {
      gotoxy(i,j);
      cout<<(char)186;
    }
    for(i=1,j=4;i<=38;i++)
    {
      gotoxy(i,j);
      cout<<(char)205;
    }
    for(i=1,j=25;i<=38;i++)
    {
      gotoxy(i,j);
      cout<<(char)205;
    }
    gotoxy(38,25);cout<<(char)188;
    gotoxy(38,4);cout<<(char)187;
    gotoxy(1,25);cout<<(char)200;
    gotoxy(1,4);cout<<(char)201;


    for(i=42,j=4;j<=25;j++)
    {
      gotoxy(i,j);
      cout<<(char)186;
    }
    for(i=79,j=4;j<=25;j++)
    {
      gotoxy(i,j);
      cout<<(char)186;
    }
    for(i=42,j=4;i<=79;i++)
    {
```

```
   gotoxy(i,j);
   cout<<(char)205;
}
for(i=42,j=25;i<=79;i++)
{
   gotoxy(i,j);
   cout<<(char)205;
}
gotoxy(79,25);cout<<(char)188;
gotoxy(79,4);cout<<(char)187;
gotoxy(42,25);cout<<(char)200;
gotoxy(42,4);cout<<(char)201;
```



```
gotoxy(3,5);cout<<"   ┌──── ┌─┐ │ ┌───┐ │    / ┌───";
gotoxy(3,6);cout<<"   │     │ │ │ │   │ │  /   │   ";
gotoxy(3,7);cout<<"   └───┐ │ │ │ ├───┤ │<     ├───";
gotoxy(3,8);cout<<"       │ │ │ │ │   │ │  \\    │   ";
gotoxy(2,9); cout<<"  ────┘ │ └─┘ │   │ │    \\   └───";
gotoxy(2,10);cout<<"$           #      0";
gotoxy(2,11);cout<<"$           x      0";
gotoxy(2,12);cout<<"$                  0";
gotoxy(2,13);cout<<"$                  0";
gotoxy(2,14);cout<<"$                  0";
gotoxy(5,15);cout<<"                0  ";
gotoxy(2,16);cout<<"******             0      x*********";
gotoxy(5,17);cout<<"                x";
gotoxy(5,18);cout<<"         #";
gotoxy(5,19);cout<<"         #";
gotoxy(5,20);cout<<"         #";
gotoxy(5,21);cout<<"         #";
gotoxy(5,22);cout<<"         #      0     ";
gotoxy(5,23);cout<<"         #      0     ";
gotoxy(2,24);cout<<"<<<<        #      0      x<<<<<<<<<";
gotoxy(44,5);cout<<"─┬─0┌───  ─┬─┌──┐┌───  ─┬─┌──┐┌───";
gotoxy(44,6);cout<<" │ ││      │ ├──┤│      │ │  │├───";
gotoxy(44,7);cout<<" │ ││      │ │  ││      │ │  ││   ";
gotoxy(44,8);cout<<" │ ││      │ │  ││      │ │  ││   ";
gotoxy(44,8);cout<<" │ │└───   │ │  │└──┘   │ └──┘└───";

gotoxy(47,10);cout<<" X │ 0 │   ";
gotoxy(47,11);cout<<"───┼───┼───";
gotoxy(47,12);cout<<" X │   │ 0 ";
gotoxy(47,13);cout<<"───┼───┼───     X │ 0 │ X ";
gotoxy(47,14);cout<<"   │ 0 │       ───┼───┼───";
gotoxy(47,15);cout<<"                X │   │ 0 ";
gotoxy(47,16);cout<<"               ───┼───┼───";
gotoxy(47,17);cout<<"                0 │ 0 │ X ";
gotoxy(47,18);cout<<"    0 │   │                ";
gotoxy(47,19);cout<<"   ───┼───┼───             ";
gotoxy(47,20);cout<<"    0 │ X │ 0        0 │ X │   ";
gotoxy(47,21);cout<<"   ───┼───┼───      ───┼───┼───";
gotoxy(47,22);cout<<"      │ X │ 0          │   │ 0 ";
gotoxy(47,23);cout<<"                    ───┼───┼───";
gotoxy(47,24);cout<<"                     X │ 0 │   ";
```

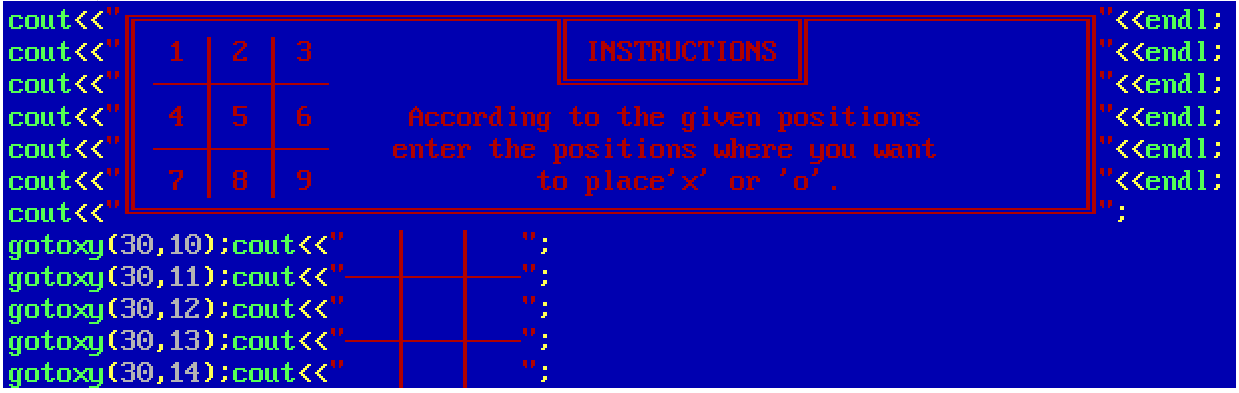
```
gotoxy(12,3);cout<<"1. RATTLE SNAKE";
gotoxy(55,3);cout<<"2. TIC TAC TOE";
gotoxy(74,1);cout<<"<0>EXIT";
while(ch!='1'&&ch!='2'&&ch!='0')
{
    if(kbhit())
    {
        ch=getch();
    }
}
```

```
    if(ch=='1')
        snake();
    if(ch=='2')
        tictactoe();
    if(ch=='0')
        exit(0);

    getch();
}


char all_places_fill(game t)//checks if user has filled all the places of tictactoe
{
        int i;
        for(i=0;i<9;i++)
        {
                if(t.tictoe[i]=='1')
                        return('n');
        }
        return('y');
}
char is_pos_repeate(int op,game t)
{
        if(t.tictoe[op-1]=='1')
                return('n');
        return('y');
}
game position_teller(int op)//returns the equivalent cursor position of int op(variable passed
to it)
{
        game position;
        position.pos[1]=0;
        if(op==1)
        {
                position.pos[0]=31;
                position.pos[2]=10;
                return(position);
        }
        if(op==2)
        {
                position.pos[0]=35;
                position.pos[2]=10;
                return(position);
        }
        if(op==3)
        {
                position.pos[0]=39;
                position.pos[2]=10;
                return(position);
        }
        if(op==4)
        {
                position.pos[0]=31;
                position.pos[2]=12;
                return(position);
        }
        if(op==5)
        {
                position.pos[0]=35;
                position.pos[2]=12;
                return(position);
        }
        if(op==6)
        {
                position.pos[0]=39;
```

```
            position.pos[2]=12;
            return(position);
      }
      if(op==7)
      {
            position.pos[0]=31;
            position.pos[2]=14;
            return(position);
      }
      if(op==8)
      {
            position.pos[0]=35;
            position.pos[2]=14;
            return(position);
      }
      if(op==9)
      {
            position.pos[0]=39;
            position.pos[2]=14;
            return(position);
      }
      position.pos[1]='-1';
      return(position);
}

char result(game &t)//checks the result of the match
{
      if(t.tictoe[0]=='o'&&t.tictoe[1]=='o'&&t.tictoe[2]=='o')
      {
            gotoxy(31,10);cout<<"O";
            gotoxy(35,10);cout<<"O";
            gotoxy(39,10);cout<<"O";
            return ('o');
      }
      if(t.tictoe[3]=='o'&&t.tictoe[4]=='o'&&t.tictoe[5]=='o')
      {
            gotoxy(31,12);cout<<"O";
            gotoxy(35,12);cout<<"O";
            gotoxy(39,12);cout<<"O";
            return ('o');
      }
      if(t.tictoe[6]=='o'&&t.tictoe[7]=='o'&&t.tictoe[8]=='o')
      {
            gotoxy(31,14);cout<<"O";
            gotoxy(35,14);cout<<"O";
            gotoxy(39,14);cout<<"O";
            return ('o');
      }
      if(t.tictoe[0]=='o'&&t.tictoe[3]=='o'&&t.tictoe[6]=='o')
      {
            gotoxy(31,10);cout<<"O";
            gotoxy(31,12);cout<<"O";
            gotoxy(31,14);cout<<"O";
            return ('o');
      }
      if(t.tictoe[1]=='o'&&t.tictoe[4]=='o'&&t.tictoe[7]=='o')
      {
            gotoxy(35,10);cout<<"O";
            gotoxy(35,12);cout<<"O";
            gotoxy(35,14);cout<<"O";
            return ('o');
      }
```

```
if(t.tictoe[2]=='o'&&t.tictoe[5]=='o'&&t.tictoe[8]=='o')
{
      gotoxy(39,10);cout<<"O";
      gotoxy(39,12);cout<<"O";
      gotoxy(39,14);cout<<"O";
      return ('o');
}
if(t.tictoe[0]=='o'&&t.tictoe[4]=='o'&&t.tictoe[8]=='o')
{
      gotoxy(31,10);cout<<"O";
      gotoxy(35,12);cout<<"O";
      gotoxy(39,14);cout<<"O";
      return ('o');
}
if(t.tictoe[2]=='o'&&t.tictoe[4]=='o'&&t.tictoe[6]=='o')
{
      gotoxy(39,10);cout<<"O";
      gotoxy(35,12);cout<<"O";
      gotoxy(31,14);cout<<"O";
      return ('o');
}

if(t.tictoe[0]=='x'&&t.tictoe[1]=='x'&&t.tictoe[2]=='x')
{
      gotoxy(31,10);cout<<"X";
      gotoxy(35,10);cout<<"X";
      gotoxy(39,10);cout<<"X";
      return ('x');
}
if(t.tictoe[3]=='x'&&t.tictoe[4]=='x'&&t.tictoe[5]=='x')
{
      gotoxy(31,12);cout<<"X";
      gotoxy(35,12);cout<<"X";
      gotoxy(39,12);cout<<"X";
      return ('x');
}
if(t.tictoe[6]=='x'&&t.tictoe[7]=='x'&&t.tictoe[8]=='x')
{
      gotoxy(31,14);cout<<"X";
      gotoxy(35,14);cout<<"X";
      gotoxy(39,14);cout<<"X";
      return ('x');
}
if(t.tictoe[0]=='x'&&t.tictoe[3]=='x'&&t.tictoe[6]=='x')
{
      gotoxy(31,10);cout<<"X";
      gotoxy(31,12);cout<<"X";
      gotoxy(31,14);cout<<"X";
      return ('x');
}
if(t.tictoe[1]=='x'&&t.tictoe[4]=='x'&&t.tictoe[7]=='x')
{
      gotoxy(35,10);cout<<"X";
      gotoxy(35,12);cout<<"X";
      gotoxy(35,14);cout<<"X";
      return ('x');
}
if(t.tictoe[2]=='x'&&t.tictoe[5]=='x'&&t.tictoe[8]=='x')
{
      gotoxy(39,10);cout<<"X";
      gotoxy(39,12);cout<<"X";
      gotoxy(39,14);cout<<"X";
      return ('x');
}
if(t.tictoe[0]=='x'&&t.tictoe[4]=='x'&&t.tictoe[8]=='x')
```

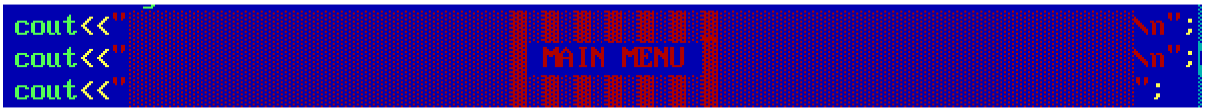
```
	{
		gotoxy(31,10);cout<<"X";
		gotoxy(35,12);cout<<"X";
		gotoxy(39,14);cout<<"X";
		return ('x');
	}
	if(t.tictoe[2]=='x'&&t.tictoe[4]=='x'&&t.tictoe[6]=='x')
	{
		gotoxy(39,10);cout<<"X";
		gotoxy(35,12);cout<<"X";
		gotoxy(31,14);cout<<"X";
		return ('x');
	}

	return ('d');
}

void multiplayer()
{
    clrscr();
    int i,sign,x,y,op;
    char fill,choice,check,res,player1,player2;
    game t,pos_right;
    do
    {
      strcpy(t.tictoe,"111111111");
      i=1;sign=1;
      pos_right.pos[1]=0;
      res='a';
      clrscr();
```



```
cout<<"╔════════════════════════════╦════════════╦══════════════════════╗"<<endl;
cout<<"║  1 │ 2 │ 3                 ║INSTRUCTIONS║                      ║"<<endl;
cout<<"║ ───┼───┼───                ╚════════════╝                      ║"<<endl;
cout<<"║  4 │ 5 │ 6        According to the given positions             ║"<<endl;
cout<<"║ ───┼───┼───      enter the positions where you want            ║"<<endl;
cout<<"║  7 │ 8 │ 9              to place'x' or 'o'.                    ║"<<endl;
cout<<"╚═════════════════════════════════════════════════════════════════╝";
gotoxy(30,10);cout<<"    │   │    ";
gotoxy(30,11);cout<<"────┼───┼────";
gotoxy(30,12);cout<<"    │   │    ";
gotoxy(30,13);cout<<"────┼───┼────";
gotoxy(30,14);cout<<"    │   │    ";
```

```
      gotoxy(25,15);
      cout<<"What do you want(x/o): ";
      cin>>player1;
      player1=tolower(player1);
      if(player1!='o'&&player1!='x')
		cout<<"\nWrong choice!";
      else
      {
	   if(player1=='o')
	     player2='x';
	   else
	     player2='o';
	   while(all_places_fill(t)!='y')
	   {
	     gotoxy(1,16);
	     if(sign==1)
		 cout<<"Enter the position("<<player1<<"'s turn)=";
	     if(sign==-1)
		 cout<<"Enter the position("<<player1<<"'s turn)=";
	     clreol();
	     cin>>op;
	     pos_right=position_teller(op);
	     if(pos_right.pos[1]=='-1')
```

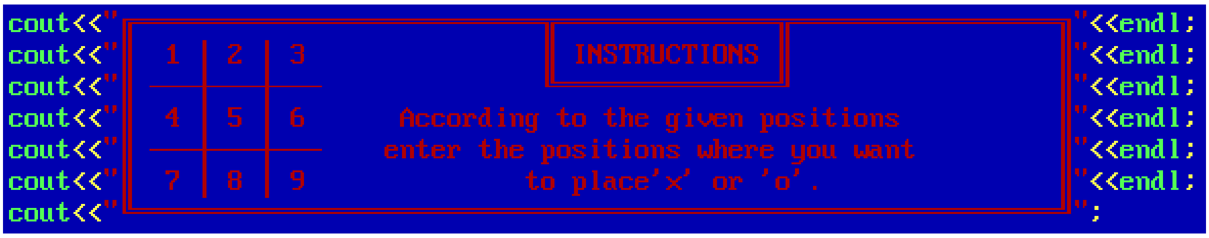
```
            {
                cout<<"\nYou entered wrong position!";
                getch();gotoxy(1,18);clreol();

            }
            else
            {
                check=is_pos_repeate(op,t);
                if(check=='y')
                  {
                      cout<<"\nPosition is already occupied!";
                      getch();gotoxy(1,18);clreol();
                      continue;
                  }
                  x=pos_right.pos[0];
                  y=pos_right.pos[2];
                  gotoxy(x,y);
                  if(sign==1)
                  {
                      cout<<player1;
                      t.tictoe[op-1]=player1;
                  }
                  else
                  {
                      cout<<player2;
                      t.tictoe[op-1]=player2;
                  }
                  sign=sign*-1;
            }
            if(i>=5)
                res=result(t);
            if(res=='x'||res=='o')
                break;
            i++;
          }
          gotoxy(1,19);
          if(res=='x')
            cout<<"\t\t\t\t'X' WINS"<<endl;
          if(res=='o')
            cout<<"\t\t\t\t'O' WINS"<<endl;
          if(res=='d')
            cout<<"\t\t\t    Match is DRAW"<<endl;
     }//end of main else
     cout<<"\n\t\t\t ONE MORE GAME ?(y/n): ";
     cin>>choice;
    }while(choice=='y'||choice=='Y');

}//end of game_start()

void tictactoe_menu()
{
    clrscr();
    int op,option;
```



```
    cout<<"                                                  \n";
    cout<<"                      MAIN MENU                   \n";
    cout<<"                                                    ";
```

```
    cout<<"\n\n\n(1)PLAY\n(2)INSTRUCTION\n(3)QUIT";
    cout<<"\n\nEnter your option : ";
    cin>>op;
    switch(op)
    {
      case 1:multiplayer();
             break;
```

```
       case 2:clrscr();
              cout<<"\n\n\n\n\n\n\n\n";
```



```
cout<<"                                                             "<<endl;
cout<<"  1 | 2 | 3                  INSTRUCTIONS                    "<<endl;
cout<<" ---+---+---                                                  "<<endl;
cout<<"  4 | 5 | 6       According to the given positions           "<<endl;
cout<<" ---+---+---     enter the positions where you want          "<<endl;
cout<<"  7 | 8 | 9             to place'x' or 'o'.                  "<<endl;
cout<<"                                                             ";
```

```
             getch();
              break;
      case 3:game_choose();
              break;
      default:cout<<"\nYou entered wrong choice!";
    }//end of switch()
    tictactoe_menu();
}//end of menu()

void tictactoe()
{
    clrscr();
    int i,j;
    cout<<"    _______  _  ______     _______  _____  ______     _______  _____  ______ \n";
    cout<<"   |__   __|| ||  ____|   |__   __||  _  ||  ____|   |__   __||  _  ||  ____|\n";
    cout<<"      | |   | || |           | |   | |_| || |           | |   | | | || |____ \n";
    cout<<"      | |   | || |           | |   |  _  || |           | |   | | | ||   ____|\n";
    cout<<"      | |   | || |____       | |   | | | || |____       | |   | |_| || |____ \n";
    cout<<"      |_|   |_||______|      |_|   |_| |_||______|      |_|   |_____||______|\n";
    for(i=31,j=7;j<=25;j++)
    {
      gotoxy(i,j);
      cout<<(char)220;
    }
    for(i=47,j=7;j<=25;j++)
    {
      gotoxy(i,j);
      cout<<(char)220;
    }
    for(i=18,j=12;i<=60;i++)
    {
      gotoxy(i,j);
      if(i%2==0)
             cout<<(char)220;
      else
             cout<<" ";
    }
    for(i=18,j=19;i<=60;i++)
    {
      gotoxy(i,j);
      if(i%2==0)
             cout<<(char)220;
      else
             cout<<" ";
    }

    gotoxy(51,7);cout<<" //---\\\\\n";
    gotoxy(51,8);cout<<"|       |\n";
    gotoxy(51,9);cout<<"|       |\n";
    gotoxy(51,10);cout<<"|       |\n";
    gotoxy(51,11);cout<<" \\\\---//\n";
    gotoxy(35,14);cout<<"=\\     /=   \n";
    gotoxy(35,15);cout<<"   \\ /     \n";
    gotoxy(35,16);cout<<"    X     \n";
    gotoxy(35,17);cout<<"   / \\    \n";
    gotoxy(35,18);cout<<"=/     \\= \n";
    gotoxy(20,7);cout<<" //---\\\\\n";
    gotoxy(20,8);cout<<"|       |\n";
    gotoxy(20,9);cout<<"|       |\n";
    gotoxy(20,10);cout<<"|       |\n";
    gotoxy(20,11);cout<<" \\\\---//\n";

    gotoxy(51,21);cout<<"=\\     /=   \n";
    gotoxy(51,22);cout<<"   \\ /     \n";
    gotoxy(51,23);cout<<"    X     \n";
    gotoxy(51,24);cout<<"   / \\    \n";
    gotoxy(51,25);cout<<"=/     \\=   ";

    gotoxy(20,21);cout<<" //---\\\\\n";
    gotoxy(20,22);cout<<"|       |\n";
    gotoxy(20,23);cout<<"|       |\n";
    gotoxy(20,24);cout<<"|       |\n";
    gotoxy(20,25);cout<<" \\\\---//";
    getch();
    tictactoe_menu();
}

/////////////////////////////////////////////////////////////////////////////////////////////
//////////////////////////////////////////SNAKE GAME/////////////////////////////////////////
/////////////////////////////////////////////////////////////////////////////////////////////

struct hss
{
    char body;// '*' or ' '
    int pos[2];
    int food_eaten;
};
struct coordinate
{
       int x,y;
};
void snake_menu();  //prototype
void instruction(); //prototype

char coordinate_check(hss snake[],coordinate level[],int dir,int size)
{
       int i,j;
       int x,y;//exact value of border coordinates

       //checks if body interects
       x=snake[0].pos[0];
       y=snake[0].pos[1];
       for(i=1;i<size;i++)
       {
           if(snake[i].pos[0]==x&&snake[i].pos[1]==y)
             return('y');
       }
       if(dir==1||dir==2||dir==3||dir==4)
       {
           for(i=level[0].x,j=level[0].y;i>11&&i<=70&&j>4&&j<=22;)
           {
             if(i==snake[0].pos[0]&&j==snake[0].pos[1])
                 return('y');
             if(dir==1)//left
                 i--;
             if(dir==3)//right
                 i++;
             if(dir==4)//up
                 j--;
             if(dir==2)//down
                 j++;
           }
       }
       //travel through main boundaries
       for(x=11,y=4;x<=70;x++)
       {
           if(x==snake[0].pos[0]&&snake[0].pos[1]==y)
           {
             snake[0].pos[1]=21;
             gotoxy(x,y);
             cout<<(char)205;
           }
       }
       for(x=11,y=22;x<=70;x++)
       {
           if(x==snake[0].pos[0]&&snake[0].pos[1]==y)
           {
             snake[0].pos[1]=5;
             gotoxy(x,y);
             cout<<(char)205;
           }
       }
       for(x=11,y=4;y<=22;y++)
       {
           if(x==snake[0].pos[0]&&snake[0].pos[1]==y)
           {
             snake[0].pos[0]=69;
             gotoxy(x,y);
             cout<<(char)186;
           }
       }
       for(x=70,y=4;y<=22;y++)
       {
           if(x==snake[0].pos[0]&&snake[0].pos[1]==y)
           {
             snake[0].pos[0]=12;
             gotoxy(x,y);
             cout<<(char)186;
           }
       }
       return('n');
}

int level_gen(coordinate level[],int &dir)
{
    int i,j;
    randomize();
    i=level[0].x;
    j=level[0].y;
    //to erase earlier boundary
    if(dir==1||dir==2||dir==3||dir==4)
    {
       if(dir==1)//left
       {
           for(;i>=12;i--)
           {
             gotoxy(i,j);
             cout<<" ";
           }
       }
```

```
      if(dir==3)//right
      {
          for(;i<70;i++)
          {
            gotoxy(i,j);
            cout<<" ";
          }
      }
      if(dir==4)//up
      {
          for(;j>=5;j--)
          {
            gotoxy(i,j);
            cout<<" ";
          }
      }
      if(dir==2)//down
      {
          for(;j<22;j++)
          {
            gotoxy(i,j);
            cout<<" ";
          }
      }
    }
    //to create new boundary
    i=level[0].x=random(10)+36;
    j=level[0].y=random(11)+8;
    dir=random(4)+1; // 1=left   2=down   3=right   4=up
    if(dir==1)//left
    {
      for(;i>=12;i--)
      {
          gotoxy(i,j);
          cout<<(char)196;
      }
    }
    if(dir==3)//right
    {
      for(;i<70;i++)
      {
          gotoxy(i,j);
          cout<<(char)196;
      }
    }
    if(dir==4)//up
    {
      for(;j>=5;j--)
      {
          gotoxy(i,j);
          cout<<(char)179;
      }
    }
    if(dir==2)//down
    {
      for(;j<22;j++)
      {
          gotoxy(i,j);
          cout<<(char)179;
      }
    }
    delay(500);
    return(dir);

}//end of level_gen()
```

```
coordinate food_gen(hss snake[],coordinate level[],coordinate &pos,int &dir)
{
    randomize();
    int i;
    pos.x=random(48)+12;
    pos.y=random(16)+5;
    if((snake[0].food_eaten%4)==0&&snake[0].food_eaten!=0)//because 0 is divisible by 4
      dir=level_gen(level,dir);
    if(pos.y==level[0].y)
    {
      if(dir==1&&pos.x<=level[0].x)
          pos.y++;
      if(dir==3&&pos.x>=level[0].x)
          pos.y++;
    }
    if(pos.x==level[0].x)
    {
      if(dir==2&&pos.y>=level[0].y)
          pos.x++;
      if(dir==4&&pos.y<=level[0].y)
          pos.x++;
    }
    gotoxy(pos.x,pos.y);
    cout<<(char)157;
    return(pos);
}//end of food_gen()

void chk_food_eaten(hss snake[],coordinate level[],coordinate &pos,int &dir,int &size,int
&speed)
{
    int x,y,i;
    int s=size;
    if(snake[0].pos[0]==pos.x&&snake[0].pos[1]==pos.y)
    {
      snake[0].food_eaten++;
      gotoxy(44,24);
      cout<<snake[0].food_eaten*10;
      pos=food_gen(snake,level,pos,dir);
      snake[size-1].body='>';
      if(snake[0].food_eaten<=18)
          size+=4;
      snake[size-1].body=' ';
      if(speed>20)
          speed-=15;
      if(speed>10&&speed<=20)
          speed-=10;
      if(speed>4&&speed<=10)
          speed-=2;
    }
}//end of chk_food_eaten()

char up(hss snake[],coordinate level[],int &size,coordinate &pos,int &dir,int &speed)
{
    char move;
    int i,j;
    while(i!=0)
    {
      if(kbhit()!=0)
      {
          move=getch();
          if(move=='a'||move=='d'||move=='A'||move=='D')
             return(move);
      }
      //to interchange the coordinates
      for(i=size+4;i>0;i--)
```

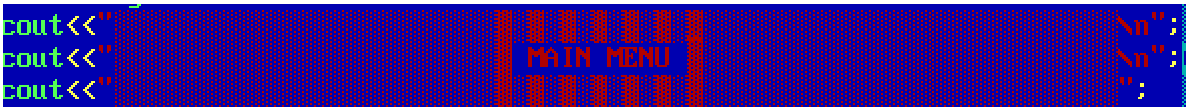
```
       {
           snake[i].pos[0]=snake[i-1].pos[0];
           snake[i].pos[1]=snake[i-1].pos[1];
       }
       snake[0].pos[1]--;

       //to display the body
       for(i=0,j=size-1;i<=j;i++,j--)
       {
           delay(speed);

           gotoxy(snake[i].pos[0],snake[i].pos[1]);
           cout<<snake[i].body;

           gotoxy(snake[j].pos[0],snake[j].pos[1]);
           cout<<snake[j].body;
       }
       chk_food_eaten(snake,level,pos,dir,size,speed);
       gotoxy(pos.x,pos.y);
       cout<<(char)235;
       if(coordinate_check(snake,level,dir,size)=='y')
           return('\0');
    }//end of while
}//end of up()

char left(hss snake[],coordinate level[],int &size,coordinate &pos,int &dir,int &speed)
{
    char move;
    int i,j;
    while(i!=0)
    {
       if(kbhit()!=0)
       {
           move=getch();
           if(move=='w'||move=='s'||move=='W'||move=='S')
             return(move);
       }
       //to interchange the coordinates
       for(i=size+4;i>0;i--)
       {
           snake[i].pos[0]=snake[i-1].pos[0];
           snake[i].pos[1]=snake[i-1].pos[1];
       }
       snake[0].pos[0]--;

       //to display the body
       for(i=0,j=size-1;i<=j;i++,j--)
       {
           delay(speed);

           gotoxy(snake[i].pos[0],snake[i].pos[1]);
           cout<<snake[i].body;

           gotoxy(snake[j].pos[0],snake[j].pos[1]);
           cout<<snake[j].body;
       }
       chk_food_eaten(snake,level,pos,dir,size,speed);
       gotoxy(pos.x,pos.y);
       cout<<(char)12;
       if(coordinate_check(snake,level,dir,size)=='y')
           return('\0');
    }//end of while
}//end of left()
```

```
char down(hss snake[],coordinate level[],int &size,coordinate &pos,int &dir,int &speed)
{
    char move;
    int i,j;
    while(i!=0)
    {
      if(kbhit()!=0)
      {
          move=getch();
          if(move=='a'||move=='d'||move=='A'||move=='D')
             return(move);
      }
      //to interchange the coordinates
      for(i=size+4;i>0;i--)
      {
          snake[i].pos[0]=snake[i-1].pos[0];
          snake[i].pos[1]=snake[i-1].pos[1];
      }
      snake[0].pos[1]++;

      //to display the body
      for(i=0,j=size-1;i<=j;i++,j--)
      {
          delay(speed);

          gotoxy(snake[i].pos[0],snake[i].pos[1]);
          cout<<snake[i].body;

          gotoxy(snake[j].pos[0],snake[j].pos[1]);
          cout<<snake[j].body;
      }
      chk_food_eaten(snake,level,pos,dir,size,speed);
      gotoxy(pos.x,pos.y);
      cout<<(char)157;
      if(coordinate_check(snake,level,dir,size)=='y')
          return('\0');
    }//end of while
}//end of down()

char right(hss snake[],coordinate level[],int &size,coordinate &pos,int &dir,int &speed)
{
    char move;
    int i,j;
    while(i!=0)
    {
      if(kbhit()!=0)
      {
          move=getch();
          if(move=='w'||move=='s'||move=='W'||move=='S')
             return(move);
      }
      //to interchange the coordinates
      for(i=size+4;i>0;i--)
      {
          snake[i].pos[0]=snake[i-1].pos[0];
          snake[i].pos[1]=snake[i-1].pos[1];
      }
      snake[0].pos[0]++;

      //to display the body
```

```
       for(i=0,j=size-1;i<=j;i++,j--)
      {
          delay(speed);

          gotoxy(snake[i].pos[0],snake[i].pos[1]);
          cout<<snake[i].body;

          gotoxy(snake[j].pos[0],snake[j].pos[1]);
          cout<<snake[j].body;
      }
      chk_food_eaten(snake,level,pos,dir,size,speed);
      gotoxy(pos.x,pos.y);
      cout<<(char)15;
      if(coordinate_check(snake,level,dir,size)=='y')
          return('\0');
    }//end of while
}//end of right()

void game_over()
{
    clrscr();
    gotoxy(30,3);cout<<" /-----\\ \n";
    gotoxy(30,4);cout<<"|         \n";
    gotoxy(30,5);cout<<"| -----\\ \n";
    gotoxy(30,6);cout<<"|       | \n";
    gotoxy(30,7);cout<<" \\-----/ \n";
    gotoxy(30,8);cout<<" /----\\   \n";
    gotoxy(30,9);cout<<"|      |   \n";
    gotoxy(30,10);cout<<"|______|   \n";
    gotoxy(30,11);cout<<"|      |   \n";
    gotoxy(30,12);cout<<"|      |   \n";
    gotoxy(30,13);cout<<"\\      / \n";
    gotoxy(30,14);cout<<"| \\  / | \n";
    gotoxy(30,15);cout<<"|  \\/  | \n";
    gotoxy(30,16);cout<<"|      |  \n";
    gotoxy(30,17);cout<<"|      |  \n";
    gotoxy(30,18);cout<<" /-----\\ \n";
    gotoxy(30,19);cout<<"|         \n";
    gotoxy(30,20);cout<<"|------   \n";
    gotoxy(30,21);cout<<"|         \n";
    gotoxy(30,22);cout<<" \\-----/ \n";
    gotoxy(45,3);cout<<" /-----\\ \n";
    gotoxy(45,4);cout<<"|       |  \n";
    gotoxy(45,5);cout<<"|       |  \n";
    gotoxy(45,6);cout<<"|       |  \n";
    gotoxy(45,7);cout<<" \\-----/ \n";
    gotoxy(45,8);cout<<"\\        / \n";
    gotoxy(45,9);cout<<" \\      /  \n";
    gotoxy(45,10);cout<<"  \\    /   \n";
    gotoxy(45,11);cout<<"   \\  /    \n";
    gotoxy(45,12);cout<<"    \\/     \n";
    gotoxy(45,13);cout<<" /-----\\ \n";
    gotoxy(45,14);cout<<"|         \n";
    gotoxy(45,15);cout<<"|------   \n";
    gotoxy(45,16);cout<<"|         \n";
    gotoxy(45,17);cout<<" \\-----/ \n";
    gotoxy(45,18);cout<<" /-----\\ \n";
    gotoxy(45,19);cout<<"|      |  \n";
    gotoxy(45,20);cout<<"|\\-----/ \n";
    gotoxy(45,21);cout<<"|  \\     \n";
    gotoxy(45,22);cout<<"|    \\/\\  \n";
    getch();
}
```

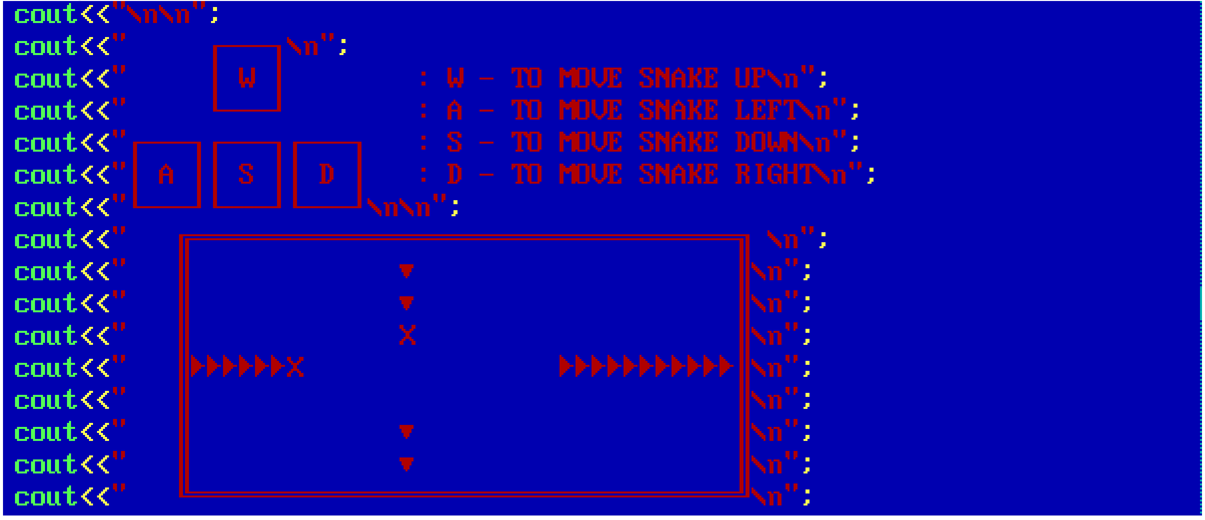
```
void score(hss snake[])
{
    clrscr();
    cout<<"\n\n\n\n\n\n\n\n\n\n\t\t\t\tYOUR SCORE IS : "<<snake[0].food_eaten*10;
    getch();
}

void game_start()
{
    clrscr();
    int i,j,x,op,sign=1,speed=70,dir=0;// x has no use except in the following loop to
                                       //initialize coordinate
    hss snake[150];
    snake[0].food_eaten=0;
    coordinate level[20],pos={40,11};
    level[0].x=0;level[0].y=0;
    char move='d';
    int size=5;//snake size
    for(i=0,x=40;i<150;i++,x--)
    {
      snake[i].pos[0]=x;
      snake[i].pos[1]=11;
    }

    //intitialization of snake body
    for(i=1;i<81;i++)
    {
      if(i%2==0)
      snake[i].body=178;
      else
      snake[i].body=177;
    }
    snake[0].body='x';
    snake[4].body=' ';

    //border making code
    for(i=11,j=4;j<=22;j++)
    {
      gotoxy(i,j);
      cout<<(char)186;
    }
    for(i=70,j=4;j<=22;j++)
    {
      gotoxy(i,j);
      cout<<(char)186;
    }
    for(i=11,j=4;i<=70;i++)
    {
      gotoxy(i,j);
      cout<<(char)205;
    }
    for(i=11,j=22;i<=70;i++)
    {
      gotoxy(i,j);
      cout<<(char)205;
    }
    gotoxy(70,22);cout<<(char)188;
    gotoxy(70,4);cout<<(char)187;
    gotoxy(11,22);cout<<(char)200;
    gotoxy(11,4);cout<<(char)201;
    gotoxy(36,24);cout<<"SCORE :  0";
    pos=food_gen(snake,level,pos,dir);
    while(1!=0)
    {
      if(move=='d'||move=='D')
```

```
            move=right(snake,level,size,pos,dir,speed);
        if(move=='s'||move=='S')
            move=down(snake,level,size,pos,dir,speed);
        if(move=='a'||move=='A')
            move=left(snake,level,size,pos,dir,speed);
        if(move=='w'||move=='W')
            move=up(snake,level,size,pos,dir,speed);
        if(move=='\0')
            break;
    }//end of while
    game_over();
    score(snake);
}//end of game_start()

void snake_menu()
{
    clrscr();
    int op,control=2;
    char body='>';
```



```
cout<<"░░░░░░░░░░░░░░░░░░░░░░░░░░▓█▓█▓█▓█▓█▓█▓░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░\n";
cout<<"░░░░░░░░░░░░░░░░░░░░░░░░░░▓ MAIN MENU ▓░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░\n";
cout<<"░░░░░░░░░░░░░░░░░░░░░░░░░░▓█▓█▓█▓█▓█▓█▓░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░░";
```

```
    cout<<"\n\n\n(1)PLAY\n(2)INSTRUCTIONS\n(3)QUIT";
    cout<<"\n\nEnter your option : ";
    cin>>op;
    switch(op)
    {
        case 1:game_start();
               break;
        case 2:instruction();
               break;
        case 3:game_choose();
               break;
        default:cout<<"\nYou entered wrong choice!";
    }//end of switch()
    snake_menu();
}//end of snake_menu()

void instruction()
{
    clrscr();
    char go_back;
```



```
cout<<"\n\n";
cout<<"      ┌───┐  \n";
cout<<"      │ W │         : W - TO MOVE SNAKE UP\n";
cout<<"      └───┘         : A - TO MOVE SNAKE LEFT\n";
cout<<" ┌───┐ ┌───┐ ┌───┐  : S - TO MOVE SNAKE DOWN\n";
cout<<" │ A │ │ S │ │ D │  : D - TO MOVE SNAKE RIGHT\n";
cout<<" └───┘ └───┘ └───┘\n\n";
cout<<"   ╔════════════════════════════════════╗ \n";
cout<<"   ║             ▼                      ║\n";
cout<<"   ║             ▼                      ║\n";
cout<<"   ║             X                      ║\n";
cout<<"   ║►►►►►►X                ►►►►►►►►►►►║\n";
cout<<"   ║                                    ║\n";
cout<<"   ║             ▼                      ║\n";
cout<<"   ║             ▼                      ║\n";
cout<<"   ╚════════════════════════════════════╝\n";
```

```
    cout<<"\n\nPress 'Q' to go back : ";
    while(go_back!='q')
    {
        go_back=getch();
```

```
       if(go_back=='q'||go_back=='Q')
      {
          cout<<go_back;
          go_back='q';
      }
    }//end of while
}//end of instruction

void snake()
{
    clrscr();
    int load_x,load_y,x,y,i;
    cout<<"                ________ _          ______        __  _________\n";
    cout<<"               (  ______| |\\       |  ____ \\      / / |  _______|  \n";
    cout<<"                  \\ \\    | |\\ \\    | |    | |  / /   | |     \n";
    cout<<"                    \\ \\  | | \\ \\   | |____| |/ /     | |-------|\n";
    cout<<"                      \\ \\| |   \\ \\ | |____| |\\ \\     |  -------|     \n";
    cout<<"                ________\\ \\|     \\ \\| |    | |  \\ \\   | |_______      \n";
    cout<<"               (__________)       \\__|    |_|    \\_\\ | ________|   \n";
    cout<<"\n\n\n\n";
    cout<<"               .     .                      \n";
    cout<<"          .              .                \n";
    cout<<"     .                       .             \n";
    cout<<"       \\    \\        /    /                  \n";
    cout<<"         \\    \\    /    /                  \n";
    cout<<"          \\     \\/     /                  \n";
    cout<<"           \\          /             \n";
    cout<<"            \\        /              \n";
    cout<<"             \\      /                       |                \n";
    cout<<"               \\    \\                   |       |           \n";
    cout<<"                 |    |              |      |                  \n";
    cout<<"                     |    |        |    |      |    |            \n";
    cout<<"                       |    |   |    |            |    |         \n";

    gotoxy(35,12);
    cout<<"LOADING";
    for(load_x=42,load_y=12,x=25,y=10,i=1;i<=27;i++,x++)
    {
      delay(100);
      gotoxy(x,y);
      cout<<(char)222;//or219
      if(i==1)
      {
          gotoxy(load_x,load_y);
          cout<<".";
          load_x+=1;
      }
      if(i%6==0)
      {
          gotoxy(42,12);
          clreol();
          load_x=42;
      }
      if(i%2==0)
      {
          gotoxy(load_x,load_y);
          cout<<".";
          load_x+=1;
      }
      delay(200);
    }//end of for loop
    snake_menu();
    getch();
}
```

```
////////////////////////////////////////////////////////////////////////////////////////////////
///////////////////////////////////////// USER ENTRY FORM //////////////////////////////////////
////////////////////////////////////////////////////////////////////////////////////////////////

class player
{
public:
    struct n
    {
      char fname[11];
      char lname[11];
    }name;
    char username[16];
    char password[17];
    char gender;
    char mobile[11];
    player()
    {
      strcpy(name.fname,"0000000000");
      strcpy(name.lname,"0000000000");
      strcpy(username,"000000000000000");
      strcpy(password,"0000000000000000");
      gender='0';
      strcpy(mobile,"0000000000");
    }
};
long s=sizeof(player);

void sign_in();
void sign_up();


void sign_up()
{
   clrscr();
   int i,j;

   //log in
   for(i=72;i<81;i++)
   {   gotoxy(i,1);cout<<char(205);   }
   gotoxy(72,1);cout<<char(201);gotoxy(73,2);cout<<"SIGN
IN";gotoxy(80,1);cout<<char(187);//left+right top
   for(i=72;i<81;i++)
   {   gotoxy(i,3);cout<<char(205);   }
   gotoxy(72,3);cout<<char(200);gotoxy(80,3);cout<<char(188);//left+right bottom
   gotoxy(72,2);cout<<char(186);gotoxy(80,2);cout<<char(186);//lft most+right most center
vertical line
   //first name
   gotoxy(1,3);cout<<" Name:";
   for(i=1;i<15;i++)
   {   gotoxy(i,4);cout<<char(196);   }
   gotoxy(1,4);cout<<char(218);gotoxy(2,5);cout<<"First";gotoxy(14,4);cout<<char(191);
   for(i=1;i<14;i++)
   {   gotoxy(i,6);cout<<char(196);   }
   gotoxy(1,6);cout<<char(192);gotoxy(14,6);cout<<char(217);
   gotoxy(1,5);cout<<char(179);gotoxy(14,5);cout<<char(179);
   //last name
   for(i=20;i<34;i++)
   {   gotoxy(i,4);cout<<char(196);   }
   gotoxy(20,4);cout<<char(218);gotoxy(21,5);cout<<"Last";gotoxy(33,4);cout<<char(191);
   for(i=20;i<33;i++)
   {   gotoxy(i,6);cout<<char(196);   }
   gotoxy(20,6);cout<<char(192);gotoxy(33,6);cout<<char(217);
   gotoxy(20,5);cout<<char(179);gotoxy(33,5);cout<<char(179);
   gotoxy(1,7);cout<<" Choose your username:";
   for(i=1;i<35;i++)
   {   gotoxy(i,8);cout<<char(196);   }
   gotoxy(1,8);cout<<char(218);gotoxy(12,9);cout<<"eg.
username@gmail.com";gotoxy(34,8);cout<<char(191);
   for(i=1;i<34;i++)
   {   gotoxy(i,10);cout<<char(196);   }
   gotoxy(1,10);cout<<char(192);gotoxy(34,10);cout<<char(217);
   gotoxy(1,9);cout<<char(179);gotoxy(34,9);cout<<char(179)<<" Maximum 15 characters";
   //create password 1,13
   gotoxy(1,11);cout<<" Create password:";
   for(i=1;i<21;i++)
   {   gotoxy(i,12);cout<<char(196);   }
   gotoxy(1,12);cout<<char(218);gotoxy(20,12);cout<<char(191);
   for(i=1;i<21;i++)
   {   gotoxy(i,14);cout<<char(196);   }
   gotoxy(1,14);cout<<char(192);gotoxy(20,14);cout<<char(217);
   gotoxy(1,13);cout<<char(179);gotoxy(20,13);cout<<char(179)<<" Maximum 16 & minimum 5
   characters";
   //gender
   gotoxy(1,15);cout<<" Gender:";
   for(i=1;i<6;i++)
   {   gotoxy(i,16);cout<<char(196);   }
   gotoxy(1,16);cout<<char(218);gotoxy(5,16);cout<<char(191);
   for(i=1;i<6;i++)
   {   gotoxy(i,18);cout<<char(196);   }
   gotoxy(1,18);cout<<char(192);gotoxy(5,18);cout<<char(217);
   gotoxy(1,17);cout<<char(179);gotoxy(5,17);cout<<char(179)<<" (M/F)";
   //mobile phone number
   gotoxy(1,19);cout<<" Mobile phone:";
   for(i=1;i<20;i++)
   {   gotoxy(i,20);cout<<char(196);   }
   gotoxy(1,20);cout<<char(218);gotoxy(2,21);cout<<"+91 ";gotoxy(19,20);cout<<char(191);
   for(i=1;i<20;i++)
   {   gotoxy(i,22);cout<<char(196);   }
   gotoxy(1,22);cout<<char(192);gotoxy(19,22);cout<<char(217);
   gotoxy(1,21);cout<<char(179);gotoxy(19,21);cout<<char(179);
   //create account
   for(i=30;i<48;i++)
   {   gotoxy(i,23);cout<<char(240);   }
   gotoxy(30,23);cout<<char(240);gotoxy(31,24);cout<<(char)240<<"CREATE
   ACCOUNT"<<(char)240;gotoxy(47,23);cout<<char(240);
   for(i=30;i<48;i++)
   {   gotoxy(i,25);cout<<char(240);   }
   gotoxy(30,25);cout<<char(240);gotoxy(47,25);cout<<char(240);
   gotoxy(30,24);cout<<char(240);gotoxy(47,24);cout<<char(240);

   player temp,obj;
   int x=2,y=5;
   char ch;

   while(1)
   {
       gotoxy(x,y);
       if(x==77&&y==2)//sign in
       {
          i=0;
          while(1)
          {
              if(kbhit())
              {
                ch=getch();
                if(ch=='\t')
                {
                    x=2;y=5;
                    break;
                }
                if(ch==13)
                {
                    sign_in();
                }
              }//end of kbhit()
          }//end of inner while
       }//if ends

       gotoxy(x,y);
       if(x==2&&y==5)//fname
       {
          i=0;
          while(1)
          {
              if(kbhit())
              {
                ch=getch();
                if(ch==13||ch=='\t')
                {
                    if(temp.name.fname[2]!='0')//name not to be short
                    {    x=21;y=5;
                        if(i!=0)
                          temp.name.fname[i]='\0';
                    }
                    break;
                }
                if(ch=='\b')
                {
                    if(i>0)
                    {
                      temp.name.fname[i]='0';
                      i--;
                      cout<<"\b \b";
                    }
                }
                else
                {
                      if(i<10)
                      {
                          if(i==0)
                          {
                                clreol();gotoxy(14,5);cout<<char(179); gotoxy(20,5);
                                cout<<char(179)<<"Last";gotoxy(33,5);cout<<char(179);
                                gotoxy(3,5);
                          }
                      temp.name.fname[i]=ch;
                      i++;
                      cout<<ch;
                    }
                }
              }//end of kbhit()
          }//end of inner while
          if(i!=0)
              temp.name.fname[i]='\0';
       }//if ends
```

```
gotoxy(x,y);
if(x==21&&y==5)//lname
{
   i=0;




  while(1)
   {
       if(kbhit())
       {
         ch=getch();
         if(ch==13||ch=='\t')
         {
             if(temp.name.lname[2]!='0')//name not to be short
             {    x=2;y=9;
                 if(i!=0)
                    temp.name.lname[i]='\0';
             }
             break;
         }
         if(ch=='\b')
         {
             if(i>0)
             {
                temp.name.lname[i]='0';
                i--;
                cout<<"\b \b";
             }
         }
         else
         {
             if(i<10)
             {
                if(i==0)
                {     clreol();gotoxy(33,5);cout<<char(179);gotoxy(22,5);    }
                temp.name.lname[i]=ch;
                i++;
                cout<<ch;
             }
         }
       }//end of kbhit()

   }//end of inner while
   if(i!=0)
       temp.name.lname[i]='\0';
}//if ends

gotoxy(x,y);
if(x==2&&y==9)//username
{
   i=0;
   while(1)
   {
       if(kbhit())
       {
         ch=getch();
         if(ch==13||ch=='\t')
         {
             if(temp.username[2]!='0')//username not to be short
             {    x=2;y=13;
```

```
                  if(i!=0)
                  {
                    cout<<"@gmail.com";
                    temp.username[i]='\0';
                  }
                  fstream f;
                  f.open("player.txt",ios::in|ios::binary);
                  while(f.read((char*)&obj,s))
                  {
                     if(strcmp(obj.username,temp.username)==0)
                     {
                          gotoxy(3,9);clreol();gotoxy(34,9);cout<<char(179);
                          gotoxy(35,9);cout<<" Username already exists!!!!!!!";
                          x=2;y=9;break;
                     }
                     else
                     {
                          clreol();gotoxy(34,9);cout<<char(179);gotoxy(35,9);
                          cout<<" Maximum 15 characters";
                     }
                  }
                  f.close();
              }
              break;
          }
          if(ch=='\b')
          {
              if(i>0)
              {
                temp.username[i]='0';
                i--;
                cout<<"\b \b";
              }
          }
          else
          {
              if(i<15)
              {
                if(i==0)
                {     clreol();gotoxy(34,9);cout<<char(179);gotoxy(35,9);
                       cout<<" Maximum 15 characters";gotoxy(3,9);
                }
                temp.username[i]=ch;
                i++;
                cout<<ch;
              }
          }
       }//end of kbhit()

   }//end of inner while
   if(i!=0)
       temp.username[i]='\0';
}//if ends

gotoxy(x,y);
if(x==2&&y==13)//password
{
   i=0;
   while(1)
   {
       if(kbhit())
       {
          ch=getch();
          if(ch==13||ch=='\t')
          {
              if(temp.password[5]!='0')//password not to be short
              {    x=3;y=17;
                  if(i!=0)
                    temp.password[i]='\0';
              }
              break;
          }
          if(ch=='\b')
          {
              if(i>0)
              {
                temp.password[i]='0';
                i--;
                cout<<"\b \b";
              }
          }
          else
          {
              if(i<16)
              {
                if(i==0)
                {     clreol();gotoxy(20,13);cout<<char(179)<<" Maximum 16 & minimum 5
                       character";;gotoxy(3,13);
                 }
                temp.password[i]=ch;
                cout<<"*";
                i++;
              }
          }
       }//end of kbhit()

   }//end of inline while
   if(i!=0)
       temp.password[i]='\0';
}//if ends

gotoxy(x,y);
if(x==3&&y==17)//gender
{
   i=0;
   while(ch!='m'||ch!='f'||ch!='M'||ch!='F')
   {
       if(kbhit())
       {
          ch=getch();
          if(ch==13||ch=='\t')
          {
              if(temp.gender!='0')
              {    x=6;y=21;    }
              break;
          }
          if(ch=='\b')
          {
              if(i>0)
              {
                temp.gender='0';
                i--;
                cout<<"\b \b";
              }
          }
          if( (ch=='m'||ch=='f'||ch=='M'||ch=='F')&&(i<1) )
          {
              if(ch<65||ch>90)
                ch-=32;
              cout<<ch;
             temp.gender=ch;
             i++;
         }
       }//end of kbhit()
   }//end of inner while
}//if ends

gotoxy(x,y);
if(x==6&&y==21)//mobile no.
{
   i=0;
   while(1)
   {
       if(kbhit())
       {
         ch=getch();
         if(ch==13||ch=='\t')
         {
             if(temp.mobile[9]!='0')//name not to be short
             {    x=38;y=24;
                 if(i!=0)
                     temp.mobile[i]='\0';
             }
             break;
         }
         if(ch=='\b')
         {
             if(i>0)
             {
               temp.mobile[i]='0';
               i--;
               cout<<"\b \b";
             }
         }
         if(ch<=58&&ch>=48&&i<10)
         {
             if(i==0)
             {     clreol();gotoxy(19,21);cout<<char(179);gotoxy(7,21);    }
             temp.mobile[i]=ch;
             i++;
             cout<<ch;
         }
       }//end of kbhit()

   }//end of inner while
   if(i!=0)
       temp.mobile[i]='\0';
}//if ends

gotoxy(x,y);
if(x==38&&y==24)//create account
{
   while(1)
   {
       if(kbhit())
       {
         ch=getch();
         if(ch=='\t')
         {
            x=77;y=2;
            break;
         }
         if(ch==13)
         {
            fstream f;
```

```
                    f.open("player.txt",ios::out|ios::in|ios::binary);
                    if(f.fail())
                    {
                         clrscr();
                         cerr<<"\nFile got corrupt!!!";
                         getch();
                         exit(1);
                    }
                    f.seekp(0,ios::end);
                    f.write((char*)&temp,s);
                    f.close();
                    sign_in();
                }

              }//end of kbhit()
          }//end of inner while
       }//if ends
   }//end of outer while
}//end of sign_up()

void sign_in()
{
   clrscr();
   int i;
   player temp,obj;

   //log in
   for(i=70;i<81;i++)
   {   gotoxy(i,1);cout<<char(205);   }
   gotoxy(70,1);cout<<char(201);gotoxy(72,2);cout<<"SIGN UP";
   gotoxy(80,1);cout<<char(187);//left+right top
   for(i=70;i<81;i++)
   {   gotoxy(i,3);cout<<char(205);   }
   gotoxy(70,3);cout<<char(200);gotoxy(80,3);cout<<char(188);//left+right bottom
   gotoxy(70,2);cout<<char(186);gotoxy(80,2);cout<<char(186);//lft most+right most center
                                                              //vertical line

   //first name
   gotoxy(25,9);cout<<" USERNAME:";
   for(i=25;i<50;i++)
   {   gotoxy(i,10);cout<<char(196);   }
   gotoxy(25,10);cout<<char(218);gotoxy(26,11);cout<<"username";gotoxy(50,10);cout<<char(191);
   for(i=25;i<50;i++)
   {   gotoxy(i,12);cout<<char(196);   }
   gotoxy(25,12);cout<<char(192);gotoxy(50,12);cout<<char(217);
   gotoxy(25,11);cout<<char(179);gotoxy(50,11);cout<<char(179);

   gotoxy(25,13);cout<<" PASSWORD:";
   for(i=25;i<50;i++)
   {   gotoxy(i,14);cout<<char(196);   }
   gotoxy(25,14);cout<<char(218);gotoxy(27,15);gotoxy(50,14);cout<<char(191);
   for(i=25;i<50;i++)
   {   gotoxy(i,16);cout<<char(196);   }
   gotoxy(25,16);cout<<char(192);gotoxy(50,16);cout<<char(217);
   gotoxy(25,15);cout<<char(179);gotoxy(50,15);cout<<char(179);

   for(i=33;i<43;i++)
   {   gotoxy(i,18);cout<<char(205);   }
   gotoxy(33,18);cout<<char(201);gotoxy(35,19);cout<<"SIGN IN";
   gotoxy(43,18);cout<<char(187);  //left+right top
   for(i=33;i<43;i++)
   {   gotoxy(i,20);cout<<char(205);   }
   gotoxy(33,20);cout<<char(200);gotoxy(43,20);cout<<char(188);//left+right bottom
   gotoxy(33,19);cout<<char(186);gotoxy(43,19);cout<<char(186);//lft most+right most center
                                                                //vertical line
```

```
int x=76,y=2;
char ch;
while(1!=0)
{
    gotoxy(x,y);
    if(x==76&&y==2)//sign up
    {
      while(1)
      {
          if(kbhit())
          {
             ch=getch();
             if(ch=='\t')
             {
                 x=26;y=11;
                 break;
             }
             if(ch==13)
             {
                 //f.close();
                 sign_up();
             }
          }//end of kbhit()
      }//end of inner while
    }//if ends

    gotoxy(x,y);
    if(x==26&&y==11)//username
    {
      i=0;
      while(1)
      {
          if(kbhit())
          {
             ch=getch();
             if(ch==13||ch=='\t')
             {
                 if(temp.username[2]!='0')//username not to be short
                 {    x=27;y=15;
                     if(i!=0)
                     {
                        cout<<"@gmail.com";
                        temp.username[i]='\0';
                     }
                 }
                 break;
             }
             if(ch=='\b')
             {
                 if(i>0)
                 {
                    temp.username[i]='0';
                    i--;
                    cout<<"\b \b";
                 }
             }
             else
             {
                 if(i<15)
                 {
                    if(i==0)
                    {     clreol();gotoxy(25,11);cout<<char(179);gotoxy(50,11);
                          cout<<char(179);gotoxy(27,11);
                    }
                    temp.username[i]=ch;
```

```
                i++;
                cout<<ch;
             }//end of inner if
         }
       }//end of kbhit()

   }//end of inner while
   if(i!=0)
       temp.username[i]='\0';
}//if ends

gotoxy(x,y);
if(x==27&&y==15)//password
{
   i=0;
   while(1)
   {
       if(kbhit())
       {
         ch=getch();
         if(ch==13||ch=='\t')
         {
             if(temp.password[5]!='0')//password not to be short
             {    x=39;y=19;
                 if(i!=0)
                   temp.password[i]='\0';
             }
             break;
         }
         if(ch=='\b')
         {
             if(i>0)
             {
                temp.password[i]='0';
                i--;
                cout<<"\b \b";
             }
         }
         else
         {
             if(i<16)
             {
                if(i==0)
                {     clreol();gotoxy(25,15);cout<<char(179);gotoxy(50,15);
                      cout<<char(179);gotoxy(27,15);
                }
                temp.password[i]=ch;
                cout<<"*";
                i++;
             }
         }
       }//end of kbhit()

   }//end of inner while
   if(i!=0)
       temp.password[i]='\0';
}//if ends
gotoxy(x,y);
if(x==39&&y==19)//sign in
{
   while(1)
   {
       if(kbhit())
       {
         ch=getch();
```

```
                if(ch=='\t')
                {
                    x=76;y=2;
                    break;
                }
                if(ch==13)
                {
                    fstream f;
                    f.open("player.txt",ios::out|ios::in|ios::binary);
                    if(f.fail())
                    {
                         clrscr();
                         cerr<<"\nFile got corrupt!!!";
                         getch();
                         exit(1);
                    }
                    while(f.read((char*)&obj,s))
                    {
                         if((strcmp(temp.username,obj.username)==0)&&(strcmp(temp.password,
                                                                    obj.password)==0))
                         {
                                f.close();
                                game_choose();
                         }
                    }
                    f.close();
                 }

              }//end of kbhit()
          }//end of inner while
        }//if ends
    }//end of outer while
}

void main()
{
    sign_in();
}
```