**Ludo**

****

Session: 2021 – 2025

**Submitted by:**

**Hamza Tariq 2021-CS-74**

**Supervised by:**

**Ma’am Maida Shahid**

Department of Computer Science

**University of Engineering and Technology**

**Table of Content**

**Topics Page No**

1. Abstract ……………………………………………………….................. 03

2. Wireframes ………………………………………………………………. 04

4.1 Start Menu ………………………………………………………….... 04

4.2 Interface …………………………………………………................... 04

3. Data Structure …………………………………………...…..................... 05

4. Functions Prototypes ………………………………………………........... 06

5. Complete Code ………………………………………………................... 08

**Lahore, Pakistan**

**Ludo**

1. **Abstract:**

The name of the game is Ludo. In this game there are two users. Every time a player gets a turn after the first one .Players have to avoid other player’s beads .In case two beads of opposite players are on same position one will get knocked out. A random number is generated by dice. Player have to choose which bead he is going to move.

**Game characters description:**

There are eight beads in my Ludo game. Each player got 4 beats.

**Rules and Interaction:**

Following are the rules:

* Player need to get a six to start a bead moving on the board.
* There are particular positions where a bead can kill other bead.
* Both player try to kill other’s player beads and get his 4 beat clear before other.

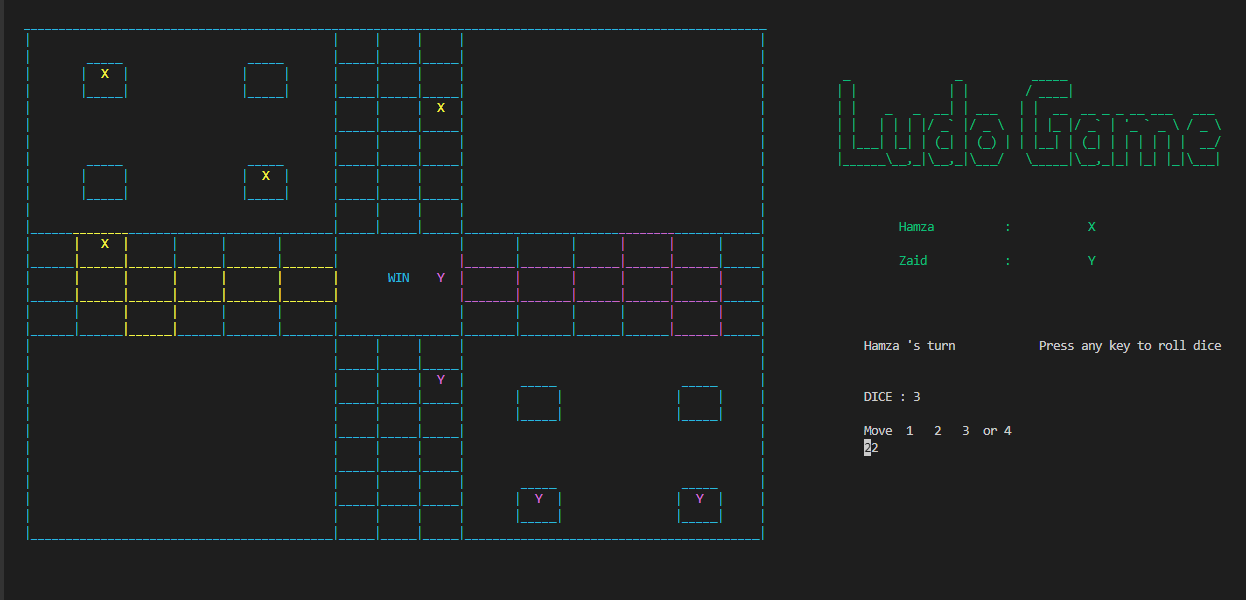
**Goal of the game:**

The goal of the game is to clear 4 beads before the other player.

1. **Wireframes:**

****

**2.1. Start Menu**



**2.2. Interface**

**3. Data Types and structures:**

char n1 = '1';

char n2 = '2';

char n3 = '3';

char n4 = '4';

int xn1 = 3;

int yn1 = 11;

int xn2 = 9;

int yn2 = 11;

int xn3 = 3;

int yn3 = 34;

int xn4 = 9;

int yn4 = 34;

// for player 2

char n5 = '5';

char n6 = '6';

char n7 = '7';

char n8 = '8';

int xn5 = 22;

int yn5 = 94;

int xn6 = 22;

int yn6 = 73;

int xn7 = 28;

int yn7 = 94;

int xn8 = 28;

int yn8 = 73;

char maze[31][106];

int rowSize = sizeof(maze) / sizeof(maze[0]);

int colSize = sizeof(maze[0]) / sizeof(maze[0][0]);

string path;

int n1score = 1;

int n2score = 0;

int n3score = 0;

    int n4score = 0;

    maze[xn1][yn1] = n1;

    maze[xn2][yn2] = n2;

    maze[xn3][yn3] = n3;

    maze[xn4][yn4] = n4;

    int n5score = 1;

    int n6score = 0;

    int n7score = 0;

    int n8score = 0;

    maze[xn5][yn5] = n5;

    maze[xn6][yn6] = n6;

    maze[xn7][yn7] = n7;

    maze[xn8][yn8] = n8;

    string player1, player2;

    char one = '1', two = '2';

    int num = 0;

**4. Function Prototypes:**

void header();

void logo(string player1, string player2);

void start(string &player1, string &player2, bool &gameRunning);

void newfunc(int &n1score, int &n2score, int &n3score, int &n4score, string op, int dice\_roll, int rowSize, int colSize);

void player2beads(int &n5score, int &n6score, int &n7score, int &n8score, string op, int dice\_roll, int rowSize, int colSize);

void colouringtheme(int rowSize, int colSize);

void colourplayer1beads(int rowSize, int colSize);

void colourplayer2beads(int rowSize, int colSize);

void samescore(int &n1score, int &n2score, int &n3score, int &n4score, int &n5score, int &n6score, int &n7score, int &n8score, char player);

void checkwin(bool &gameRunning, int n1score, int n2score, int n3score, int n4score, int n5score, int n6score, int n7score, int n8score, string player1, string player2);

void gotoxy(int x, int y);

void printMaze(char maze[31][106], int colSize, int rowSize);

int random();

int scorefunc(int &score, int dice\_number);

void load(char maze[][106], int rowSize, int colSize, string path);

**5. Complete Code**

#include <iostream>

#include <fstream>

#include <windows.h>

#include <conio.h>

#include <ctime>

using namespace std;

void header();

void logo(string player1, string player2);

void start(string &player1, string &player2, bool &gameRunning);

void newfunc(int &n1score, int &n2score, int &n3score, int &n4score, string op, int dice\_roll, int rowSize, int colSize);

void player2beads(int &n5score, int &n6score, int &n7score, int &n8score, string op, int dice\_roll, int rowSize, int colSize);

void colouringtheme(int rowSize, int colSize);

void colourplayer1beads(int rowSize, int colSize);

void colourplayer2beads(int rowSize, int colSize);

void samescore(int &n1score, int &n2score, int &n3score, int &n4score, int &n5score, int &n6score, int &n7score, int &n8score, char player);

void checkwin(bool &gameRunning, int n1score, int n2score, int n3score, int n4score, int n5score, int n6score, int n7score, int n8score, string player1, string player2);

void gotoxy(int x, int y);

void printMaze(char maze[31][106], int colSize, int rowSize);

int random();

int scorefunc(int &score, int dice\_number);

void load(char maze[][106], int rowSize, int colSize, string path);

HANDLE h = GetStdHandle(STD\_OUTPUT\_HANDLE);

// for player 1

char n1 = '1';

char n2 = '2';

char n3 = '3';

char n4 = '4';

int xn1 = 3;

int yn1 = 11;

int xn2 = 9;

int yn2 = 11;

int xn3 = 3;

int yn3 = 34;

int xn4 = 9;

int yn4 = 34;

// for player 2

char n5 = '5';

char n6 = '6';

char n7 = '7';

char n8 = '8';

int xn5 = 22;

int yn5 = 94;

int xn6 = 22;

int yn6 = 73;

int xn7 = 28;

int yn7 = 94;

int xn8 = 28;

int yn8 = 73;

char maze[31][106];

main() // Main Function

{

int rowSize = sizeof(maze) / sizeof(maze[0]);

int colSize = sizeof(maze[0]) / sizeof(maze[0][0]);

string path = "game.txt";

load(maze, rowSize, colSize, path);

srand(time(0));

bool gameRunning = true;

int n1score = 1;

int n2score = 0;

int n3score = 0;

int n4score = 0;

maze[xn1][yn1] = n1;

maze[xn2][yn2] = n2;

maze[xn3][yn3] = n3;

maze[xn4][yn4] = n4;

int n5score = 1;

int n6score = 0;

int n7score = 0;

int n8score = 0;

maze[xn5][yn5] = n5;

maze[xn6][yn6] = n6;

maze[xn7][yn7] = n7;

maze[xn8][yn8] = n8;

string player1, player2;

char one = '1', two = '2';

int num = 0;

system("CLS");

header();

start(player1, player2, gameRunning);

system("CLS");

printMaze(maze, colSize, rowSize);

colouringtheme(rowSize, colSize);

logo(player1, player2);

while (gameRunning)

{

string op;

gotoxy(120, 19);

cout << player1 << " 's turn" << endl;

gotoxy(140, 19);

cout << " Press any key to roll dice" << endl;

gotoxy(120, 20);

// getch();

int dice\_roll = random();

gotoxy(120, 22);

cout << "DICE : " << dice\_roll << endl;

gotoxy(120, 24);

cout << "Move 1 2 3 or 4 " << endl;

gotoxy(120, 25);

cin >> op;

while (op != "1" && op != "2" && op != "3" && op != "4")

{

gotoxy(120, 25);

cin >> op;

}

newfunc(n1score, n2score, n3score, n4score, op, dice\_roll, rowSize, colSize);

samescore(n1score, n2score, n3score, n4score, n5score, n6score, n7score, n8score, one);

player2beads(n5score, n6score, n7score, n8score, op, num, rowSize, colSize);

colourplayer2beads(rowSize, colSize);

colourplayer1beads(rowSize, colSize);

checkwin(gameRunning, n1score, n2score, n3score, n4score, n5score, n6score, n7score, n8score, player1, player2);

bool flag1 = true;

if (dice\_roll == 6)

{

flag1 = false;

}

while (flag1 == false)

{

int y = 25;

dice\_roll = random();

gotoxy(120, 22);

cout << "DICE : " << dice\_roll << endl;

if (dice\_roll != 6)

{

flag1 = true;

}

gotoxy(120, y);

cin >> op;

while (op != "1" && op != "2" && op != "3" && op != "4")

{

gotoxy(120, 25);

cin >> op;

}

newfunc(n1score, n2score, n3score, n4score, op, dice\_roll, rowSize, colSize);

samescore(n1score, n2score, n3score, n4score, n5score, n6score, n7score, n8score, one);

colourplayer2beads(rowSize, colSize);

player2beads(n5score, n6score, n7score, n8score, op, num, rowSize, colSize);

colourplayer1beads(rowSize, colSize);

checkwin(gameRunning, n1score, n2score, n3score, n4score, n5score, n6score, n7score, n8score, player1, player2);

}

gotoxy(120, 19);

cout << player2 << " 's turn" << endl;

gotoxy(140, 19);

cout << " Press any key to roll dice" << endl;

// getch();

dice\_roll = random();

gotoxy(120, 22);

cout << "DICE : " << dice\_roll << endl;

gotoxy(120, 24);

cout << "Move 1 2 3 or 4 " << endl;

gotoxy(120, 25);

cin >> op;

while (op != "1" && op != "2" && op != "3" && op != "4")

{

gotoxy(120, 25);

cin >> op;

}

player2beads(n5score, n6score, n7score, n8score, op, dice\_roll, rowSize, colSize);

samescore(n1score, n2score, n3score, n4score, n5score, n6score, n7score, n8score, two);

newfunc(n1score, n2score, n3score, n4score, op, num, rowSize, colSize);

colourplayer1beads(rowSize, colSize);

colourplayer2beads(rowSize, colSize);

checkwin(gameRunning, n1score, n2score, n3score, n4score, n5score, n6score, n7score, n8score, player1, player2);

flag1 = true;

if (dice\_roll == 6)

{

flag1 = false;

}

while (flag1 == false)

{

int y = 25;

dice\_roll = random();

gotoxy(120, 22);

cout << "DICE : " << dice\_roll << endl;

if (dice\_roll != 6)

{

flag1 = true;

}

gotoxy(120, y);

cin >> op;

while (op != "1" && op != "2" && op != "3" && op != "4")

{

gotoxy(120, 25);

cin >> op;

}

player2beads(n5score, n6score, n7score, n8score, op, dice\_roll, rowSize, colSize);

samescore(n1score, n2score, n3score, n4score, n5score, n6score, n7score, n8score, two);

newfunc(n1score, n2score, n3score, n4score, op, num, rowSize, colSize);

colourplayer1beads(rowSize, colSize);

colourplayer2beads(rowSize, colSize);

checkwin(gameRunning, n1score, n2score, n3score, n4score, n5score, n6score, n7score, n8score, player1, player2);

}

if (GetAsyncKeyState(VK\_ESCAPE))

{

gameRunning = false; // Stop the game

}

}

gotoxy(120, 32);

cout << "Press any key to continue" << endl;

gotoxy(120, 33);

getch();

}

void gotoxy(int x, int y)

{

COORD coordinates;

coordinates.X = x;

coordinates.Y = y;

SetConsoleCursorPosition(GetStdHandle(STD\_OUTPUT\_HANDLE), coordinates);

}

void start(string &player1, string &player2, bool &gameRunning)

{

string op;

gotoxy(10, 4);

cout << " W E L C O M E " << endl;

gotoxy(10, 6);

cout << "1.Start Game " << endl;

gotoxy(10, 7);

cout << "2.Exit" << endl;

gotoxy(10, 8);

cin >> op;

if (op == "1")

{

gotoxy(10, 10);

cout << "Enter Player 1 Name" << endl;

gotoxy(40, 10);

cout << "Enter Player 2 Name" << endl;

gotoxy(10, 11);

cin.ignore();

getline(cin, player1);

gotoxy(40, 11);

getline(cin, player2);

}

else if (op == "2")

{

gameRunning = false;

}

}

void logo(string player1, string player2)

{

header();

SetConsoleTextAttribute(h, 2);

gotoxy(125, 12);

cout << player1 << "\t\t"

<< endl;

gotoxy(140, 12);

cout << ":"

<< "\t\t"

<< "X" << endl;

gotoxy(125, 14);

cout << player2 << "\t\t"

<< endl;

gotoxy(140, 14);

cout << ":"

<< "\t\t"

<< "Y" << endl;

SetConsoleTextAttribute(h, 7);

}

void header()

{

SetConsoleTextAttribute(h, 2);

gotoxy(115, 3);

cout << " \_ \_ \_\_\_\_\_ " << endl;

gotoxy(115, 4);

cout << " | | | | / \_\_\_\_|" << endl;

gotoxy(115, 5);

cout << " | | \_ \_ \_\_| | \_\_\_ | | \_\_ \_\_ \_ \_ \_\_ \_\_\_ \_\_\_" << endl;

gotoxy(115, 6);

cout << " | | | | | |/ \_` |/ \_ \\ | | |\_ |/ \_` | '\_ ` \_ \\ / \_ \\ " << endl;

gotoxy(115, 7);

cout << " | |\_\_\_| |\_| | (\_| | (\_) | | |\_\_| | (\_| | | | | | | \_\_/" << endl;

gotoxy(115, 8);

cout << " |\_\_\_\_\_\_\\\_\_,\_|\\\_\_,\_|\\\_\_\_/ \\\_\_\_\_\_|\\\_\_,\_|\_| |\_| |\_|\\\_\_\_|" << endl;

SetConsoleTextAttribute(h, 7);

}

void printMaze(char maze[31][106], int colSize, int rowSize)

{

SetConsoleTextAttribute(h, 11);

for (int x = 0; x < rowSize; x++)

{

for (int y = 0; y < colSize; y++)

{

cout << maze[x][y];

}

cout << endl;

}

SetConsoleTextAttribute(h, 7);

}

int random()

{

// Code that generates the Random Number

int num;

num = rand() % 6 + 1;

return num;

}

int scorefunc(int &score, int dice\_number)

{

// condition for start to get a six

if (score == 0)

{

if (dice\_number != 6)

{

dice\_number = 0;

}

else if (dice\_number == 6)

{

return 1;

}

}

score = score + dice\_number;

return score;

}

void checkwin(bool &gameRunning, int n1score, int n2score, int n3score, int n4score, int n5score, int n6score, int n7score, int n8score, string player1, string player2)

{

if (n1score == 57 && n2score == 57 && n3score == 57 && n4score == 57)

{

gotoxy(120, 29);

cout << "Congratulations " << player1 << " You Won !!" << endl;

gameRunning = false;

}

else if (n5score == 57 && n6score == 57 && n7score == 57 && n8score == 57)

{

gotoxy(120, 29);

cout << "Congratulations " << player2 << " You Won !!" << endl;

gameRunning = false;

}

}

void load(char maze[][106], int rowSize, int colSize, string path)

{

fstream myFile;

string record;

myFile.open(path, ios::in);

for (int row = 0; row < rowSize; row++)

{

getline(myFile, record);

for (int col = 0; col < colSize; col++)

{

maze[row][col] = record[col];

// myFile >> maze[row][col];

}

};

}

void colouringtheme(int rowSize, int colSize)

{

for (int i = 12; i < 20; i++)

{

SetConsoleTextAttribute(h, 14);

gotoxy(14, i);

cout << maze[i][14];

}

SetConsoleTextAttribute(h, 7);

for (int i = 12; i < 15; i++)

{

SetConsoleTextAttribute(h, 14);

gotoxy(7, i);

cout << maze[i][7];

}

SetConsoleTextAttribute(h, 7);

for (int i = 16; i < 19; i++)

{

SetConsoleTextAttribute(h, 14);

gotoxy(21, i);

cout << maze[i][21];

}

SetConsoleTextAttribute(h, 7);

for (int i = 7; i < 49; i++)

{

if (i == 21)

{

continue;

}

if (i == 28)

{

continue;

}

if (i == 36)

{

continue;

}

if (i == 44)

{

continue;

}

SetConsoleTextAttribute(h, 14);

gotoxy(i, 14);

cout << maze[14][i];

}

SetConsoleTextAttribute(h, 7);

for (int i = 7; i < 50; i++)

{

SetConsoleTextAttribute(h, 14);

gotoxy(i, 15);

cout << maze[15][i];

}

SetConsoleTextAttribute(h, 7);

for (int i = 7; i < 50; i++)

{

SetConsoleTextAttribute(h, 14);

gotoxy(i, 15);

cout << maze[15][i];

}

SetConsoleTextAttribute(h, 7);

for (int i = 7; i < 50; i++)

{

SetConsoleTextAttribute(h, 14);

gotoxy(i, 16);

cout << maze[16][i];

}

SetConsoleTextAttribute(h, 7);

for (int i = 7; i < 14; i++)

{

SetConsoleTextAttribute(h, 14);

gotoxy(i, 12);

cout << maze[12][i];

}

SetConsoleTextAttribute(h, 7);

for (int i = 14; i < 21; i++)

{

SetConsoleTextAttribute(h, 14);

gotoxy(i, 18);

cout << maze[18][i];

}

SetConsoleTextAttribute(h, 7);

// for player 2

for (int i = 92; i < 100; i++)

{

SetConsoleTextAttribute(h, 5);

gotoxy(i, 18);

cout << maze[18][i];

}

SetConsoleTextAttribute(h, 7);

for (int i = 92; i < 100; i++)

{

SetConsoleTextAttribute(h, 5);

gotoxy(i, 17);

cout << maze[17][i];

}

SetConsoleTextAttribute(h, 7);

for (int i = 62; i < 100; i++)

{

SetConsoleTextAttribute(h, 5);

gotoxy(i, 16);

cout << maze[16][i];

}

SetConsoleTextAttribute(h, 7);

for (int i = 62; i < 100; i++)

{

SetConsoleTextAttribute(h, 5);

gotoxy(i, 16);

cout << maze[16][i];

}

SetConsoleTextAttribute(h, 7);

for (int i = 62; i < 100; i++)

{

SetConsoleTextAttribute(h, 5);

gotoxy(i, 15);

cout << maze[15][i];

}

SetConsoleTextAttribute(h, 7);

for (int i = 62; i < 100; i++)

{

if (i == 70)

{

continue;

}

if (i == 78)

{

continue;

}

if (i == 85)

{

continue;

}

if (i == 92)

{

continue;

}

if (i == 99)

{

continue;

}

SetConsoleTextAttribute(h, 5);

gotoxy(i, 14);

cout << maze[14][i];

}

SetConsoleTextAttribute(h, 7);

for (int i = 85; i < 93; i++)

{

SetConsoleTextAttribute(h, 5);

gotoxy(i, 14);

cout << maze[14][i];

}

SetConsoleTextAttribute(h, 7);

for (int i = 85; i < 93; i++)

{

SetConsoleTextAttribute(h, 5);

gotoxy(i, 13);

cout << maze[13][i];

}

SetConsoleTextAttribute(h, 7);

for (int i = 85; i < 93; i++)

{

SetConsoleTextAttribute(h, 5);

gotoxy(i, 12);

cout << maze[12][i];

}

SetConsoleTextAttribute(h, 7);

}

void samescore(int &n1score, int &n2score, int &n3score, int &n4score, int &n5score, int &n6score, int &n7score, int &n8score, char player)

{

bool flagx = false;

if (player == '1')

{

if (n5score < 26 && n5score > 0)

{

n5score = n5score + 26;

if (n1score == n5score || n2score == n5score || n3score == n5score || n4score == n5score)

{

if (n5score != 27 && n5score != 35 && n5score != 40 && n5score != 48)

{

cout << "1" << endl;

n5score = 0;

flagx = true;

}

}

n5score = n5score - 26;

}

else if (n5score > 26 && n5score < 52)

{

n5score = n5score - 26;

if (n1score == n5score || n2score == n5score || n3score == n5score || n4score == n5score)

{

if (n5score != 1 && n5score != 9 && n5score != 14 && n5score != 22)

{

cout << "2" << endl;

n5score = 0;

flagx = true;

}

}

n5score = n5score + 26;

}

if (n6score < 26 && n6score > 0)

{

n6score = n6score + 26;

if (n1score == n6score || n2score == n6score || n3score == n6score || n4score == n6score)

{

if (n6score != 27 && n6score != 35 && n6score != 40 && n6score != 48)

{

cout << "3" << endl;

n6score = 0;

flagx = true;

}

}

n6score = n6score - 26;

}

else if (n6score > 26 && n6score < 52)

{

n6score = n6score - 26;

if (n1score == n6score || n2score == n6score || n3score == n6score || n4score == n6score)

{

if (n6score != 1 && n6score != 9 && n6score != 14 && n6score != 22)

{

cout << "4" << endl;

n6score = 0;

flagx = true;

}

}

n6score = n6score + 26;

}

if (n7score < 26 && n7score > 0)

{

n7score = n7score + 26;

if (n1score == n7score || n2score == n7score || n3score == n7score || n4score == n7score)

{

if (n7score != 27 && n7score != 35 && n7score != 40 && n7score != 48)

{

cout << "5" << endl;

n7score = 0;

flagx = true;

}

}

n7score = n7score - 26;

}

else if (n7score > 26 && n7score < 52)

{

n7score = n7score - 26;

if (n1score == n7score || n2score == n7score || n3score == n7score || n4score == n7score)

{

if (n7score != 1 && n7score != 9 && n7score != 14 && n7score != 22)

{

cout << "6" << endl;

n7score = 0;

flagx = true;

}

}

n7score = n7score + 26;

}

if (n8score < 26 && n8score > 0)

{

n8score = n8score + 26;

if (n1score == n8score || n2score == n8score || n3score == n8score || n4score == n8score)

{

if (n8score != 27 && n8score != 35 && n8score != 40 && n8score != 48)

{

cout << "7" << endl;

n8score = 0;

flagx = true;

}

}

n8score = n8score - 26;

}

else if (n8score > 26 && n8score < 52)

{

n8score = n8score - 26;

if (n1score == n8score || n2score == n8score || n3score == n8score || n4score == n8score)

{

if (n8score != 1 && n8score != 9 && n8score != 14 && n8score != 22)

{

cout << "8" << endl;

n8score = 0;

flagx = true;

}

}

n8score = n8score + 26;

}

if (flagx == true)

{

if (n5score < 0 || n5score == 26)

{

n5score = 0;

}

else if (n6score < 0 || n6score == 26)

{

n6score = 0;

}

else if (n7score < 0 || n7score == 26)

{

n7score = 0;

}

else if (n8score < 0 || n8score == 26)

{

n8score = 0;

}

}

flagx = false;

}

else if (player == '2')

{

if (n1score < 26 && n1score > 0)

{

n1score = n1score + 26;

if (n5score == n1score || n6score == n1score || n7score == n1score || n8score == n1score)

{

if (n1score != 27 && n1score != 35 && n1score != 40 && n1score != 48)

{

cout << "9" << endl;

n1score = 0;

flagx = true;

}

}

n1score = n1score - 26;

}

else if (n1score > 26 && n1score < 52)

{

n1score = n1score - 26;

if (n5score == n1score || n6score == n1score || n7score == n1score || n8score == n1score)

{

if (n1score != 1 && n1score != 9 && n1score != 14 && n1score != 22)

{

n1score = 0;

flagx = true;

}

}

n1score = n1score + 26;

}

if (n2score < 26 && n2score > 0)

{

n2score = n2score + 26;

if (n5score == n2score || n6score == n2score || n7score == n2score || n8score == n2score)

{

if (n2score != 27 && n2score != 35 && n2score != 40 && n2score != 48)

{

n2score = 0;

flagx = true;

}

}

n2score = n2score - 26;

}

else if (n2score > 26 && n2score < 52)

{

n2score = n2score - 26;

if (n5score == n2score || n6score == n2score || n7score == n2score || n8score == n2score)

{

if (n2score != 1 && n2score != 9 && n2score != 14 && n2score != 22)

{

n2score = 0;

flagx = true;

}

}

n2score = n2score + 26;

}

if (n3score < 26 && n3score > 0)

{

n3score = n3score + 26;

if (n5score == n3score || n6score == n3score || n7score == n3score || n8score == n3score)

{

if (n3score != 27 && n3score != 35 && n3score != 40 && n3score != 48)

{

n3score = 0;

flagx = true;

}

}

n3score = n3score - 26;

}

else if (n3score > 26 && n3score < 52)

{

n3score = n3score - 26;

if (n5score == n3score || n6score == n3score || n7score == n3score || n8score == n3score)

{

if (n3score != 1 && n3score != 9 && n3score != 14 && n3score != 22)

{

n3score = 0;

flagx = true;

}

}

n3score = n3score + 26;

}

if (n4score < 26 && n4score > 0)

{

n4score = n4score + 26;

if (n5score == n4score || n6score == n4score || n7score == n4score || n8score == n4score)

{

if (n4score != 27 && n4score != 35 && n4score != 40 && n4score != 48)

{

n4score = 0;

flagx = true;

}

}

n4score = n4score - 26;

}

else if (n4score > 26 && n4score < 52)

{

n4score = n4score - 26;

if (n5score == n4score || n6score == n4score || n7score == n4score || n8score == n4score)

{

if (n4score != 1 && n4score != 9 && n4score != 14 && n4score != 22)

{

n4score = 0;

flagx = true;

}

}

n4score = n4score + 26;

}

if (flagx == true)

{

if (n1score < 0 || n1score == 26)

{

n1score = 0;

}

else if (n2score < 0 || n2score == 26)

{

n2score = 0;

}

else if (n3score < 0 || n3score == 26)

{

n3score = 0;

}

else if (n4score < 0 || n4score == 26)

{

n4score = 0;

}

}

flagx = false;

}

}

void newfunc(int &n1score, int &n2score, int &n3score, int &n4score, string op, int dice\_roll, int rowSize, int colSize)

{

if (dice\_roll != 0)

{

bool hello = false;

bool flagx = false;

if (op == "1")

{

if (n1score == 0)

{

if (dice\_roll == 6)

{

n1score = scorefunc(n1score, dice\_roll);

hello = false;

}

else

{

hello = true;

}

flagx = true;

}

else if (n1score == 57)

{

hello = true;

flagx = true;

}

else if (n1score + dice\_roll <= 57)

{

flagx = true;

n1score = scorefunc(n1score, dice\_roll);

hello = false;

}

else if (n1score + dice\_roll > 57 || n1score == 0 && n2score + dice\_roll > 57 || n2score == 0 && n3score + dice\_roll > 57 || n3score == 0 && n4score + dice\_roll > 57 || n4score == 0 && dice\_roll == 6)

{

hello = true;

flagx = true;

}

if (flagx == false)

{

hello = true;

}

if (flagx == true)

{

flagx = false;

}

if (n1score == 57 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n1score == 56 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n1score == 55 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n1score == 54 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n1score == 53 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n1score == 52 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n1score == 52 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n1score > 45 && n2score > 45 && n3score > 45 && n4score > 45)

{

hello = false;

}

}

else if (op == "2")

{

if (n2score == 0)

{

if (dice\_roll == 6)

{

n2score = scorefunc(n2score, dice\_roll);

hello = false;

}

else

{

hello = true;

}

flagx = true;

}

else if (n2score == 57)

{

hello = true;

flagx = true;

}

else if (n2score + dice\_roll <= 57)

{

n2score = scorefunc(n2score, dice\_roll);

hello = false;

flagx = true;

}

else if (n1score + dice\_roll > 57 || n1score == 0 && n2score + dice\_roll > 57 || n2score == 0 && n3score + dice\_roll > 57 || n3score == 0 && n4score + dice\_roll > 57 || n4score == 0 && dice\_roll == 6)

{

hello = true;

flagx = true;

}

if (flagx == false)

{

hello = true;

}

if (flagx == true)

{

flagx = false;

}

if (n2score == 57 && n1score == 0 || n1score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n2score == 56 && n1score == 0 || n1score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n2score == 55 && n1score == 0 || n1score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n2score == 54 && n1score == 0 || n1score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n2score == 53 && n1score == 0 || n1score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n2score == 52 && n1score == 0 || n1score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n1score != 0 && n2score != 0 && n3score != 0 && n4score != 0)

{

hello = false;

}

//

if (n2score > 0)

{

xn2 = 3;

yn2 = 11;

}

}

else if (op == "3")

{

if (n3score == 0)

{

if (dice\_roll == 6)

{

n3score = scorefunc(n3score, dice\_roll);

hello = false;

}

else

{

hello = true;

}

flagx = true;

}

else if (n3score == 57)

{

hello = true;

flagx = true;

}

else if (n3score + dice\_roll <= 57)

{

n3score = scorefunc(n3score, dice\_roll);

hello = false;

flagx = true;

}

else if (n1score + dice\_roll > 57 || n1score == 0 && n2score + dice\_roll > 57 || n2score == 0 && n3score + dice\_roll > 57 || n3score == 0 && n4score + dice\_roll > 57 || n4score == 0 && dice\_roll == 6)

{

hello = true;

flagx = true;

}

if (flagx == false)

{

hello = true;

}

if (flagx == true)

{

flagx = false;

}

if (n3score == 57 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

if (n3score == 56 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n3score == 55 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n3score == 54 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n3score == 53 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n3score == 52 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n1score > 45 && n2score > 45 && n3score > 45 && n4score > 45)

{

hello = false;

}

//

if (n3score > 0)

{

xn3 = 3;

yn3 = 11;

}

}

else if (op == "4")

{

if (n4score == 0)

{

if (dice\_roll == 6)

{

n4score = scorefunc(n4score, dice\_roll);

hello = false;

}

else

{

hello = true;

}

flagx = true;

}

else if (n4score == 57)

{

hello = true;

flagx = true;

}

else if (n4score + dice\_roll <= 57)

{

n4score = scorefunc(n4score, dice\_roll);

hello = false;

flagx = true;

}

else if (n1score + dice\_roll > 57 || n1score == 0 && n2score + dice\_roll > 57 || n2score == 0 && n3score + dice\_roll > 57 || n3score == 0 && n4score + dice\_roll > 57 || n4score == 0 && dice\_roll == 6)

{

hello = true;

flagx = true;

}

if (flagx == false)

{

hello = true;

}

if (flagx == true)

{

flagx = false;

}

if (n4score == 57 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

if (n4score == 56 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n4score == 55 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n4score == 54 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n4score == 53 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n4score == 52 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

hello = false;

}

else if (n1score > 45 && n2score > 45 && n3score > 45 && n4score > 45)

{

hello = false;

}

//

if (n4score > 0)

{

xn4 = 3;

yn4 = 11;

}

}

if (n1score == 0 && n2score == 0 && n3score == 0 && n4score == 0)

{

hello = false;

}

// while loop for conditions near win score

while (hello == true)

{

gotoxy(120, 25);

cin >> op;

if (op == "1")

{

if (n1score == 0)

{

if (dice\_roll == 6)

{

n1score = scorefunc(n1score, dice\_roll);

hello = false;

}

else

{

hello = true;

}

flagx = true;

}

else if (n1score == 57)

{

hello = true;

flagx = true;

}

else if (n1score + dice\_roll <= 57)

{

flagx = true;

n1score = scorefunc(n1score, dice\_roll);

hello = false;

}

else if (n1score + dice\_roll > 57 || n1score == 0 && n2score + dice\_roll > 57 || n2score == 0 && n3score + dice\_roll > 57 || n3score == 0 && n4score + dice\_roll > 57 || n4score == 0 && dice\_roll == 6)

{

hello = true;

flagx = true;

}

if (flagx == false)

{

hello = true;

}

if (flagx == true)

{

flagx = false;

}

if (n1score == 57 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n1score == 56 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n1score == 55 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n1score == 54 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n1score == 53 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n1score == 52 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n1score > 45 && n2score > 45 && n3score > 45 && n4score > 45)

{

hello = false;

}

}

else if (op == "2")

{

if (n2score == 0)

{

if (dice\_roll == 6)

{

n2score = scorefunc(n2score, dice\_roll);

hello = false;

}

else

{

hello = true;

}

flagx = true;

}

else if (n2score == 57)

{

hello = true;

flagx = true;

}

else if (n2score + dice\_roll <= 57)

{

n2score = scorefunc(n2score, dice\_roll);

hello = false;

flagx = true;

}

else if (n1score + dice\_roll > 57 || n1score == 0 && n2score + dice\_roll > 57 || n2score == 0 && n3score + dice\_roll > 57 || n3score == 0 && n4score + dice\_roll > 57 || n4score == 0 && dice\_roll == 6)

{

hello = true;

flagx = true;

}

if (flagx == false)

{

hello = true;

}

if (flagx == true)

{

flagx = false;

}

if (n2score == 57 && n1score == 0 || n1score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n2score == 56 && n1score == 0 || n1score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

hello = false;

}

else if (n2score == 55 && n1score == 0 || n1score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n2score == 54 && n1score == 0 || n1score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n2score == 53 && n1score == 0 || n1score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n2score == 52 && n1score == 0 || n1score == 57 && n3score == 0 || n3score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n1score > 45 && n2score > 45 && n3score > 45 && n4score > 45)

{

hello = false;

}

//

if (n2score > 0)

{

xn2 = 3;

yn2 = 11;

}

}

else if (op == "3")

{

if (n3score == 0)

{

if (dice\_roll == 6)

{

n3score = scorefunc(n3score, dice\_roll);

hello = false;

}

else

{

hello = true;

}

flagx = true;

}

else if (n3score == 57)

{

hello = true;

flagx = true;

}

else if (n3score + dice\_roll <= 57)

{

n3score = scorefunc(n3score, dice\_roll);

hello = false;

flagx = true;

}

else if (n1score + dice\_roll > 57 || n1score == 0 && n2score + dice\_roll > 57 || n2score == 0 && n3score + dice\_roll > 57 || n3score == 0 && n4score + dice\_roll > 57 || n4score == 0 && dice\_roll == 6)

{

hello = true;

flagx = true;

}

if (flagx == false)

{

hello = true;

}

if (flagx == true)

{

flagx = false;

}

if (n3score == 57 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

if (n3score == 56 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n3score == 55 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n3score == 54 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n3score == 53 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n3score == 52 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n4score == 0 || n4score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n1score > 45 && n2score > 45 && n3score > 45 && n4score > 45)

{

hello = false;

}

//

if (n3score > 0)

{

xn3 = 3;

yn3 = 11;

}

}

else if (op == "4")

{

if (n4score == 0)

{

if (dice\_roll == 6)

{

n4score = scorefunc(n4score, dice\_roll);

hello = false;

}

else

{

hello = true;

}

flagx = true;

}

else if (n4score == 57)

{

hello = true;

flagx = true;

}

else if (n4score + dice\_roll <= 57)

{

n4score = scorefunc(n4score, dice\_roll);

hello = false;

flagx = true;

}

else if (n1score + dice\_roll > 57 || n1score == 0 && n2score + dice\_roll > 57 || n2score == 0 && n3score + dice\_roll > 57 || n3score == 0 && n4score + dice\_roll > 57 || n4score == 0 && dice\_roll == 6)

{

hello = true;

flagx = true;

}

if (flagx == false)

{

hello = true;

}

if (flagx == true)

{

flagx = false;

}

if (n4score == 57 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

if (n4score == 56 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n4score == 55 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n4score == 54 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n4score == 53 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n4score == 52 && n1score == 0 || n1score == 57 && n2score == 0 || n2score == 57 && n3score == 0 || n3score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n1score > 45 && n2score > 45 && n3score > 45 && n4score > 45)

{

hello = false;

}

//

if (n4score > 0)

{

xn4 = 3;

yn4 = 11;

}

}

if (n1score == 0 && n2score == 0 && n3score == 0 && n4score == 0)

{

hello = false;

}

}

}

// clearing old beads indexes

for (int i = 0; i < rowSize; i++)

{

for (int z = 0; z < colSize; z++)

{

if (maze[i][z] == '1')

{

maze[i][z] = ' ';

gotoxy(z, i);

cout << maze[i][z];

}

}

}

for (int i = 0; i < rowSize; i++)

{

for (int z = 0; z < colSize; z++)

{

if (maze[i][z] == '2')

{

maze[i][z] = ' ';

gotoxy(z, i);

cout << maze[i][z];

}

}

}

for (int i = 0; i < rowSize; i++)

{

for (int z = 0; z < colSize; z++)

{

if (maze[i][z] == '3')

{

maze[i][z] = ' ';

gotoxy(z, i);

cout << maze[i][z];

}

}

}

for (int i = 0; i < rowSize; i++)

{

for (int z = 0; z < colSize; z++)

{

if (maze[i][z] == '4')

{

maze[i][z] = ' ';

gotoxy(z, i);

cout << maze[i][z];

}

}

}

//

if (n1score == 0)

{

maze[xn1][yn1] = n1;

gotoxy(yn1, xn1);

cout << "X" << endl;

}

if (n1score == 1)

{

maze[xn1 + 10][yn1] = n1;

gotoxy(yn1, xn1 + 10);

cout << "X" << endl;

}

else if (n1score == 2)

{

maze[xn1 + 10][yn1 + 6] = n1;

gotoxy(yn1 + 6, xn1 + 10);

cout << "X" << endl;

}

else if (n1score == 3)

{

maze[xn1 + 10][yn1 + 13] = n1;

gotoxy(yn1 + 13, xn1 + 10);

cout << "X" << endl;

}

else if (n1score == 4)

{

maze[xn1 + 10][yn1 + 21] = n1;

gotoxy(yn1 + 21, xn1 + 10);

cout << "X" << endl;

}

else if (n1score == 5)

{

maze[xn1 + 10][yn1 + 29] = n1;

gotoxy(yn1 + 29, xn1 + 10);

cout << "X" << endl;

}

else if (n1score == 6)

{

maze[xn1 + 8][yn1 + 36] = n1;

gotoxy(yn1 + 36, xn1 + 8);

cout << "X" << endl;

}

else if (n1score == 7)

{

maze[xn1 + 6][yn1 + 36] = n1;

gotoxy(yn1 + 36, xn1 + 6);

cout << "X" << endl;

}

else if (n1score == 8)

{

maze[xn1 + 4][yn1 + 36] = n1;

gotoxy(yn1 + 36, xn1 + 4);

cout << "X" << endl;

}

else if (n1score == 9)

{

maze[xn1 + 2][yn1 + 36] = n1;

gotoxy(yn1 + 36, xn1 + 2);

cout << "X" << endl;

}

else if (n1score == 10)

{

maze[xn1][yn1 + 36] = n1;

gotoxy(yn1 + 36, xn1);

cout << "X" << endl;

}

else if (n1score == 11)

{

maze[xn1 - 2][yn1 + 36] = n1;

gotoxy(yn1 + 36, xn1 - 2);

cout << "X" << endl;

}

else if (n1score == 12)

{

maze[xn1 - 2][yn1 + 42] = n1;

gotoxy(yn1 + 42, xn1 - 2);

cout << "X" << endl;

}

else if (n1score == 13)

{

maze[xn1 - 2][yn1 + 48] = n1;

gotoxy(yn1 + 48, xn1 - 2);

cout << "X" << endl;

}

else if (n1score == 14)

{

maze[xn1][yn1 + 48] = n1;

gotoxy(yn1 + 48, xn1);

cout << "X" << endl;

}

else if (n1score == 15)

{

maze[xn1 + 2][yn1 + 48] = n1;

gotoxy(yn1 + 48, xn1 + 2);

cout << "X" << endl;

}

else if (n1score == 16)

{

maze[xn1 + 4][yn1 + 48] = n1;

gotoxy(yn1 + 48, xn1 + 4);

cout << "X" << endl;

}

else if (n1score == 17)

{

maze[xn1 + 6][yn1 + 48] = n1;

gotoxy(yn1 + 48, xn1 + 6);

cout << "X" << endl;

}

else if (n1score == 18)

{

maze[xn1 + 8][yn1 + 48] = n1;

gotoxy(yn1 + 48, xn1 + 8);

cout << "X" << endl;

}

else if (n1score == 19)

{

maze[xn1 + 10][yn1 + 55] = n1;

gotoxy(yn1 + 55, xn1 + 10);

cout << "X" << endl;

}

else if (n1score == 20)

{

maze[xn1 + 10][yn1 + 63] = n1;

gotoxy(yn1 + 63, xn1 + 10);

cout << "X" << endl;

}

else if (n1score == 21)

{

maze[xn1 + 10][yn1 + 71] = n1;

gotoxy(yn1 + 71, xn1 + 10);

cout << "X" << endl;

}

else if (n1score == 22)

{

maze[xn1 + 10][yn1 + 78] = n1;

gotoxy(yn1 + 78, xn1 + 10);

cout << "X" << endl;

}

else if (n1score == 23)

{

maze[xn1 + 10][yn1 + 85] = n1;

gotoxy(yn1 + 85, xn1 + 10);

cout << "X" << endl;

}

else if (n1score == 24)

{

maze[xn1 + 10][yn1 + 91] = n1;

gotoxy(yn1 + 91, xn1 + 10);

cout << "X" << endl;

}

else if (n1score == 24)

{

maze[xn1 + 10][yn1 + 91] = n1;

gotoxy(yn1 + 91, xn1 + 10);

cout << "X" << endl;

}

else if (n1score == 25)

{

maze[xn1 + 12][yn1 + 91] = n1;

gotoxy(yn1 + 91, xn1 + 12);

cout << "X" << endl;

}

else if (n1score == 26)

{

maze[xn1 + 14][yn1 + 91] = n1;

gotoxy(yn1 + 91, xn1 + 14);

cout << "X" << endl;

}

else if (n1score == 27)

{

maze[xn1 + 14][yn1 + 85] = n1;

gotoxy(yn1 + 85, xn1 + 14);

cout << "X" << endl;

}

else if (n1score == 28)

{

maze[xn1 + 14][yn1 + 78] = n1;

gotoxy(yn1 + 78, xn1 + 14);

cout << "X" << endl;

}

else if (n1score == 29)

{

maze[xn1 + 14][yn1 + 71] = n1;

gotoxy(yn1 + 71, xn1 + 14);

cout << "X" << endl;

}

else if (n1score == 30)

{

maze[xn1 + 14][yn1 + 63] = n1;

gotoxy(yn1 + 63, xn1 + 14);

cout << "X" << endl;

}

else if (n1score == 31)

{

maze[xn1 + 14][yn1 + 55] = n1;

gotoxy(yn1 + 55, xn1 + 14);

cout << "X" << endl;

}

else if (n1score == 32)

{

maze[xn1 + 16][yn1 + 48] = n1;

gotoxy(yn1 + 48, xn1 + 16);

cout << "X" << endl;

}

else if (n1score == 33)

{

maze[xn1 + 18][yn1 + 48] = n1;

gotoxy(yn1 + 48, xn1 + 18);

cout << "X" << endl;

}

else if (n1score == 34)

{

maze[xn1 + 20][yn1 + 48] = n1;

gotoxy(yn1 + 48, xn1 + 20);

cout << "X" << endl;

}

else if (n1score == 35)

{

maze[xn1 + 22][yn1 + 48] = n1;

gotoxy(yn1 + 48, xn1 + 22);

cout << "X" << endl;

}

else if (n1score == 36)

{

maze[xn1 + 24][yn1 + 48] = n1;

gotoxy(yn1 + 48, xn1 + 24);

cout << "X" << endl;

}

else if (n1score == 37)

{

maze[xn1 + 26][yn1 + 48] = n1;

gotoxy(yn1 + 48, xn1 + 26);

cout << "X" << endl;

}

else if (n1score == 38)

{

maze[xn1 + 26][yn1 + 42] = n1;

gotoxy(yn1 + 42, xn1 + 26);

cout << "X" << endl;

}

else if (n1score == 39)

{

maze[xn1 + 26][yn1 + 36] = n1;

gotoxy(yn1 + 36, xn1 + 26);

cout << "X" << endl;

}

else if (n1score == 40)

{

maze[xn1 + 24][yn1 + 36] = n1;

gotoxy(yn1 + 36, xn1 + 24);

cout << "X" << endl;

}

else if (n1score == 41)

{

maze[xn1 + 22][yn1 + 36] = n1;

gotoxy(yn1 + 36, xn1 + 22);

cout << "X" << endl;

}

else if (n1score == 42)

{

maze[xn1 + 20][yn1 + 36] = n1;

gotoxy(yn1 + 36, xn1 + 20);

cout << "X" << endl;

}

else if (n1score == 43)

{

maze[xn1 + 18][yn1 + 36] = n1;

gotoxy(yn1 + 36, xn1 + 18);

cout << "X" << endl;

}

else if (n1score == 44)

{

maze[xn1 + 16][yn1 + 36] = n1;

gotoxy(yn1 + 36, xn1 + 16);

cout << "X" << endl;

}

else if (n1score == 45)

{

maze[xn1 + 14][yn1 + 29] = n1;

gotoxy(yn1 + 29, xn1 + 14);

cout << "X" << endl;

}

else if (n1score == 46)

{

maze[xn1 + 14][yn1 + 21] = n1;

gotoxy(yn1 + 21, xn1 + 14);

cout << "X" << endl;

}

else if (n1score == 47)

{

maze[xn1 + 14][yn1 + 14] = n1;

gotoxy(yn1 + 14, xn1 + 14);

cout << "X" << endl;

}

else if (n1score == 48)

{

maze[xn1 + 14][yn1 + 7] = n1;

gotoxy(yn1 + 7, xn1 + 14);

cout << "X" << endl;

}

else if (n1score == 49)

{

maze[xn1 + 14][yn1] = n1;

gotoxy(yn1, xn1 + 14);

cout << "X" << endl;

}

else if (n1score == 50)

{

maze[xn1 + 14][yn1 - 7] = n1;

gotoxy(yn1 - 7, xn1 + 14);

cout << "X" << endl;

}

else if (n1score == 51)

{

maze[xn1 + 12][yn1 - 7] = n1;

gotoxy(yn1 - 7, xn1 + 12);

cout << "X" << endl;

}

else if (n1score == 52)

{

maze[xn1 + 12][yn1] = n1;

gotoxy(yn1, xn1 + 12);

cout << "X" << endl;

}

else if (n1score == 53)

{

maze[xn1 + 12][yn1 + 7] = n1;

gotoxy(yn1 + 7, xn1 + 12);

cout << "X" << endl;

}

else if (n1score == 54)

{

maze[xn1 + 12][yn1 + 14] = n1;

gotoxy(yn1 + 14, xn1 + 12);

cout << "X" << endl;

}

else if (n1score == 55)

{

maze[xn1 + 12][yn1 + 21] = n1;

gotoxy(yn1 + 21, xn1 + 12);

cout << "X" << endl;

}

else if (n1score == 56)

{

maze[xn1 + 12][yn1 + 28] = n1;

gotoxy(yn1 + 28, xn1 + 12);

cout << "X" << endl;

}

else if (n1score == 57)

{

maze[xn1 + 12][yn1 + 34] = n1;

gotoxy(yn1 + 34, xn1 + 12);

cout << "X" << endl;

}

// for second bead of player 1

if (n2score == 0)

{

maze[xn2][yn2] = n2;

gotoxy(yn2, xn2);

cout << "X" << endl;

}

else if (n2score == 1)

{

maze[xn2 + 10][yn2] = n2;

gotoxy(yn2, xn2 + 10);

cout << "X" << endl;

}

else if (n2score == 2)

{

maze[xn2 + 10][yn2 + 6] = n2;

gotoxy(yn2 + 6, xn2 + 10);

cout << "X" << endl;

}

else if (n2score == 3)

{

maze[xn2 + 10][yn2 + 13] = n2;

gotoxy(yn2 + 13, xn2 + 10);

cout << "X" << endl;

}

else if (n2score == 4)

{

maze[xn2 + 10][yn2 + 21] = n2;

gotoxy(yn2 + 21, xn2 + 10);

cout << "X" << endl;

}

else if (n2score == 5)

{

maze[xn2 + 10][yn2 + 29] = n2;

gotoxy(yn2 + 29, xn2 + 10);

cout << "X" << endl;

}

else if (n2score == 6)

{

maze[xn2 + 8][yn2 + 36] = n2;

gotoxy(yn2 + 36, xn2 + 8);

cout << "X" << endl;

}

else if (n2score == 7)

{

maze[xn2 + 6][yn2 + 36] = n2;

gotoxy(yn2 + 36, xn2 + 6);

cout << "X" << endl;

}

else if (n2score == 8)

{

maze[xn2 + 4][yn2 + 36] = n2;

gotoxy(yn2 + 36, xn2 + 4);

cout << "X" << endl;

}

else if (n2score == 9)

{

maze[xn2 + 2][yn2 + 36] = n2;

gotoxy(yn2 + 36, xn2 + 2);

cout << "X" << endl;

}

else if (n2score == 10)

{

maze[xn2][yn2 + 36] = n2;

gotoxy(yn2 + 36, xn2);

cout << "X" << endl;

}

else if (n2score == 11)

{

maze[xn2 - 2][yn2 + 36] = n2;

gotoxy(yn2 + 36, xn2 - 2);

cout << "X" << endl;

}

else if (n2score == 12)

{

maze[xn2 - 2][yn2 + 42] = n2;

gotoxy(yn2 + 42, xn2 - 2);

cout << "X" << endl;

}

else if (n2score == 13)

{

maze[xn2 - 2][yn2 + 48] = n2;

gotoxy(yn2 + 48, xn2 - 2);

cout << "X" << endl;

}

else if (n2score == 14)

{

maze[xn2][yn2 + 48] = n2;

gotoxy(yn2 + 48, xn2);

cout << "X" << endl;

}

else if (n2score == 15)

{

maze[xn2 + 2][yn2 + 48] = n2;

gotoxy(yn2 + 48, xn2 + 2);

cout << "X" << endl;

}

else if (n2score == 16)

{

maze[xn2 + 4][yn2 + 48] = n2;

gotoxy(yn2 + 48, xn2 + 4);

cout << "X" << endl;

}

else if (n2score == 17)

{

maze[xn2 + 6][yn2 + 48] = n2;

gotoxy(yn2 + 48, xn2 + 6);

cout << "X" << endl;

}

else if (n2score == 18)

{

maze[xn2 + 8][yn2 + 48] = n2;

gotoxy(yn2 + 48, xn2 + 8);

cout << "X" << endl;

}

else if (n2score == 19)

{

maze[xn2 + 10][yn2 + 55] = n2;

gotoxy(yn2 + 55, xn2 + 10);

cout << "X" << endl;

}

else if (n2score == 20)

{

maze[xn2 + 10][yn2 + 63] = n2;

gotoxy(yn2 + 63, xn2 + 10);

cout << "X" << endl;

}

else if (n2score == 21)

{

maze[xn2 + 10][yn2 + 71] = n2;

gotoxy(yn2 + 71, xn2 + 10);

cout << "X" << endl;

}

else if (n2score == 22)

{

maze[xn2 + 10][yn2 + 78] = n2;

gotoxy(yn2 + 78, xn2 + 10);

cout << "X" << endl;

}

else if (n2score == 23)

{

maze[xn2 + 10][yn2 + 85] = n2;

gotoxy(yn2 + 85, xn2 + 10);

cout << "X" << endl;

}

else if (n2score == 24)

{

maze[xn2 + 10][yn2 + 91] = n2;

gotoxy(yn2 + 91, xn2 + 10);

cout << "X" << endl;

}

else if (n2score == 24)

{

maze[xn2 + 10][yn2 + 91] = n2;

gotoxy(yn2 + 91, xn2 + 10);

cout << "X" << endl;

}

else if (n2score == 25)

{

maze[xn2 + 12][yn2 + 91] = n2;

gotoxy(yn2 + 91, xn2 + 12);

cout << "X" << endl;

}

else if (n2score == 26)

{

maze[xn2 + 14][yn2 + 91] = n2;

gotoxy(yn2 + 91, xn2 + 14);

cout << "X" << endl;

}

else if (n2score == 27)

{

maze[xn2 + 14][yn2 + 85] = n2;

gotoxy(yn2 + 85, xn2 + 14);

cout << "X" << endl;

}

else if (n2score == 28)

{

maze[xn2 + 14][yn2 + 78] = n2;

gotoxy(yn2 + 78, xn2 + 14);

cout << "X" << endl;

}

else if (n2score == 29)

{

maze[xn2 + 14][yn2 + 71] = n2;

gotoxy(yn2 + 71, xn2 + 14);

cout << "X" << endl;

}

else if (n2score == 30)

{

maze[xn2 + 14][yn2 + 63] = n2;

gotoxy(yn2 + 63, xn2 + 14);

cout << "X" << endl;

}

else if (n2score == 31)

{

maze[xn2 + 14][yn2 + 55] = n2;

gotoxy(yn2 + 55, xn2 + 14);

cout << "X" << endl;

}

else if (n2score == 32)

{

maze[xn2 + 16][yn2 + 48] = n2;

gotoxy(yn2 + 48, xn2 + 16);

cout << "X" << endl;

}

else if (n2score == 33)

{

maze[xn2 + 18][yn2 + 48] = n2;

gotoxy(yn2 + 48, xn2 + 18);

cout << "X" << endl;

}

else if (n2score == 34)

{

maze[xn2 + 20][yn2 + 48] = n2;

gotoxy(yn2 + 48, xn2 + 20);

cout << "X" << endl;

}

else if (n2score == 35)

{

maze[xn2 + 22][yn2 + 48] = n2;

gotoxy(yn2 + 48, xn2 + 22);

cout << "X" << endl;

}

else if (n2score == 36)

{

maze[xn2 + 24][yn2 + 48] = n2;

gotoxy(yn2 + 48, xn2 + 24);

cout << "X" << endl;

}

else if (n2score == 37)

{

maze[xn2 + 26][yn2 + 48] = n2;

gotoxy(yn2 + 48, xn2 + 26);

cout << "X" << endl;

}

else if (n2score == 38)

{

maze[xn2 + 26][yn2 + 42] = n2;

gotoxy(yn2 + 42, xn2 + 26);

cout << "X" << endl;

}

else if (n2score == 39)

{

maze[xn2 + 26][yn2 + 36] = n2;

gotoxy(yn2 + 36, xn2 + 26);

cout << "X" << endl;

}

else if (n2score == 40)

{

maze[xn2 + 24][yn2 + 36] = n2;

gotoxy(yn2 + 36, xn2 + 24);

cout << "X" << endl;

}

else if (n2score == 41)

{

maze[xn2 + 22][yn2 + 36] = n2;

gotoxy(yn2 + 36, xn2 + 22);

cout << "X" << endl;

}

else if (n2score == 42)

{

maze[xn2 + 20][yn2 + 36] = n2;

gotoxy(yn2 + 36, xn2 + 20);

cout << "X" << endl;

}

else if (n2score == 43)

{

maze[xn2 + 18][yn2 + 36] = n2;

gotoxy(yn2 + 36, xn2 + 18);

cout << "X" << endl;

}

else if (n2score == 44)

{

maze[xn2 + 16][yn2 + 36] = n2;

gotoxy(yn2 + 36, xn2 + 16);

cout << "X" << endl;

}

else if (n2score == 45)

{

maze[xn2 + 14][yn2 + 29] = n2;

gotoxy(yn2 + 29, xn2 + 14);

cout << "X" << endl;

}

else if (n2score == 46)

{

maze[xn2 + 14][yn2 + 21] = n2;

gotoxy(yn2 + 21, xn2 + 14);

cout << "X" << endl;

}

else if (n2score == 47)

{

maze[xn2 + 14][yn2 + 14] = n2;

gotoxy(yn2 + 14, xn2 + 14);

cout << "X" << endl;

}

else if (n2score == 48)

{

maze[xn2 + 14][yn2 + 7] = n2;

gotoxy(yn2 + 7, xn2 + 14);

cout << "X" << endl;

}

else if (n2score == 49)

{

maze[xn2 + 14][yn2] = n2;

gotoxy(yn2, xn2 + 14);

cout << "X" << endl;

}

else if (n2score == 50)

{

maze[xn2 + 14][yn2 - 7] = n2;

gotoxy(yn2 - 7, xn2 + 14);

cout << "X" << endl;

}

else if (n2score == 51)

{

maze[xn2 + 12][yn2 - 7] = n2;

gotoxy(yn2 - 7, xn2 + 12);

cout << "X" << endl;

}

else if (n2score == 52)

{

maze[xn2 + 12][yn2] = n2;

gotoxy(yn2, xn2 + 12);

cout << "X" << endl;

}

else if (n2score == 53)

{

maze[xn2 + 12][yn2 + 7] = n2;

gotoxy(yn2 + 7, xn2 + 12);

cout << "X" << endl;

}

else if (n2score == 54)

{

maze[xn2 + 12][yn2 + 14] = n2;

gotoxy(yn2 + 14, xn2 + 12);

cout << "X" << endl;

}

else if (n2score == 55)

{

maze[xn2 + 12][yn2 + 21] = n2;

gotoxy(yn2 + 21, xn2 + 12);

cout << "X" << endl;

}

else if (n2score == 56)

{

maze[xn2 + 12][yn2 + 28] = n2;

gotoxy(yn2 + 28, xn2 + 12);

cout << "X" << endl;

}

else if (n2score == 57)

{

maze[xn2 + 12][yn2 + 34] = n2;

gotoxy(yn2 + 34, xn2 + 12);

cout << "X" << endl;

}

// displaying third bead

if (n3score == 0)

{

maze[xn3][yn3] = n3;

gotoxy(yn3, xn3);

cout << "X" << endl;

}

else if (n3score == 1)

{

maze[xn3 + 10][yn3] = n3;

gotoxy(yn3, xn3 + 10);

cout << "X" << endl;

}

else if (n3score == 2)

{

maze[xn3 + 10][yn3 + 6] = n3;

gotoxy(yn3 + 6, xn3 + 10);

cout << "X" << endl;

}

else if (n3score == 3)

{

maze[xn3 + 10][yn3 + 13] = n3;

gotoxy(yn3 + 13, xn3 + 10);

cout << "X" << endl;

}

else if (n3score == 4)

{

maze[xn3 + 10][yn3 + 21] = n3;

gotoxy(yn3 + 21, xn3 + 10);

cout << "X" << endl;

}

else if (n3score == 5)

{

maze[xn3 + 10][yn3 + 29] = n3;

gotoxy(yn3 + 29, xn3 + 10);

cout << "X" << endl;

}

else if (n3score == 6)

{

maze[xn3 + 8][yn3 + 36] = n3;

gotoxy(yn3 + 36, xn3 + 8);

cout << "X" << endl;

}

else if (n3score == 7)

{

maze[xn3 + 6][yn3 + 36] = n3;

gotoxy(yn3 + 36, xn3 + 6);

cout << "X" << endl;

}

else if (n3score == 8)

{

maze[xn3 + 4][yn3 + 36] = n3;

gotoxy(yn3 + 36, xn3 + 4);

cout << "X" << endl;

}

else if (n3score == 9)

{

maze[xn3 + 2][yn3 + 36] = n3;

gotoxy(yn3 + 36, xn3 + 2);

cout << "X" << endl;

}

else if (n3score == 10)

{

maze[xn3][yn3 + 36] = n3;

gotoxy(yn3 + 36, xn3);

cout << "X" << endl;

}

else if (n3score == 11)

{

maze[xn3 - 2][yn3 + 36] = n3;

gotoxy(yn3 + 36, xn3 - 2);

cout << "X" << endl;

}

else if (n3score == 12)

{

maze[xn3 - 2][yn3 + 42] = n3;

gotoxy(yn3 + 42, xn3 - 2);

cout << "X" << endl;

}

else if (n3score == 13)

{

maze[xn3 - 2][yn3 + 48] = n3;

gotoxy(yn3 + 48, xn3 - 2);

cout << "X" << endl;

}

else if (n3score == 14)

{

maze[xn3][yn3 + 48] = n3;

gotoxy(yn3 + 48, xn3);

cout << "X" << endl;

}

else if (n3score == 15)

{

maze[xn3 + 2][yn3 + 48] = n3;

gotoxy(yn3 + 48, xn3 + 2);

cout << "X" << endl;

}

else if (n3score == 16)

{

maze[xn3 + 4][yn3 + 48] = n3;

gotoxy(yn3 + 48, xn3 + 4);

cout << "X" << endl;

}

else if (n3score == 17)

{

maze[xn3 + 6][yn3 + 48] = n3;

gotoxy(yn3 + 48, xn3 + 6);

cout << "X" << endl;

}

else if (n3score == 18)

{

maze[xn3 + 8][yn3 + 48] = n3;

gotoxy(yn3 + 48, xn3 + 8);

cout << "X" << endl;

}

else if (n3score == 19)

{

maze[xn3 + 10][yn3 + 55] = n3;

gotoxy(yn3 + 55, xn3 + 10);

cout << "X" << endl;

}

else if (n3score == 20)

{

maze[xn3 + 10][yn3 + 63] = n3;

gotoxy(yn3 + 63, xn3 + 10);

cout << "X" << endl;

}

else if (n3score == 21)

{

maze[xn3 + 10][yn3 + 71] = n3;

gotoxy(yn3 + 71, xn3 + 10);

cout << "X" << endl;

}

else if (n3score == 22)

{

maze[xn3 + 10][yn3 + 78] = n3;

gotoxy(yn3 + 78, xn3 + 10);

cout << "X" << endl;

}

else if (n3score == 23)

{

maze[xn3 + 10][yn3 + 85] = n3;

gotoxy(yn3 + 85, xn3 + 10);

cout << "X" << endl;

}

else if (n3score == 24)

{

maze[xn3 + 10][yn3 + 91] = n3;

gotoxy(yn3 + 91, xn3 + 10);

cout << "X" << endl;

}

else if (n3score == 24)

{

maze[xn3 + 10][yn3 + 91] = n3;

gotoxy(yn3 + 91, xn3 + 10);

cout << "X" << endl;

}

else if (n3score == 25)

{

maze[xn3 + 12][yn3 + 91] = n3;

gotoxy(yn3 + 91, xn3 + 12);

cout << "X" << endl;

}

else if (n3score == 26)

{

maze[xn3 + 14][yn3 + 91] = n3;

gotoxy(yn3 + 91, xn3 + 14);

cout << "X" << endl;

}

else if (n3score == 27)

{

maze[xn3 + 14][yn3 + 85] = n3;

gotoxy(yn3 + 85, xn3 + 14);

cout << "X" << endl;

}

else if (n3score == 28)

{

maze[xn3 + 14][yn3 + 78] = n3;

gotoxy(yn3 + 78, xn3 + 14);

cout << "X" << endl;

}

else if (n3score == 29)

{

maze[xn3 + 14][yn3 + 71] = n3;

gotoxy(yn3 + 71, xn3 + 14);

cout << "X" << endl;

}

else if (n3score == 30)

{

maze[xn3 + 14][yn3 + 63] = n3;

gotoxy(yn3 + 63, xn3 + 14);

cout << "X" << endl;

}

else if (n3score == 31)

{

maze[xn3 + 14][yn3 + 55] = n3;

gotoxy(yn3 + 55, xn3 + 14);

cout << "X" << endl;

}

else if (n3score == 32)

{

maze[xn3 + 16][yn3 + 48] = n3;

gotoxy(yn3 + 48, xn3 + 16);

cout << "X" << endl;

}

else if (n3score == 33)

{

maze[xn3 + 18][yn3 + 48] = n3;

gotoxy(yn3 + 48, xn3 + 18);

cout << "X" << endl;

}

else if (n3score == 34)

{

maze[xn3 + 20][yn3 + 48] = n3;

gotoxy(yn3 + 48, xn3 + 20);

cout << "X" << endl;

}

else if (n3score == 35)

{

maze[xn3 + 22][yn3 + 48] = n3;

gotoxy(yn3 + 48, xn3 + 22);

cout << "X" << endl;

}

else if (n3score == 36)

{

maze[xn3 + 24][yn3 + 48] = n3;

gotoxy(yn3 + 48, xn3 + 24);

cout << "X" << endl;

}

else if (n3score == 37)

{

maze[xn3 + 26][yn3 + 48] = n3;

gotoxy(yn3 + 48, xn3 + 26);

cout << "X" << endl;

}

else if (n3score == 38)

{

maze[xn3 + 26][yn3 + 42] = n3;

gotoxy(yn3 + 42, xn3 + 26);

cout << "X" << endl;

}

else if (n3score == 39)

{

maze[xn3 + 26][yn3 + 36] = n3;

gotoxy(yn3 + 36, xn3 + 26);

cout << "X" << endl;

}

else if (n3score == 40)

{

maze[xn3 + 24][yn3 + 36] = n3;

gotoxy(yn3 + 36, xn3 + 24);

cout << "X" << endl;

}

else if (n3score == 41)

{

maze[xn3 + 22][yn3 + 36] = n3;

gotoxy(yn3 + 36, xn3 + 22);

cout << "X" << endl;

}

else if (n3score == 42)

{

maze[xn3 + 20][yn3 + 36] = n3;

gotoxy(yn3 + 36, xn3 + 20);

cout << "X" << endl;

}

else if (n3score == 43)

{

maze[xn3 + 18][yn3 + 36] = n3;

gotoxy(yn3 + 36, xn3 + 18);

cout << "X" << endl;

}

else if (n3score == 44)

{

maze[xn3 + 16][yn3 + 36] = n3;

gotoxy(yn3 + 36, xn3 + 16);

cout << "X" << endl;

}

else if (n3score == 45)

{

maze[xn3 + 14][yn3 + 29] = n3;

gotoxy(yn3 + 29, xn3 + 14);

cout << "X" << endl;

}

else if (n3score == 46)

{

maze[xn3 + 14][yn3 + 21] = n3;

gotoxy(yn3 + 21, xn3 + 14);

cout << "X" << endl;

}

else if (n3score == 47)

{

maze[xn3 + 14][yn3 + 14] = n3;

gotoxy(yn3 + 14, xn3 + 14);

cout << "X" << endl;

}

else if (n3score == 48)

{

maze[xn3 + 14][yn3 + 7] = n3;

gotoxy(yn3 + 7, xn3 + 14);

cout << "X" << endl;

}

else if (n3score == 49)

{

maze[xn3 + 14][yn3] = n3;

gotoxy(yn3, xn3 + 14);

cout << "X" << endl;

}

else if (n3score == 50)

{

maze[xn3 + 14][yn3 - 7] = n3;

gotoxy(yn3 - 7, xn3 + 14);

cout << "X" << endl;

}

else if (n3score == 51)

{

maze[xn3 + 12][yn3 - 7] = n3;

gotoxy(yn3 - 7, xn3 + 12);

cout << "X" << endl;

}

else if (n3score == 52)

{

maze[xn3 + 12][yn3] = n3;

gotoxy(yn3, xn3 + 12);

cout << "X" << endl;

}

else if (n3score == 53)

{

maze[xn3 + 12][yn3 + 7] = n3;

gotoxy(yn3 + 7, xn3 + 12);

cout << "X" << endl;

}

else if (n3score == 54)

{

maze[xn3 + 12][yn3 + 14] = n3;

gotoxy(yn3 + 14, xn3 + 12);

cout << "X" << endl;

}

else if (n3score == 55)

{

maze[xn3 + 12][yn3 + 21] = n3;

gotoxy(yn3 + 21, xn3 + 12);

cout << "X" << endl;

}

else if (n3score == 56)

{

maze[xn3 + 12][yn3 + 28] = n3;

gotoxy(yn3 + 28, xn3 + 12);

cout << "X" << endl;

}

else if (n3score == 57)

{

maze[xn3 + 12][yn3 + 34] = n3;

gotoxy(yn3 + 34, xn3 + 12);

cout << "X" << endl;

}

// fourth bead

if (n4score == 0)

{

maze[xn4][yn4] = n4;

gotoxy(yn4, xn4);

cout << "X" << endl;

}

else if (n4score == 1)

{

maze[xn4 + 10][yn4] = n4;

gotoxy(yn4, xn4 + 10);

cout << "X" << endl;

}

else if (n4score == 2)

{

maze[xn4 + 10][yn4 + 6] = n4;

gotoxy(yn4 + 6, xn4 + 10);

cout << "X" << endl;

}

else if (n4score == 3)

{

maze[xn4 + 10][yn4 + 13] = n4;

gotoxy(yn4 + 13, xn4 + 10);

cout << "X" << endl;

}

else if (n4score == 4)

{

maze[xn4 + 10][yn4 + 21] = n4;

gotoxy(yn4 + 21, xn4 + 10);

cout << "X" << endl;

}

else if (n4score == 5)

{

maze[xn4 + 10][yn4 + 29] = n4;

gotoxy(yn4 + 29, xn4 + 10);

cout << "X" << endl;

}

else if (n4score == 6)

{

maze[xn4 + 8][yn4 + 36] = n4;

gotoxy(yn4 + 36, xn4 + 8);

cout << "X" << endl;

}

else if (n4score == 7)

{

maze[xn4 + 6][yn4 + 36] = n4;

gotoxy(yn4 + 36, xn4 + 6);

cout << "X" << endl;

}

else if (n4score == 8)

{

maze[xn4 + 4][yn4 + 36] = n4;

gotoxy(yn4 + 36, xn4 + 4);

cout << "X" << endl;

}

else if (n4score == 9)

{

maze[xn4 + 2][yn4 + 36] = n4;

gotoxy(yn4 + 36, xn4 + 2);

cout << "X" << endl;

}

else if (n4score == 10)

{

maze[xn4][yn4 + 36] = n4;

gotoxy(yn4 + 36, xn4);

cout << "X" << endl;

}

else if (n4score == 11)

{

maze[xn4 - 2][yn4 + 36] = n4;

gotoxy(yn4 + 36, xn4 - 2);

cout << "X" << endl;

}

else if (n4score == 12)

{

maze[xn4 - 2][yn4 + 42] = n4;

gotoxy(yn4 + 42, xn4 - 2);

cout << "X" << endl;

}

else if (n4score == 13)

{

maze[xn4 - 2][yn4 + 48] = n4;

gotoxy(yn4 + 48, xn4 - 2);

cout << "X" << endl;

}

else if (n4score == 14)

{

maze[xn4][yn4 + 48] = n4;

gotoxy(yn4 + 48, xn4);

cout << "X" << endl;

}

else if (n4score == 15)

{

maze[xn4 + 2][yn4 + 48] = n4;

gotoxy(yn4 + 48, xn4 + 2);

cout << "X" << endl;

}

else if (n4score == 16)

{

maze[xn4 + 4][yn4 + 48] = n4;

gotoxy(yn4 + 48, xn4 + 4);

cout << "X" << endl;

}

else if (n4score == 17)

{

maze[xn4 + 6][yn4 + 48] = n4;

gotoxy(yn4 + 48, xn4 + 6);

cout << "X" << endl;

}

else if (n4score == 18)

{

maze[xn4 + 8][yn4 + 48] = n4;

gotoxy(yn4 + 48, xn4 + 8);

cout << "X" << endl;

}

else if (n4score == 19)

{

maze[xn4 + 10][yn4 + 55] = n4;

gotoxy(yn4 + 55, xn4 + 10);

cout << "X" << endl;

}

else if (n4score == 20)

{

maze[xn4 + 10][yn4 + 63] = n4;

gotoxy(yn4 + 63, xn4 + 10);

cout << "X" << endl;

}

else if (n4score == 21)

{

maze[xn4 + 10][yn4 + 71] = n4;

gotoxy(yn4 + 71, xn4 + 10);

cout << "X" << endl;

}

else if (n4score == 22)

{

maze[xn4 + 10][yn4 + 78] = n4;

gotoxy(yn4 + 78, xn4 + 10);

cout << "X" << endl;

}

else if (n4score == 23)

{

maze[xn4 + 10][yn4 + 85] = n4;

gotoxy(yn4 + 85, xn4 + 10);

cout << "X" << endl;

}

else if (n4score == 24)

{

maze[xn4 + 10][yn4 + 91] = n4;

gotoxy(yn4 + 91, xn4 + 10);

cout << "X" << endl;

}

else if (n4score == 24)

{

maze[xn4 + 10][yn4 + 91] = n4;

gotoxy(yn4 + 91, xn4 + 10);

cout << "X" << endl;

}

else if (n4score == 25)

{

maze[xn4 + 12][yn4 + 91] = n4;

gotoxy(yn4 + 91, xn4 + 12);

cout << "X" << endl;

}

else if (n4score == 26)

{

maze[xn4 + 14][yn4 + 91] = n4;

gotoxy(yn4 + 91, xn4 + 14);

cout << "X" << endl;

}

else if (n4score == 27)

{

maze[xn4 + 14][yn4 + 85] = n4;

gotoxy(yn4 + 85, xn4 + 14);

cout << "X" << endl;

}

else if (n4score == 28)

{

maze[xn4 + 14][yn4 + 78] = n4;

gotoxy(yn4 + 78, xn4 + 14);

cout << "X" << endl;

}

else if (n4score == 29)

{

maze[xn4 + 14][yn4 + 71] = n4;

gotoxy(yn4 + 71, xn4 + 14);

cout << "X" << endl;

}

else if (n4score == 30)

{

maze[xn4 + 14][yn4 + 63] = n4;

gotoxy(yn4 + 63, xn4 + 14);

cout << "X" << endl;

}

else if (n4score == 31)

{

maze[xn4 + 14][yn4 + 55] = n4;

gotoxy(yn4 + 55, xn4 + 14);

cout << "X" << endl;

}

else if (n4score == 32)

{

maze[xn4 + 16][yn4 + 48] = n4;

gotoxy(yn4 + 48, xn4 + 16);

cout << "X" << endl;

}

else if (n4score == 33)

{

maze[xn4 + 18][yn4 + 48] = n4;

gotoxy(yn4 + 48, xn4 + 18);

cout << "X" << endl;

}

else if (n4score == 34)

{

maze[xn4 + 20][yn4 + 48] = n4;

gotoxy(yn4 + 48, xn4 + 20);

cout << "X" << endl;

}

else if (n4score == 35)

{

maze[xn4 + 22][yn4 + 48] = n4;

gotoxy(yn4 + 48, xn4 + 22);

cout << "X" << endl;

}

else if (n4score == 36)

{

maze[xn4 + 24][yn4 + 48] = n4;

gotoxy(yn4 + 48, xn4 + 24);

cout << "X" << endl;

}

else if (n4score == 37)

{

maze[xn4 + 26][yn4 + 48] = n4;

gotoxy(yn4 + 48, xn4 + 26);

cout << "X" << endl;

}

else if (n4score == 38)

{

maze[xn4 + 26][yn4 + 42] = n4;

gotoxy(yn4 + 42, xn4 + 26);

cout << "X" << endl;

}

else if (n4score == 39)

{

maze[xn4 + 26][yn4 + 36] = n4;

gotoxy(yn4 + 36, xn4 + 26);

cout << "X" << endl;

}

else if (n4score == 40)

{

maze[xn4 + 24][yn4 + 36] = n4;

gotoxy(yn4 + 36, xn4 + 24);

cout << "X" << endl;

}

else if (n4score == 41)

{

maze[xn4 + 22][yn4 + 36] = n4;

gotoxy(yn4 + 36, xn4 + 22);

cout << "X" << endl;

}

else if (n4score == 42)

{

maze[xn4 + 20][yn4 + 36] = n4;

gotoxy(yn4 + 36, xn4 + 20);

cout << "X" << endl;

}

else if (n4score == 43)

{

maze[xn4 + 18][yn4 + 36] = n4;

gotoxy(yn4 + 36, xn4 + 18);

cout << "X" << endl;

}

else if (n4score == 44)

{

maze[xn4 + 16][yn4 + 36] = n4;

gotoxy(yn4 + 36, xn4 + 16);

cout << "X" << endl;

}

else if (n4score == 45)

{

maze[xn4 + 14][yn4 + 29] = n4;

gotoxy(yn4 + 29, xn4 + 14);

cout << "X" << endl;

}

else if (n4score == 46)

{

maze[xn4 + 14][yn4 + 21] = n4;

gotoxy(yn4 + 21, xn4 + 14);

cout << "X" << endl;

}

else if (n4score == 47)

{

maze[xn4 + 14][yn4 + 14] = n4;

gotoxy(yn4 + 14, xn4 + 14);

cout << "X" << endl;

}

else if (n4score == 48)

{

maze[xn4 + 14][yn4 + 7] = n4;

gotoxy(yn4 + 7, xn4 + 14);

cout << "X" << endl;

}

else if (n4score == 49)

{

maze[xn4 + 14][yn4] = n4;

gotoxy(yn4, xn4 + 14);

cout << "X" << endl;

}

else if (n4score == 50)

{

maze[xn4 + 14][yn4 - 7] = n4;

gotoxy(yn4 - 7, xn4 + 14);

cout << "X" << endl;

}

else if (n4score == 51)

{

maze[xn4 + 12][yn4 - 7] = n4;

gotoxy(yn4 - 7, xn4 + 12);

cout << "X" << endl;

}

else if (n4score == 52)

{

maze[xn4 + 12][yn4] = n4;

gotoxy(yn4, xn4 + 12);

cout << "X" << endl;

}

else if (n4score == 53)

{

maze[xn4 + 12][yn4 + 7] = n4;

gotoxy(yn4 + 7, xn4 + 12);

cout << "X" << endl;

}

else if (n4score == 54)

{

maze[xn4 + 12][yn4 + 14] = n4;

gotoxy(yn4 + 14, xn4 + 12);

cout << "X" << endl;

}

else if (n4score == 55)

{

maze[xn4 + 12][yn4 + 21] = n4;

gotoxy(yn4 + 21, xn4 + 12);

cout << "X" << endl;

}

else if (n4score == 56)

{

maze[xn4 + 12][yn4 + 28] = n4;

gotoxy(yn4 + 28, xn4 + 12);

cout << "X" << endl;

}

else if (n4score == 57)

{

maze[xn4 + 12][yn4 + 34] = n4;

gotoxy(yn4 + 34, xn4 + 12);

cout << "X" << endl;

}

//

}

void colourplayer1beads(int rowSize, int colSize)

{

for (int i = 0; i < rowSize; i++)

{

for (int z = 0; z < colSize; z++)

{

if (maze[i][z] == '1' || maze[i][z] == '2' || maze[i][z] == '3' || maze[i][z] == '4')

{

gotoxy(z, i);

SetConsoleTextAttribute(h, 14);

cout << "X" << endl;

}

}

}

SetConsoleTextAttribute(h, 7);

}

void colourplayer2beads(int rowSize, int colSize)

{

for (int i = 0; i < rowSize; i++)

{

for (int z = 0; z < colSize; z++)

{

if (maze[i][z] == '5' || maze[i][z] == '6' || maze[i][z] == '7' || maze[i][z] == '8')

{

gotoxy(z, i);

SetConsoleTextAttribute(h, 13);

cout << "Y" << endl;

}

}

}

SetConsoleTextAttribute(h, 7);

}

void player2beads(int &n5score, int &n6score, int &n7score, int &n8score, string op, int dice\_roll, int rowSize, int colSize)

{

if (dice\_roll != 0)

{

bool hello = false;

bool flagx = false;

if (op == "1")

{

if (n5score == 0)

{

if (dice\_roll == 6)

{

n5score = scorefunc(n5score, dice\_roll);

hello = false;

}

else

{

hello = true;

}

flagx = true;

}

else if (n5score == 57)

{

hello = true;

flagx = true;

}

else if (n5score + dice\_roll <= 57)

{

flagx = true;

n5score = scorefunc(n5score, dice\_roll);

hello = false;

}

else if (n5score + dice\_roll > 57 || n5score == 0 && n6score + dice\_roll > 57 || n6score == 0 && n7score + dice\_roll > 57 || n7score == 0 && n8score + dice\_roll > 57 || n8score == 0 && dice\_roll == 6)

{

hello = true;

flagx = true;

}

if (flagx == false)

{

hello = true;

}

if (flagx == true)

{

flagx = false;

}

if (n5score == 57 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n5score == 56 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n5score == 55 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n5score == 54 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n5score == 53 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n5score == 52 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n5score == 52 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n5score > 45 && n6score > 45 && n7score > 45 && n8score > 45)

{

hello = false;

}

}

else if (op == "2")

{

if (n6score == 0)

{

if (dice\_roll == 6)

{

n6score = scorefunc(n6score, dice\_roll);

hello = false;

}

else

{

hello = true;

}

flagx = true;

}

else if (n6score == 57)

{

hello = true;

flagx = true;

}

else if (n6score + dice\_roll <= 57)

{

n6score = scorefunc(n6score, dice\_roll);

hello = false;

flagx = true;

}

else if (n5score + dice\_roll > 57 || n5score == 0 && n6score + dice\_roll > 57 || n6score == 0 && n7score + dice\_roll > 57 || n7score == 0 && n8score + dice\_roll > 57 || n8score == 0 && dice\_roll == 6)

{

hello = true;

flagx = true;

}

if (flagx == false)

{

hello = true;

}

if (flagx == true)

{

flagx = false;

}

if (n6score == 57 && n5score == 0 || n5score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n6score == 56 && n5score == 0 || n5score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n6score == 55 && n5score == 0 || n5score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n6score == 54 && n5score == 0 || n5score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n6score == 53 && n5score == 0 || n5score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n6score == 52 && n5score == 0 || n5score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n5score > 45 && n6score > 45 && n7score > 45 && n8score > 45)

{

hello = false;

}

//

if (n6score > 0)

{

xn6 = 22;

yn6 = 94;

}

}

else if (op == "3")

{

if (n7score == 0)

{

if (dice\_roll == 6)

{

n7score = scorefunc(n7score, dice\_roll);

hello = false;

}

else

{

hello = true;

}

flagx = true;

}

else if (n7score == 57)

{

hello = true;

flagx = true;

}

else if (n7score + dice\_roll <= 57)

{

n7score = scorefunc(n7score, dice\_roll);

hello = false;

flagx = true;

}

else if (n5score + dice\_roll > 57 || n5score == 0 && n6score + dice\_roll > 57 || n6score == 0 && n7score + dice\_roll > 57 || n7score == 0 && n8score + dice\_roll > 57 || n8score == 0 && dice\_roll == 6)

{

hello = true;

flagx = true;

}

if (flagx == false)

{

hello = true;

}

if (flagx == true)

{

flagx = false;

}

if (n7score == 57 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

if (n7score == 56 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n7score == 55 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n7score == 54 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n7score == 53 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n7score == 52 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n5score > 45 && n6score > 45 && n7score > 45 && n8score > 45)

{

hello = false;

}

//

if (n7score > 0)

{

xn7 = 22;

yn7 = 94;

}

}

else if (op == "4")

{

if (n8score == 0)

{

if (dice\_roll == 6)

{

n8score = scorefunc(n8score, dice\_roll);

hello = false;

}

else

{

hello = true;

}

flagx = true;

}

else if (n8score == 57)

{

hello = true;

flagx = true;

}

else if (n8score + dice\_roll <= 57)

{

n8score = scorefunc(n8score, dice\_roll);

hello = false;

flagx = true;

}

else if (n5score + dice\_roll > 57 || n5score == 0 && n6score + dice\_roll > 57 || n6score == 0 && n7score + dice\_roll > 57 || n7score == 0 && n8score + dice\_roll > 57 || n8score == 0 && dice\_roll == 6)

{

hello = true;

flagx = true;

}

if (flagx == false)

{

hello = true;

}

if (flagx == true)

{

flagx = false;

}

if (n8score == 57 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

if (n8score == 56 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n8score == 55 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n8score == 54 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n8score == 53 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n8score == 52 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

hello = false;

}

else if (n5score > 45 && n6score > 45 && n7score > 45 && n8score > 45)

{

hello = false;

}

//

if (n8score > 0)

{

xn8 = 22;

yn8 = 94;

}

}

if (n5score == 0 && n6score == 0 && n7score == 0 && n8score == 0)

{

hello = false;

}

// while loop for conditions near win score

while (hello == true)

{

gotoxy(120, 25);

cin >> op;

if (op == "1")

{

if (n5score == 0)

{

if (dice\_roll == 6)

{

n5score = scorefunc(n5score, dice\_roll);

hello = false;

}

else

{

hello = true;

}

flagx = true;

}

else if (n5score == 57)

{

hello = true;

flagx = true;

}

else if (n5score + dice\_roll <= 57)

{

flagx = true;

n5score = scorefunc(n5score, dice\_roll);

hello = false;

}

else if (n5score + dice\_roll > 57 || n5score == 0 && n6score + dice\_roll > 57 || n6score == 0 && n7score + dice\_roll > 57 || n7score == 0 && n8score + dice\_roll > 57 || n8score == 0 && dice\_roll == 6)

{

hello = true;

flagx = true;

}

if (flagx == false)

{

hello = true;

}

if (flagx == true)

{

flagx = false;

}

if (n5score == 57 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n5score == 56 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n5score == 55 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n5score == 54 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n5score == 53 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n5score == 52 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n5score > 45 && n6score > 45 && n7score > 45 && n8score > 45)

{

hello = false;

}

}

else if (op == "2")

{

if (n6score == 0)

{

if (dice\_roll == 6)

{

n6score = scorefunc(n6score, dice\_roll);

hello = false;

}

else

{

hello = true;

}

flagx = true;

}

else if (n6score == 57)

{

hello = true;

flagx = true;

}

else if (n6score + dice\_roll <= 57)

{

n6score = scorefunc(n6score, dice\_roll);

hello = false;

flagx = true;

}

else if (n5score + dice\_roll > 57 || n5score == 0 && n6score + dice\_roll > 57 || n6score == 0 && n7score + dice\_roll > 57 || n7score == 0 && n8score + dice\_roll > 57 || n8score == 0 && dice\_roll == 6)

{

hello = true;

flagx = true;

}

if (flagx == false)

{

hello = true;

}

if (flagx == true)

{

flagx = false;

}

if (n6score == 57 && n5score == 0 || n5score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n6score == 56 && n5score == 0 || n5score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

hello = false;

}

else if (n6score == 55 && n5score == 0 || n5score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n6score == 54 && n5score == 0 || n5score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n6score == 53 && n5score == 0 || n5score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n6score == 52 && n5score == 0 || n5score == 57 && n7score == 0 || n7score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n5score > 45 && n6score > 45 && n7score > 45 && n8score > 45)

{

hello = false;

}

//

if (n6score > 0)

{

xn6 = 22;

yn6 = 94;

}

}

else if (op == "3")

{

if (n7score == 0)

{

if (dice\_roll == 6)

{

n7score = scorefunc(n7score, dice\_roll);

hello = false;

}

else

{

hello = true;

}

flagx = true;

}

else if (n7score == 57)

{

hello = true;

flagx = true;

}

else if (n7score + dice\_roll <= 57)

{

n7score = scorefunc(n7score, dice\_roll);

hello = false;

flagx = true;

}

else if (n5score + dice\_roll > 57 || n5score == 0 && n6score + dice\_roll > 57 || n6score == 0 && n7score + dice\_roll > 57 || n7score == 0 && n8score + dice\_roll > 57 || n8score == 0 && dice\_roll == 6)

{

hello = true;

flagx = true;

}

if (flagx == false)

{

hello = true;

}

if (flagx == true)

{

flagx = false;

}

if (n7score == 57 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

if (n7score == 56 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n7score == 55 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n7score == 54 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n7score == 53 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n7score == 52 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n8score == 0 || n8score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n5score > 45 && n6score > 45 && n7score > 45 && n8score > 45)

{

hello = false;

}

//

if (n7score > 0)

{

xn7 = 22;

yn7 = 94;

}

}

else if (op == "4")

{

if (n8score == 0)

{

if (dice\_roll == 6)

{

n8score = scorefunc(n8score, dice\_roll);

hello = false;

}

else

{

hello = true;

}

flagx = true;

}

else if (n8score == 57)

{

hello = true;

flagx = true;

}

else if (n8score + dice\_roll <= 57)

{

n8score = scorefunc(n8score, dice\_roll);

hello = false;

flagx = true;

}

else if (n5score + dice\_roll > 57 || n5score == 0 && n6score + dice\_roll > 57 || n6score == 0 && n7score + dice\_roll > 57 || n7score == 0 && n8score + dice\_roll > 57 || n8score == 0 && dice\_roll == 6)

{

hello = true;

flagx = true;

}

if (flagx == false)

{

hello = true;

}

if (flagx == true)

{

flagx = false;

}

if (n8score == 57 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

if (n8score == 56 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n8score == 55 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n8score == 54 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n8score == 53 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n8score == 52 && n5score == 0 || n5score == 57 && n6score == 0 || n6score == 57 && n7score == 0 || n7score == 57)

{

if (dice\_roll != 6)

{

hello = false;

}

}

else if (n5score > 45 && n6score > 45 && n7score > 45 && n8score > 45)

{

hello = false;

}

//

if (n8score > 0)

{

xn8 = 22;

yn8 = 94;

}

}

if (n5score == 0 && n6score == 0 && n7score == 0 && n8score == 0)

{

hello = false;

}

}

}

// clearing old beads indexes

for (int i = 0; i < rowSize; i++)

{

for (int z = 0; z < colSize; z++)

{

if (maze[i][z] == '5')

{

maze[i][z] = ' ';

gotoxy(z, i);

cout << maze[i][z];

}

}

}

for (int i = 0; i < rowSize; i++)

{

for (int z = 0; z < colSize; z++)

{

if (maze[i][z] == '6')

{

maze[i][z] = ' ';

gotoxy(z, i);

cout << maze[i][z];

}

}

}

for (int i = 0; i < rowSize; i++)

{

for (int z = 0; z < colSize; z++)

{

if (maze[i][z] == '7')

{

maze[i][z] = ' ';

gotoxy(z, i);

cout << maze[i][z];

}

}

}

for (int i = 0; i < rowSize; i++)

{

for (int z = 0; z < colSize; z++)

{

if (maze[i][z] == '8')

{

maze[i][z] = ' ';

gotoxy(z, i);

cout << maze[i][z];

}

}

}

// first bead of second player

if (n5score == 0)

{

maze[xn5][yn5 + 2] = n5;

gotoxy(yn5 + 2, xn5);

cout << "Y" << endl;

}

else if (n5score == 1)

{

maze[xn5 - 5][yn5 + 1] = n5;

gotoxy(yn5 + 1, xn5 - 5);

cout << "Y" << endl;

}

else if (n5score == 2)

{

maze[xn5 - 5][yn5 - 5] = n5;

gotoxy(yn5 - 5, xn5 - 5);

cout << "Y" << endl;

}

else if (n5score == 3)

{

maze[xn5 - 5][yn5 - 12] = n5;

gotoxy(yn5 - 12, xn5 - 5);

cout << "Y" << endl;

}

else if (n5score == 4)

{

maze[xn5 - 5][yn5 - 19] = n5;

gotoxy(yn5 - 19, xn5 - 5);

cout << "Y" << endl;

}

else if (n5score == 5)

{

maze[xn5 - 5][yn5 - 28] = n5;

gotoxy(yn5 - 28, xn5 - 5);

cout << "Y" << endl;

}

else if (n5score == 6)

{

maze[xn5 - 3][yn5 - 35] = n5;

gotoxy(yn5 - 35, xn5 - 3);

cout << "Y" << endl;

}

else if (n5score == 7)

{

maze[xn5 - 1][yn5 - 35] = n5;

gotoxy(yn5 - 35, xn5 - 1);

cout << "Y" << endl;

}

else if (n5score == 8)

{

maze[xn5 + 1][yn5 - 35] = n5;

gotoxy(yn5 - 35, xn5 + 1);

cout << "Y" << endl;

}

else if (n5score == 9)

{

maze[xn5 + 3][yn5 - 35] = n5;

gotoxy(yn5 - 35, xn5 + 3);

cout << "Y" << endl;

}

else if (n5score == 10)

{

maze[xn5 + 5][yn5 - 35] = n5;

gotoxy(yn5 - 35, xn5 + 5);

cout << "Y" << endl;

}

else if (n5score == 11)

{

maze[xn5 + 7][yn5 - 35] = n5;

gotoxy(yn5 - 35, xn5 + 7);

cout << "Y" << endl;

}

else if (n5score == 12)

{

maze[xn5 + 7][yn5 - 41] = n5;

gotoxy(yn5 - 41, xn5 + 7);

cout << "Y" << endl;

}

else if (n5score == 13)

{

maze[xn5 + 7][yn5 - 47] = n5;

gotoxy(yn5 - 47, xn5 + 7);

cout << "Y" << endl;

}

else if (n5score == 14)

{

maze[xn5 + 5][yn5 - 47] = n5;

gotoxy(yn5 - 47, xn5 + 5);

cout << "Y" << endl;

}

else if (n5score == 15)

{

maze[xn5 + 3][yn5 - 47] = n5;

gotoxy(yn5 - 47, xn5 + 3);

cout << "Y" << endl;

}

else if (n5score == 16)

{

maze[xn5 + 1][yn5 - 47] = n5;

gotoxy(yn5 - 47, xn5 + 1);

cout << "Y" << endl;

}

else if (n5score == 17)

{

maze[xn5 - 1][yn5 - 47] = n5;

gotoxy(yn5 - 47, xn5 - 1);

cout << "Y" << endl;

}

else if (n5score == 18)

{

maze[xn5 - 3][yn5 - 47] = n5;

gotoxy(yn5 - 47, xn5 - 3);

cout << "Y" << endl;

}

else if (n5score == 19)

{

maze[xn5 - 5][yn5 - 54] = n5;

gotoxy(yn5 - 54, xn5 - 5);

cout << "Y" << endl;

}

else if (n5score == 20)

{

maze[xn5 - 5][yn5 - 62] = n5;

gotoxy(yn5 - 62, xn5 - 5);

cout << "Y" << endl;

}

else if (n5score == 21)

{

maze[xn5 - 5][yn5 - 69] = n5;

gotoxy(yn5 - 69, xn5 - 5);

cout << "Y" << endl;

}

else if (n5score == 22)

{

maze[xn5 - 5][yn5 - 76] = n5;

gotoxy(yn5 - 76, xn5 - 5);

cout << "Y" << endl;

}

else if (n5score == 23)

{

maze[xn5 - 5][yn5 - 83] = n5;

gotoxy(yn5 - 83, xn5 - 5);

cout << "Y" << endl;

}

else if (n5score == 24)

{

maze[xn5 - 5][yn5 - 90] = n5;

gotoxy(yn5 - 90, xn5 - 5);

cout << "Y" << endl;

}

else if (n5score == 25)

{

maze[xn5 - 7][yn5 - 90] = n5;

gotoxy(yn5 - 90, xn5 - 7);

cout << "Y" << endl;

}

else if (n5score == 26)

{

maze[xn5 - 9][yn5 - 90] = n5;

gotoxy(yn5 - 90, xn5 - 9);

cout << "Y" << endl;

}

else if (n5score == 27)

{

maze[xn5 - 9][yn5 - 83] = n5;

gotoxy(yn5 - 83, xn5 - 9);

cout << "Y" << endl;

}

else if (n5score == 28)

{

maze[xn5 - 9][yn5 - 76] = n5;

gotoxy(yn5 - 76, xn5 - 9);

cout << "Y" << endl;

}

else if (n5score == 29)

{

maze[xn5 - 9][yn5 - 69] = n5;

gotoxy(yn5 - 69, xn5 - 9);

cout << "Y" << endl;

}

else if (n5score == 30)

{

maze[xn5 - 9][yn5 - 62] = n5;

gotoxy(yn5 - 62, xn5 - 9);

cout << "Y" << endl;

}

else if (n5score == 31)

{

maze[xn5 - 9][yn5 - 54] = n5;

gotoxy(yn5 - 54, xn5 - 9);

cout << "Y" << endl;

}

else if (n5score == 32)

{

maze[xn5 - 11][yn5 - 47] = n5;

gotoxy(yn5 - 47, xn5 - 11);

cout << "Y" << endl;

}

else if (n5score == 33)

{

maze[xn5 - 13][yn5 - 47] = n5;

gotoxy(yn5 - 47, xn5 - 13);

cout << "Y" << endl;

}

else if (n5score == 34)

{

maze[xn5 - 15][yn5 - 47] = n5;

gotoxy(yn5 - 47, xn5 - 15);

cout << "Y" << endl;

}

else if (n5score == 35)

{

maze[xn5 - 17][yn5 - 47] = n5;

gotoxy(yn5 - 47, xn5 - 17);

cout << "Y" << endl;

}

else if (n5score == 36)

{

maze[xn5 - 19][yn5 - 47] = n5;

gotoxy(yn5 - 47, xn5 - 19);

cout << "Y" << endl;

}

else if (n5score == 37)

{

maze[xn5 - 21][yn5 - 47] = n5;

gotoxy(yn5 - 47, xn5 - 21);

cout << "Y" << endl;

}

else if (n5score == 38)

{

maze[xn5 - 21][yn5 - 41] = n5;

gotoxy(yn5 - 41, xn5 - 21);

cout << "Y" << endl;

}

else if (n5score == 39)

{

maze[xn5 - 21][yn5 - 35] = n5;

gotoxy(yn5 - 35, xn5 - 21);

cout << "Y" << endl;

}

else if (n5score == 40)

{

maze[xn5 - 19][yn5 - 35] = n5;

gotoxy(yn5 - 35, xn5 - 19);

cout << "Y" << endl;

}

else if (n5score == 41)

{

maze[xn5 - 17][yn5 - 35] = n5;

gotoxy(yn5 - 35, xn5 - 17);

cout << "Y" << endl;

}

else if (n5score == 42)

{

maze[xn5 - 15][yn5 - 35] = n5;

gotoxy(yn5 - 35, xn5 - 15);

cout << "Y" << endl;

}

else if (n5score == 43)

{

maze[xn5 - 13][yn5 - 35] = n5;

gotoxy(yn5 - 35, xn5 - 13);

cout << "Y" << endl;

}

else if (n5score == 44)

{

maze[xn5 - 11][yn5 - 35] = n5;

gotoxy(yn5 - 35, xn5 - 11);

cout << "Y" << endl;

}

else if (n5score == 45)

{

maze[xn5 - 9][yn5 - 28] = n5;

gotoxy(yn5 - 28, xn5 - 9);

cout << "Y" << endl;

}

else if (n5score == 46)

{

maze[xn5 - 9][yn5 - 20] = n5;

gotoxy(yn5 - 20, xn5 - 9);

cout << "Y" << endl;

}

else if (n5score == 47)

{

maze[xn5 - 9][yn5 - 13] = n5;

gotoxy(yn5 - 13, xn5 - 9);

cout << "Y" << endl;

}

else if (n5score == 48)

{

maze[xn5 - 9][yn5 - 6] = n5;

gotoxy(yn5 - 6, xn5 - 9);

cout << "Y" << endl;

}

else if (n5score == 49)

{

maze[xn5 - 9][yn5 + 1] = n5;

gotoxy(yn5 + 1, xn5 - 9);

cout << "Y" << endl;

}

else if (n5score == 50)

{

maze[xn5 - 9][yn5 + 8] = n5;

gotoxy(yn5 + 8, xn5 - 9);

cout << "Y" << endl;

}

else if (n5score == 51)

{

maze[xn5 - 7][yn5 + 8] = n5;

gotoxy(yn5 + 8, xn5 - 7);

cout << "Y" << endl;

}

else if (n5score == 52)

{

maze[xn5 - 7][yn5 + 1] = n5;

gotoxy(yn5 + 1, xn5 - 7);

cout << "Y" << endl;

}

else if (n5score == 53)

{

maze[xn5 - 7][yn5 - 6] = n5;

gotoxy(yn5 - 6, xn5 - 7);

cout << "Y" << endl;

}

else if (n5score == 54)

{

maze[xn5 - 7][yn5 - 13] = n5;

gotoxy(yn5 - 13, xn5 - 7);

cout << "Y" << endl;

}

else if (n5score == 55)

{

maze[xn5 - 7][yn5 - 20] = n5;

gotoxy(yn5 - 20, xn5 - 7);

cout << "Y" << endl;

}

else if (n5score == 56)

{

maze[xn5 - 7][yn5 - 28] = n5;

gotoxy(yn5 - 28, xn5 - 7);

cout << "Y" << endl;

}

else if (n5score == 57)

{

maze[xn5 - 7][yn5 - 35] = n5;

gotoxy(yn5 - 35, xn5 - 7);

cout << "Y" << endl;

}

// second bead of second player

if (n6score == 0)

{

maze[xn6][yn6] = n6;

gotoxy(yn6, xn6);

cout << "Y" << endl;

}

else if (n6score == 1)

{

maze[xn6 - 5][yn6 + 1] = n6;

gotoxy(yn6 + 1, xn6 - 5);

cout << "Y" << endl;

}

else if (n6score == 2)

{

maze[xn6 - 5][yn6 - 5] = n6;

gotoxy(yn6 - 5, xn6 - 5);

cout << "Y" << endl;

}

else if (n6score == 3)

{

maze[xn6 - 5][yn6 - 12] = n6;

gotoxy(yn6 - 12, xn6 - 5);

cout << "Y" << endl;

}

else if (n6score == 4)

{

maze[xn6 - 5][yn6 - 19] = n6;

gotoxy(yn6 - 19, xn6 - 5);

cout << "Y" << endl;

}

else if (n6score == 5)

{

maze[xn6 - 5][yn6 - 28] = n6;

gotoxy(yn6 - 28, xn6 - 5);

cout << "Y" << endl;

}

else if (n6score == 6)

{

maze[xn6 - 3][yn6 - 35] = n6;

gotoxy(yn6 - 35, xn6 - 3);

cout << "Y" << endl;

}

else if (n6score == 7)

{

maze[xn6 - 1][yn6 - 35] = n6;

gotoxy(yn6 - 35, xn6 - 1);

cout << "Y" << endl;

}

else if (n6score == 8)

{

maze[xn6 + 1][yn6 - 35] = n6;

gotoxy(yn6 - 35, xn6 + 1);

cout << "Y" << endl;

}

else if (n6score == 9)

{

maze[xn6 + 3][yn6 - 35] = n6;

gotoxy(yn6 - 35, xn6 + 3);

cout << "Y" << endl;

}

else if (n6score == 10)

{

maze[xn6 + 5][yn6 - 35] = n6;

gotoxy(yn6 - 35, xn6 + 5);

cout << "Y" << endl;

}

else if (n6score == 11)

{

maze[xn6 + 7][yn6 - 35] = n6;

gotoxy(yn6 - 35, xn6 + 7);

cout << "Y" << endl;

}

else if (n6score == 12)

{

maze[xn6 + 7][yn6 - 41] = n6;

gotoxy(yn6 - 41, xn6 + 7);

cout << "Y" << endl;

}

else if (n6score == 13)

{

maze[xn6 + 7][yn6 - 47] = n6;

gotoxy(yn6 - 47, xn6 + 7);

cout << "Y" << endl;

}

else if (n6score == 14)

{

maze[xn6 + 5][yn6 - 47] = n6;

gotoxy(yn6 - 47, xn6 + 5);

cout << "Y" << endl;

}

else if (n6score == 15)

{

maze[xn6 + 3][yn6 - 47] = n6;

gotoxy(yn6 - 47, xn6 + 3);

cout << "Y" << endl;

}

else if (n6score == 16)

{

maze[xn6 + 1][yn6 - 47] = n6;

gotoxy(yn6 - 47, xn6 + 1);

cout << "Y" << endl;

}

else if (n6score == 17)

{

maze[xn6 - 1][yn6 - 47] = n6;

gotoxy(yn6 - 47, xn6 - 1);

cout << "Y" << endl;

}

else if (n6score == 18)

{

maze[xn6 - 3][yn6 - 47] = n6;

gotoxy(yn6 - 47, xn6 - 3);

cout << "Y" << endl;

}

else if (n6score == 19)

{

maze[xn6 - 5][yn6 - 54] = n6;

gotoxy(yn6 - 54, xn6 - 5);

cout << "Y" << endl;

}

else if (n6score == 20)

{

maze[xn6 - 5][yn6 - 62] = n6;

gotoxy(yn6 - 62, xn6 - 5);

cout << "Y" << endl;

}

else if (n6score == 21)

{

maze[xn6 - 5][yn6 - 69] = n6;

gotoxy(yn6 - 69, xn6 - 5);

cout << "Y" << endl;

}

else if (n6score == 22)

{

maze[xn6 - 5][yn6 - 76] = n6;

gotoxy(yn6 - 76, xn6 - 5);

cout << "Y" << endl;

}

else if (n6score == 23)

{

maze[xn6 - 5][yn6 - 83] = n6;

gotoxy(yn6 - 83, xn6 - 5);

cout << "Y" << endl;

}

else if (n6score == 24)

{

maze[xn6 - 5][yn6 - 90] = n6;

gotoxy(yn6 - 90, xn6 - 5);

cout << "Y" << endl;

}

else if (n6score == 25)

{

maze[xn6 - 7][yn6 - 90] = n6;

gotoxy(yn6 - 90, xn6 - 7);

cout << "Y" << endl;

}

else if (n6score == 26)

{

maze[xn6 - 9][yn6 - 90] = n6;

gotoxy(yn6 - 90, xn6 - 9);

cout << "Y" << endl;

}

else if (n6score == 27)

{

maze[xn6 - 9][yn6 - 83] = n6;

gotoxy(yn6 - 83, xn6 - 9);

cout << "Y" << endl;

}

else if (n6score == 28)

{

maze[xn6 - 9][yn6 - 76] = n6;

gotoxy(yn6 - 76, xn6 - 9);

cout << "Y" << endl;

}

else if (n6score == 29)

{

maze[xn6 - 9][yn6 - 69] = n6;

gotoxy(yn6 - 69, xn6 - 9);

cout << "Y" << endl;

}

else if (n6score == 30)

{

maze[xn6 - 9][yn6 - 62] = n6;

gotoxy(yn6 - 62, xn6 - 9);

cout << "Y" << endl;

}

else if (n6score == 31)

{

maze[xn6 - 9][yn6 - 54] = n6;

gotoxy(yn6 - 54, xn6 - 9);

cout << "Y" << endl;

}

else if (n6score == 32)

{

maze[xn6 - 11][yn6 - 47] = n6;

gotoxy(yn6 - 47, xn6 - 11);

cout << "Y" << endl;

}

else if (n6score == 33)

{

maze[xn6 - 13][yn6 - 47] = n6;

gotoxy(yn6 - 47, xn6 - 13);

cout << "Y" << endl;

}

else if (n6score == 34)

{

maze[xn6 - 15][yn6 - 47] = n6;

gotoxy(yn6 - 47, xn6 - 15);

cout << "Y" << endl;

}

else if (n6score == 35)

{

maze[xn6 - 17][yn6 - 47] = n6;

gotoxy(yn6 - 47, xn6 - 17);

cout << "Y" << endl;

}

else if (n6score == 36)

{

maze[xn6 - 19][yn6 - 47] = n6;

gotoxy(yn6 - 47, xn6 - 19);

cout << "Y" << endl;

}

else if (n6score == 37)

{

maze[xn6 - 21][yn6 - 47] = n6;

gotoxy(yn6 - 47, xn6 - 21);

cout << "Y" << endl;

}

else if (n6score == 38)

{

maze[xn6 - 21][yn6 - 41] = n6;

gotoxy(yn6 - 41, xn6 - 21);

cout << "Y" << endl;

}

else if (n6score == 39)

{

maze[xn6 - 21][yn6 - 35] = n6;

gotoxy(yn6 - 35, xn6 - 21);

cout << "Y" << endl;

}

else if (n6score == 40)

{

maze[xn6 - 19][yn6 - 35] = n6;

gotoxy(yn6 - 35, xn6 - 19);

cout << "Y" << endl;

}

else if (n6score == 41)

{

maze[xn6 - 17][yn6 - 35] = n6;

gotoxy(yn6 - 35, xn6 - 17);

cout << "Y" << endl;

}

else if (n6score == 42)

{

maze[xn6 - 15][yn6 - 35] = n6;

gotoxy(yn6 - 35, xn6 - 15);

cout << "Y" << endl;

}

else if (n6score == 43)

{

maze[xn6 - 13][yn6 - 35] = n6;

gotoxy(yn6 - 35, xn6 - 13);

cout << "Y" << endl;

}

else if (n6score == 44)

{

maze[xn6 - 11][yn6 - 35] = n6;

gotoxy(yn6 - 35, xn6 - 11);

cout << "Y" << endl;

}

else if (n6score == 45)

{

maze[xn6 - 9][yn6 - 28] = n6;

gotoxy(yn6 - 28, xn6 - 9);

cout << "Y" << endl;

}

else if (n6score == 46)

{

maze[xn6 - 9][yn6 - 20] = n6;

gotoxy(yn6 - 20, xn6 - 9);

cout << "Y" << endl;

}

else if (n6score == 47)

{

maze[xn6 - 9][yn6 - 13] = n6;

gotoxy(yn6 - 13, xn6 - 9);

cout << "Y" << endl;

}

else if (n6score == 48)

{

maze[xn6 - 9][yn6 - 6] = n6;

gotoxy(yn6 - 6, xn6 - 9);

cout << "Y" << endl;

}

else if (n6score == 49)

{

maze[xn6 - 9][yn6 + 1] = n6;

gotoxy(yn6 + 1, xn6 - 9);

cout << "Y" << endl;

}

else if (n6score == 50)

{

maze[xn6 - 9][yn6 + 8] = n6;

gotoxy(yn6 + 8, xn6 - 9);

cout << "Y" << endl;

}

else if (n6score == 51)

{

maze[xn6 - 7][yn6 + 8] = n6;

gotoxy(yn6 + 8, xn6 - 7);

cout << "Y" << endl;

}

else if (n6score == 52)

{

maze[xn6 - 7][yn6 + 1] = n6;

gotoxy(yn6 + 1, xn6 - 7);

cout << "Y" << endl;

}

else if (n6score == 53)

{

maze[xn6 - 7][yn6 - 6] = n6;

gotoxy(yn6 - 6, xn6 - 7);

cout << "Y" << endl;

}

else if (n6score == 54)

{

maze[xn6 - 7][yn6 - 13] = n6;

gotoxy(yn6 - 13, xn6 - 7);

cout << "Y" << endl;

}

else if (n6score == 55)

{

maze[xn6 - 7][yn6 - 20] = n6;

gotoxy(yn6 - 20, xn6 - 7);

cout << "Y" << endl;

}

else if (n6score == 56)

{

maze[xn6 - 7][yn6 - 28] = n6;

gotoxy(yn6 - 28, xn6 - 7);

cout << "Y" << endl;

}

else if (n6score == 57)

{

maze[xn6 - 7][yn6 - 35] = n6;

gotoxy(yn6 - 35, xn6 - 7);

cout << "Y" << endl;

}

// third bead of second player

if (n7score == 0)

{

maze[xn7][yn7 + 2] = n8;

gotoxy(yn7 + 2, xn7);

cout << "Y" << endl;

}

else if (n7score == 1)

{

maze[xn7 - 5][yn7 + 1] = n8;

gotoxy(yn7 + 1, xn7 - 5);

cout << "Y" << endl;

}

else if (n7score == 2)

{

maze[xn7 - 5][yn7 - 5] = n8;

gotoxy(yn7 - 5, xn7 - 5);

cout << "Y" << endl;

}

else if (n7score == 3)

{

maze[xn7 - 5][yn7 - 12] = n8;

gotoxy(yn7 - 12, xn7 - 5);

cout << "Y" << endl;

}

else if (n7score == 4)

{

maze[xn7 - 5][yn7 - 19] = n8;

gotoxy(yn7 - 19, xn7 - 5);

cout << "Y" << endl;

}

else if (n7score == 5)

{

maze[xn7 - 5][yn7 - 28] = n8;

gotoxy(yn7 - 28, xn7 - 5);

cout << "Y" << endl;

}

else if (n7score == 6)

{

maze[xn7 - 3][yn7 - 35] = n8;

gotoxy(yn7 - 35, xn7 - 3);

cout << "Y" << endl;

}

else if (n7score == 7)

{

maze[xn7 - 1][yn7 - 35] = n8;

gotoxy(yn7 - 35, xn7 - 1);

cout << "Y" << endl;

}

else if (n7score == 8)

{

maze[xn7 + 1][yn7 - 35] = n8;

gotoxy(yn7 - 35, xn7 + 1);

cout << "Y" << endl;

}

else if (n7score == 9)

{

maze[xn7 + 3][yn7 - 35] = n8;

gotoxy(yn7 - 35, xn7 + 3);

cout << "Y" << endl;

}

else if (n7score == 10)

{

maze[xn7 + 5][yn7 - 35] = n8;

gotoxy(yn7 - 35, xn7 + 5);

cout << "Y" << endl;

}

else if (n7score == 11)

{

maze[xn7 + 7][yn7 - 35] = n8;

gotoxy(yn7 - 35, xn7 + 7);

cout << "Y" << endl;

}

else if (n7score == 12)

{

maze[xn7 + 7][yn7 - 41] = n8;

gotoxy(yn7 - 41, xn7 + 7);

cout << "Y" << endl;

}

else if (n7score == 13)

{

maze[xn7 + 7][yn7 - 47] = n8;

gotoxy(yn7 - 47, xn7 + 7);

cout << "Y" << endl;

}

else if (n7score == 14)

{

maze[xn7 + 5][yn7 - 47] = n8;

gotoxy(yn7 - 47, xn7 + 5);

cout << "Y" << endl;

}

else if (n7score == 15)

{

maze[xn7 + 3][yn7 - 47] = n8;

gotoxy(yn7 - 47, xn7 + 3);

cout << "Y" << endl;

}

else if (n7score == 16)

{

maze[xn7 + 1][yn7 - 47] = n8;

gotoxy(yn7 - 47, xn7 + 1);

cout << "Y" << endl;

}

else if (n7score == 17)

{

maze[xn7 - 1][yn7 - 47] = n8;

gotoxy(yn7 - 47, xn7 - 1);

cout << "Y" << endl;

}

else if (n7score == 18)

{

maze[xn7 - 3][yn7 - 47] = n8;

gotoxy(yn7 - 47, xn7 - 3);

cout << "Y" << endl;

}

else if (n7score == 19)

{

maze[xn7 - 5][yn7 - 54] = n8;

gotoxy(yn7 - 54, xn7 - 5);

cout << "Y" << endl;

}

else if (n7score == 20)

{

maze[xn7 - 5][yn7 - 62] = n8;

gotoxy(yn7 - 62, xn7 - 5);

cout << "Y" << endl;

}

else if (n7score == 21)

{

maze[xn7 - 5][yn7 - 69] = n8;

gotoxy(yn7 - 69, xn7 - 5);

cout << "Y" << endl;

}

else if (n7score == 22)

{

maze[xn7 - 5][yn7 - 76] = n8;

gotoxy(yn7 - 76, xn7 - 5);

cout << "Y" << endl;

}

else if (n7score == 23)

{

maze[xn7 - 5][yn7 - 83] = n8;

gotoxy(yn7 - 83, xn7 - 5);

cout << "Y" << endl;

}

else if (n7score == 24)

{

maze[xn7 - 5][yn7 - 90] = n8;

gotoxy(yn7 - 90, xn7 - 5);

cout << "Y" << endl;

}

else if (n7score == 25)

{

maze[xn7 - 7][yn7 - 90] = n8;

gotoxy(yn7 - 90, xn7 - 7);

cout << "Y" << endl;

}

else if (n7score == 26)

{

maze[xn7 - 9][yn7 - 90] = n8;

gotoxy(yn7 - 90, xn7 - 9);

cout << "Y" << endl;

}

else if (n7score == 27)

{

maze[xn7 - 9][yn7 - 83] = n8;

gotoxy(yn7 - 83, xn7 - 9);

cout << "Y" << endl;

}

else if (n7score == 28)

{

maze[xn7 - 9][yn7 - 76] = n8;

gotoxy(yn7 - 76, xn7 - 9);

cout << "Y" << endl;

}

else if (n7score == 29)

{

maze[xn7 - 9][yn7 - 69] = n8;

gotoxy(yn7 - 69, xn7 - 9);

cout << "Y" << endl;

}

else if (n7score == 30)

{

maze[xn7 - 9][yn7 - 62] = n8;

gotoxy(yn7 - 62, xn7 - 9);

cout << "Y" << endl;

}

else if (n7score == 31)

{

maze[xn7 - 9][yn7 - 54] = n8;

gotoxy(yn7 - 54, xn7 - 9);

cout << "Y" << endl;

}

else if (n7score == 32)

{

maze[xn7 - 11][yn7 - 47] = n8;

gotoxy(yn7 - 47, xn7 - 11);

cout << "Y" << endl;

}

else if (n7score == 33)

{

maze[xn7 - 13][yn7 - 47] = n8;

gotoxy(yn7 - 47, xn7 - 13);

cout << "Y" << endl;

}

else if (n7score == 34)

{

maze[xn7 - 15][yn7 - 47] = n8;

gotoxy(yn7 - 47, xn7 - 15);

cout << "Y" << endl;

}

else if (n7score == 35)

{

maze[xn7 - 17][yn7 - 47] = n8;

gotoxy(yn7 - 47, xn7 - 17);

cout << "Y" << endl;

}

else if (n7score == 36)

{

maze[xn7 - 19][yn7 - 47] = n8;

gotoxy(yn7 - 47, xn7 - 19);

cout << "Y" << endl;

}

else if (n7score == 37)

{

maze[xn7 - 21][yn7 - 47] = n8;

gotoxy(yn7 - 47, xn7 - 21);

cout << "Y" << endl;

}

else if (n7score == 38)

{

maze[xn7 - 21][yn7 - 41] = n8;

gotoxy(yn7 - 41, xn7 - 21);

cout << "Y" << endl;

}

else if (n7score == 39)

{

maze[xn7 - 21][yn7 - 35] = n8;

gotoxy(yn7 - 35, xn7 - 21);

cout << "Y" << endl;

}

else if (n7score == 40)

{

maze[xn7 - 19][yn7 - 35] = n8;

gotoxy(yn7 - 35, xn7 - 19);

cout << "Y" << endl;

}

else if (n7score == 41)

{

maze[xn7 - 17][yn7 - 35] = n8;

gotoxy(yn7 - 35, xn7 - 17);

cout << "Y" << endl;

}

else if (n7score == 42)

{

maze[xn7 - 15][yn7 - 35] = n8;

gotoxy(yn7 - 35, xn7 - 15);

cout << "Y" << endl;

}

else if (n7score == 43)

{

maze[xn7 - 13][yn7 - 35] = n8;

gotoxy(yn7 - 35, xn7 - 13);

cout << "Y" << endl;

}

else if (n7score == 44)

{

maze[xn7 - 11][yn7 - 35] = n8;

gotoxy(yn7 - 35, xn7 - 11);

cout << "Y" << endl;

}

else if (n7score == 45)

{

maze[xn7 - 9][yn7 - 28] = n8;

gotoxy(yn7 - 28, xn7 - 9);

cout << "Y" << endl;

}

else if (n7score == 46)

{

maze[xn7 - 9][yn7 - 20] = n8;

gotoxy(yn7 - 20, xn7 - 9);

cout << "Y" << endl;

}

else if (n7score == 47)

{

maze[xn7 - 9][yn7 - 13] = n8;

gotoxy(yn7 - 13, xn7 - 9);

cout << "Y" << endl;

}

else if (n7score == 48)

{

maze[xn7 - 9][yn7 - 6] = n8;

gotoxy(yn7 - 6, xn7 - 9);

cout << "Y" << endl;

}

else if (n7score == 49)

{

maze[xn7 - 9][yn7 + 1] = n8;

gotoxy(yn7 + 1, xn7 - 9);

cout << "Y" << endl;

}

else if (n7score == 50)

{

maze[xn7 - 9][yn7 + 8] = n8;

gotoxy(yn7 + 8, xn7 - 9);

cout << "Y" << endl;

}

else if (n7score == 51)

{

maze[xn7 - 7][yn7 + 8] = n8;

gotoxy(yn7 + 8, xn7 - 7);

cout << "Y" << endl;

}

else if (n7score == 52)

{

maze[xn7 - 7][yn7 + 1] = n8;

gotoxy(yn7 + 1, xn7 - 7);

cout << "Y" << endl;

}

else if (n7score == 53)

{

maze[xn7 - 7][yn7 - 6] = n8;

gotoxy(yn7 - 6, xn7 - 7);

cout << "Y" << endl;

}

else if (n7score == 54)

{

maze[xn7 - 7][yn7 - 13] = n8;

gotoxy(yn7 - 13, xn7 - 7);

cout << "Y" << endl;

}

else if (n7score == 55)

{

maze[xn7 - 7][yn7 - 20] = n8;

gotoxy(yn7 - 20, xn7 - 7);

cout << "Y" << endl;

}

else if (n7score == 56)

{

maze[xn7 - 7][yn7 - 28] = n8;

gotoxy(yn7 - 28, xn7 - 7);

cout << "Y" << endl;

}

else if (n7score == 57)

{

maze[xn7 - 7][yn7 - 35] = n8;

gotoxy(yn7 - 35, xn7 - 7);

cout << "Y" << endl;

}

// fourth bead of second player

if (n8score == 0)

{

maze[xn8][yn8] = n8;

gotoxy(yn8, xn8);

cout << "Y" << endl;

}

else if (n8score == 1)

{

maze[xn8 - 5][yn8 + 1] = n8;

gotoxy(yn8 + 1, xn8 - 5);

cout << "Y" << endl;

}

else if (n8score == 2)

{

maze[xn8 - 5][yn8 - 5] = n8;

gotoxy(yn8 - 5, xn8 - 5);

cout << "Y" << endl;

}

else if (n8score == 3)

{

maze[xn8 - 5][yn8 - 12] = n8;

gotoxy(yn8 - 12, xn8 - 5);

cout << "Y" << endl;

}

else if (n8score == 4)

{

maze[xn8 - 5][yn8 - 19] = n8;

gotoxy(yn8 - 19, xn8 - 5);

cout << "Y" << endl;

}

else if (n8score == 5)

{

maze[xn8 - 5][yn8 - 28] = n8;

gotoxy(yn8 - 28, xn8 - 5);

cout << "Y" << endl;

}

else if (n8score == 6)

{

maze[xn8 - 3][yn8 - 35] = n8;

gotoxy(yn8 - 35, xn8 - 3);

cout << "Y" << endl;

}

else if (n8score == 7)

{

maze[xn8 - 1][yn8 - 35] = n8;

gotoxy(yn8 - 35, xn8 - 1);

cout << "Y" << endl;

}

else if (n8score == 8)

{

maze[xn8 + 1][yn8 - 35] = n8;

gotoxy(yn8 - 35, xn8 + 1);

cout << "Y" << endl;

}

else if (n8score == 9)

{

maze[xn8 + 3][yn8 - 35] = n8;

gotoxy(yn8 - 35, xn8 + 3);

cout << "Y" << endl;

}

else if (n8score == 10)

{

maze[xn8 + 5][yn8 - 35] = n8;

gotoxy(yn8 - 35, xn8 + 5);

cout << "Y" << endl;

}

else if (n8score == 11)

{

maze[xn8 + 7][yn8 - 35] = n8;

gotoxy(yn8 - 35, xn8 + 7);

cout << "Y" << endl;

}

else if (n8score == 12)

{

maze[xn8 + 7][yn8 - 41] = n8;

gotoxy(yn8 - 41, xn8 + 7);

cout << "Y" << endl;

}

else if (n8score == 13)

{

maze[xn8 + 7][yn8 - 47] = n8;

gotoxy(yn8 - 47, xn8 + 7);

cout << "Y" << endl;

}

else if (n8score == 14)

{

maze[xn8 + 5][yn8 - 47] = n8;

gotoxy(yn8 - 47, xn8 + 5);

cout << "Y" << endl;

}

else if (n8score == 15)

{

maze[xn8 + 3][yn8 - 47] = n8;

gotoxy(yn8 - 47, xn8 + 3);

cout << "Y" << endl;

}

else if (n8score == 16)

{

maze[xn8 + 1][yn8 - 47] = n8;

gotoxy(yn8 - 47, xn8 + 1);

cout << "Y" << endl;

}

else if (n8score == 17)

{

maze[xn8 - 1][yn8 - 47] = n8;

gotoxy(yn8 - 47, xn8 - 1);

cout << "Y" << endl;

}

else if (n8score == 18)

{

maze[xn8 - 3][yn8 - 47] = n8;

gotoxy(yn8 - 47, xn8 - 3);

cout << "Y" << endl;

}

else if (n8score == 19)

{

maze[xn8 - 5][yn8 - 54] = n8;

gotoxy(yn8 - 54, xn8 - 5);

cout << "Y" << endl;

}

else if (n8score == 20)

{

maze[xn8 - 5][yn8 - 62] = n8;

gotoxy(yn8 - 62, xn8 - 5);

cout << "Y" << endl;

}

else if (n8score == 21)

{

maze[xn8 - 5][yn8 - 69] = n8;

gotoxy(yn8 - 69, xn8 - 5);

cout << "Y" << endl;

}

else if (n8score == 22)

{

maze[xn8 - 5][yn8 - 76] = n8;

gotoxy(yn8 - 76, xn8 - 5);

cout << "Y" << endl;

}

else if (n8score == 23)

{

maze[xn8 - 5][yn8 - 83] = n8;

gotoxy(yn8 - 83, xn8 - 5);

cout << "Y" << endl;

}

else if (n8score == 24)

{

maze[xn8 - 5][yn8 - 90] = n8;

gotoxy(yn8 - 90, xn8 - 5);

cout << "Y" << endl;

}

else if (n8score == 25)

{

maze[xn8 - 7][yn8 - 90] = n8;

gotoxy(yn8 - 90, xn8 - 7);

cout << "Y" << endl;

}

else if (n8score == 26)

{

maze[xn8 - 9][yn8 - 90] = n8;

gotoxy(yn8 - 90, xn8 - 9);

cout << "Y" << endl;

}

else if (n8score == 27)

{

maze[xn8 - 9][yn8 - 83] = n8;

gotoxy(yn8 - 83, xn8 - 9);

cout << "Y" << endl;

}

else if (n8score == 28)

{

maze[xn8 - 9][yn8 - 76] = n8;

gotoxy(yn8 - 76, xn8 - 9);

cout << "Y" << endl;

}

else if (n8score == 29)

{

maze[xn8 - 9][yn8 - 69] = n8;

gotoxy(yn8 - 69, xn8 - 9);

cout << "Y" << endl;

}

else if (n8score == 30)

{

maze[xn8 - 9][yn8 - 62] = n8;

gotoxy(yn8 - 62, xn8 - 9);

cout << "Y" << endl;

}

else if (n8score == 31)

{

maze[xn8 - 9][yn8 - 54] = n8;

gotoxy(yn8 - 54, xn8 - 9);

cout << "Y" << endl;

}

else if (n8score == 32)

{

maze[xn8 - 11][yn8 - 47] = n8;

gotoxy(yn8 - 47, xn8 - 11);

cout << "Y" << endl;

}

else if (n8score == 33)

{

maze[xn8 - 13][yn8 - 47] = n8;

gotoxy(yn8 - 47, xn8 - 13);

cout << "Y" << endl;

}

else if (n8score == 34)

{

maze[xn8 - 15][yn8 - 47] = n8;

gotoxy(yn8 - 47, xn8 - 15);

cout << "Y" << endl;

}

else if (n8score == 35)

{

maze[xn8 - 17][yn8 - 47] = n8;

gotoxy(yn8 - 47, xn8 - 17);

cout << "Y" << endl;

}

else if (n8score == 36)

{

maze[xn8 - 19][yn8 - 47] = n8;

gotoxy(yn8 - 47, xn8 - 19);

cout << "Y" << endl;

}

else if (n8score == 37)

{

maze[xn8 - 21][yn8 - 47] = n8;

gotoxy(yn8 - 47, xn8 - 21);

cout << "Y" << endl;

}

else if (n8score == 38)

{

maze[xn8 - 21][yn8 - 41] = n8;

gotoxy(yn8 - 41, xn8 - 21);

cout << "Y" << endl;

}

else if (n8score == 39)

{

maze[xn8 - 21][yn8 - 35] = n8;

gotoxy(yn8 - 35, xn8 - 21);

cout << "Y" << endl;

}

else if (n8score == 40)

{

maze[xn8 - 19][yn8 - 35] = n8;

gotoxy(yn8 - 35, xn8 - 19);

cout << "Y" << endl;

}

else if (n8score == 41)

{

maze[xn8 - 17][yn8 - 35] = n8;

gotoxy(yn8 - 35, xn8 - 17);

cout << "Y" << endl;

}

else if (n8score == 42)

{

maze[xn8 - 15][yn8 - 35] = n8;

gotoxy(yn8 - 35, xn8 - 15);

cout << "Y" << endl;

}

else if (n8score == 43)

{

maze[xn8 - 13][yn8 - 35] = n8;

gotoxy(yn8 - 35, xn8 - 13);

cout << "Y" << endl;

}

else if (n8score == 44)

{

maze[xn8 - 11][yn8 - 35] = n8;

gotoxy(yn8 - 35, xn8 - 11);

cout << "Y" << endl;

}

else if (n8score == 45)

{

maze[xn8 - 9][yn8 - 28] = n8;

gotoxy(yn8 - 28, xn8 - 9);

cout << "Y" << endl;

}

else if (n8score == 46)

{

maze[xn8 - 9][yn8 - 20] = n8;

gotoxy(yn8 - 20, xn8 - 9);

cout << "Y" << endl;

}

else if (n8score == 47)

{

maze[xn8 - 9][yn8 - 13] = n8;

gotoxy(yn8 - 13, xn8 - 9);

cout << "Y" << endl;

}

else if (n8score == 48)

{

maze[xn8 - 9][yn8 - 6] = n8;

gotoxy(yn8 - 6, xn8 - 9);

cout << "Y" << endl;

}

else if (n8score == 49)

{

maze[xn8 - 9][yn8 + 1] = n8;

gotoxy(yn8 + 1, xn8 - 9);

cout << "Y" << endl;

}

else if (n8score == 50)

{

maze[xn8 - 9][yn8 + 8] = n8;

gotoxy(yn8 + 8, xn8 - 9);

cout << "Y" << endl;

}

else if (n8score == 51)

{

maze[xn8 - 7][yn8 + 8] = n8;

gotoxy(yn8 + 8, xn8 - 7);

cout << "Y" << endl;

}

else if (n8score == 52)

{

maze[xn8 - 7][yn8 + 1] = n8;

gotoxy(yn8 + 1, xn8 - 7);

cout << "Y" << endl;

}

else if (n8score == 53)

{

maze[xn8 - 7][yn8 - 6] = n8;

gotoxy(yn8 - 6, xn8 - 7);

cout << "Y" << endl;

}

else if (n8score == 54)

{

maze[xn8 - 7][yn8 - 13] = n8;

gotoxy(yn8 - 13, xn8 - 7);

cout << "Y" << endl;

}

else if (n8score == 55)

{

maze[xn8 - 7][yn8 - 20] = n8;

gotoxy(yn8 - 20, xn8 - 7);

cout << "Y" << endl;

}

else if (n8score == 56)

{

maze[xn8 - 7][yn8 - 28] = n8;

gotoxy(yn8 - 28, xn8 - 7);

cout << "Y" << endl;

}

else if (n8score == 57)

{

maze[xn8 - 7][yn8 - 35] = n8;

gotoxy(yn8 - 35, xn8 - 7);

cout << "Y" << endl;

}

}