Course Code: CS 3101

Course Title: Rapid Application Development and Visual Programming Technologies

Programming Assignment

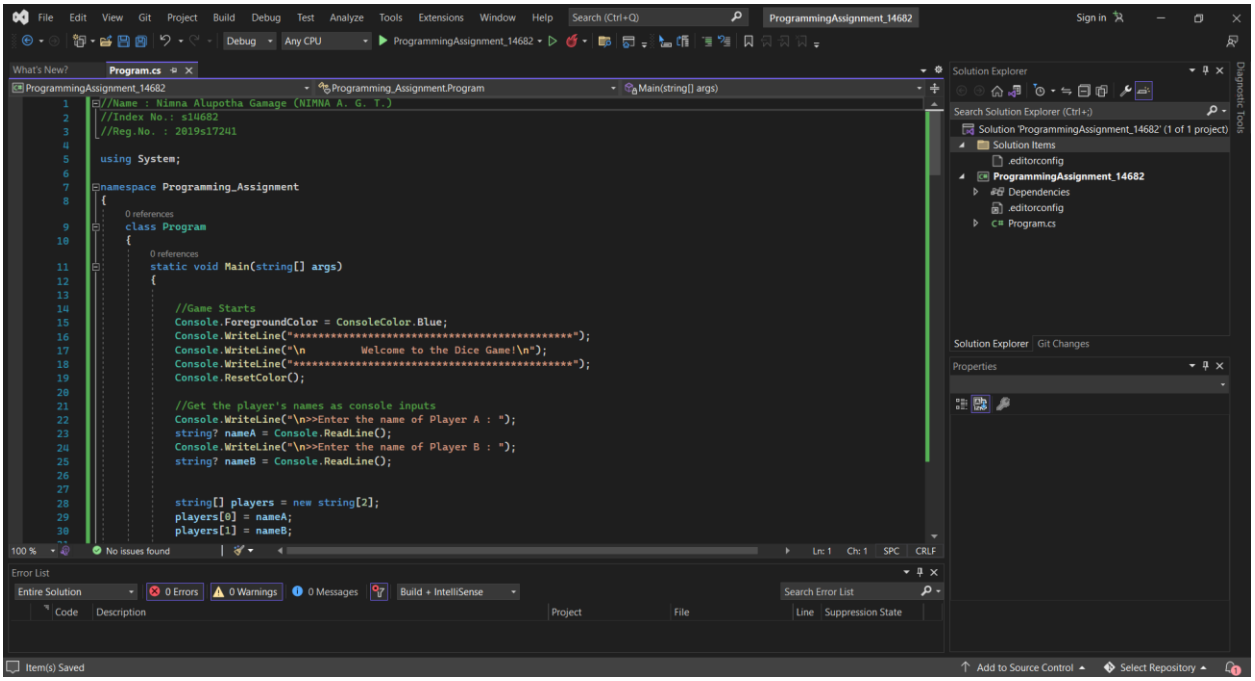
Name : Nimna Alupotha Gamage (NIMNA A. G. T.)

Index No.: 14682

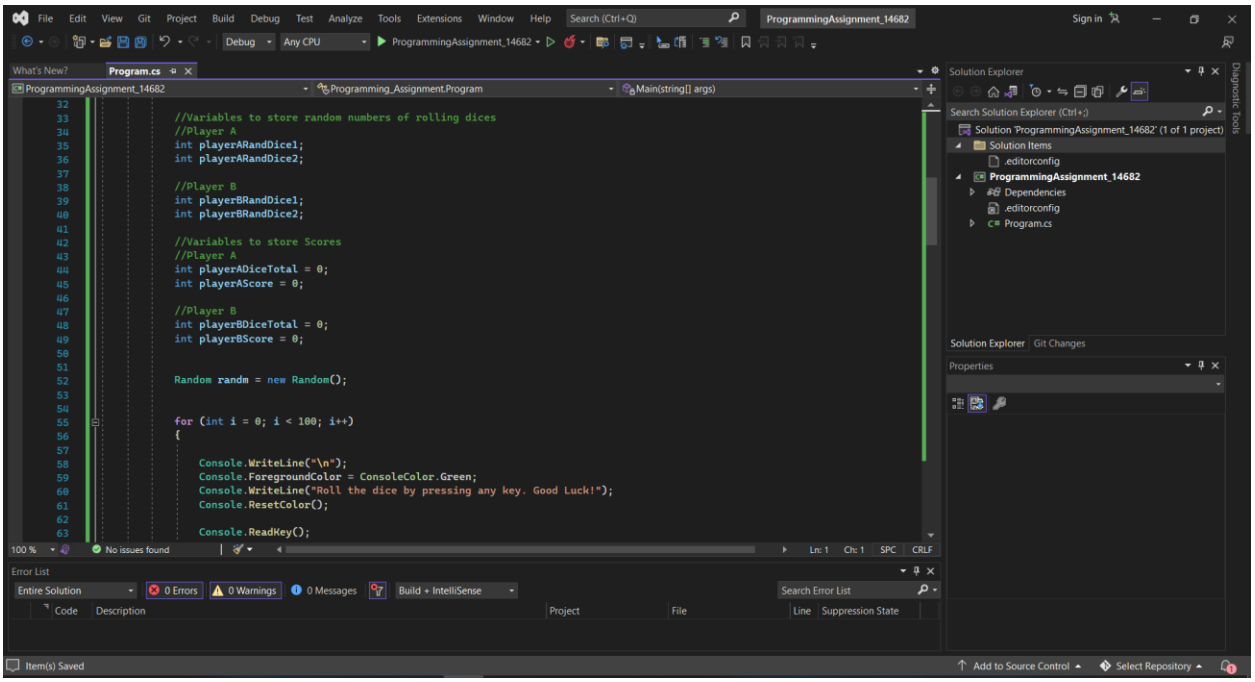
Reg. No. : 2019s17241

Year : 3rd year

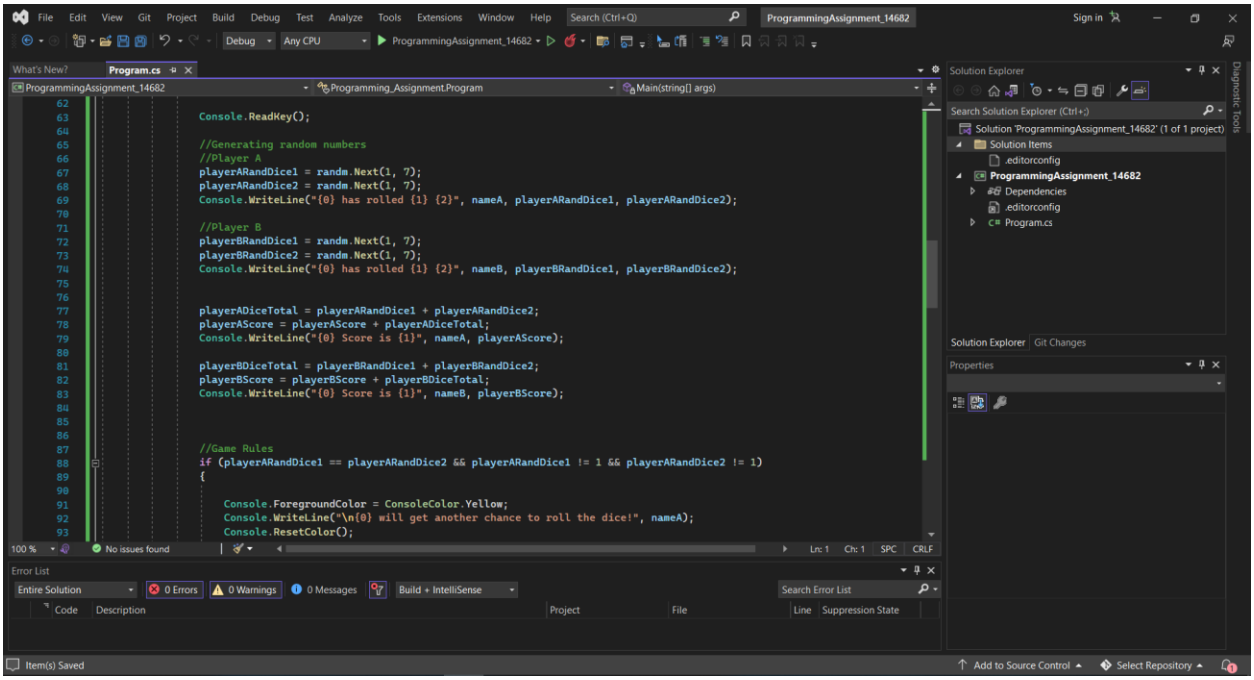
Screenshots of the Visual Studio Code



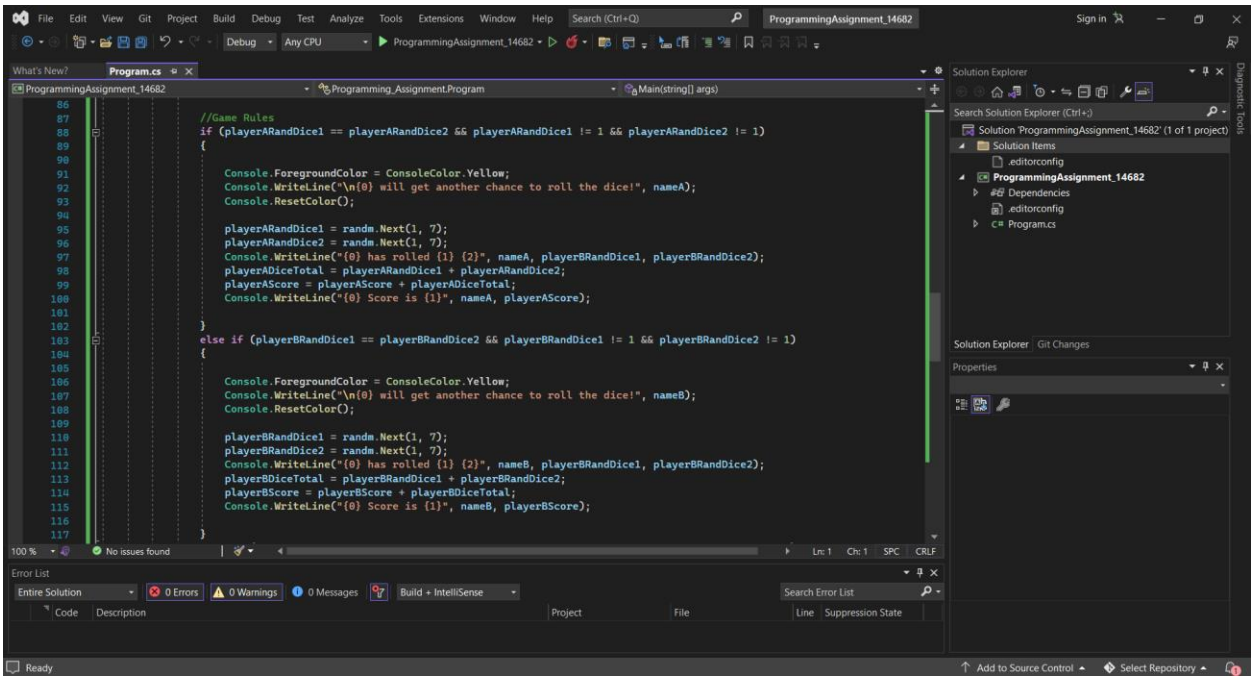
```
//Name : Nimna Alupotha Gamage (NIMNA A. G. T.)
//Index No.: s14682
//Reg.No. : 2019s17241

using System;

namespace Programming_Assignment
{
    0 references
    class Program
    {
        0 references
        static void Main(string[] args)
        {

            //Game Starts
            Console.ForegroundColor = ConsoleColor.Blue;
            Console.WriteLine("*********************************************");
            Console.WriteLine("\n          Welcome to the Dice Game!\n");
            Console.WriteLine("*********************************************");
            Console.ResetColor();

            //Get the player's names as console inputs
            Console.WriteLine("\n>>Enter the name of Player A : ");
            string? nameA = Console.ReadLine();
            Console.WriteLine("\n>>Enter the name of Player B : ");
            string? nameB = Console.ReadLine();


            string[] players = new string[2];
            players[0] = nameA;
            players[1] = nameB;
```



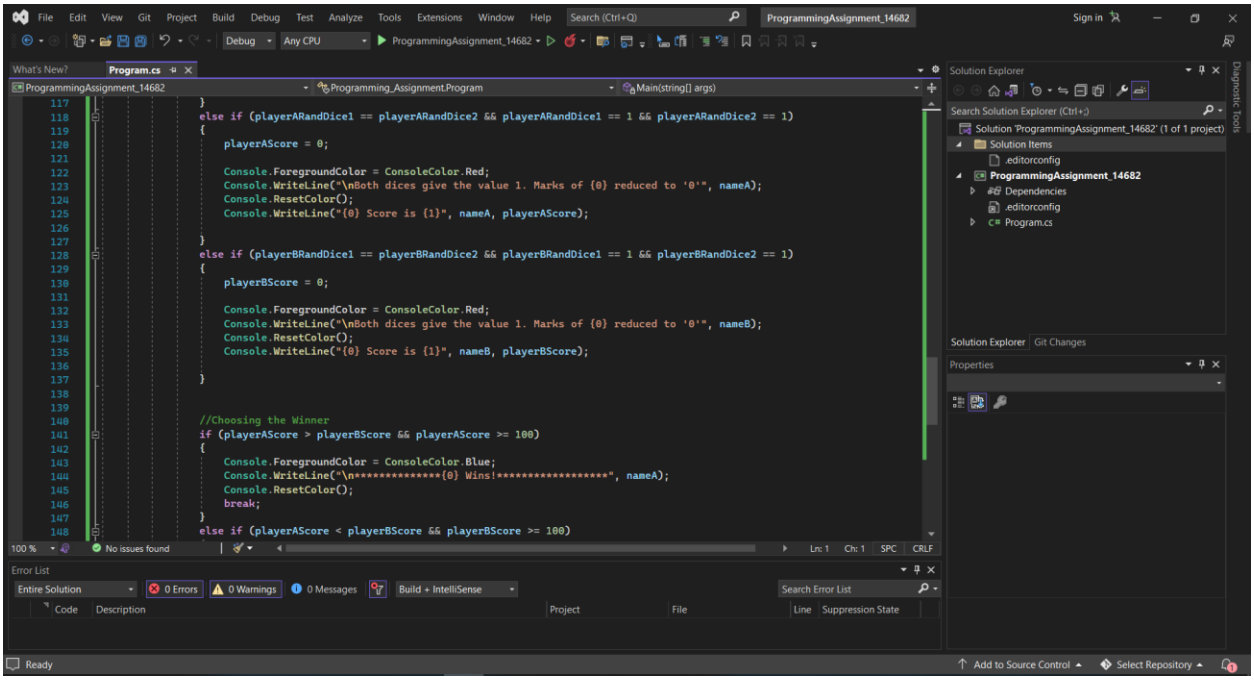
```
            //Variables to store random numbers of rolling dices
            //Player A
            int playerARandDice1;
            int playerARandDice2;

            //Player B
            int playerBRandDice1;
            int playerBRandDice2;

            //Variables to store Scores
            //Player A
            int playerADiceTotal = 0;
            int playerAScore = 0;

            //Player B
            int playerBDiceTotal = 0;
            int playerBScore = 0;


            Random randm = new Random();


            for (int i = 0; i < 100; i++)
            {

                Console.WriteLine("\n");
                Console.ForegroundColor = ConsoleColor.Green;
                Console.WriteLine("Roll the dice by pressing any key. Good Luck!");
                Console.ResetColor();

                Console.ReadKey();
```



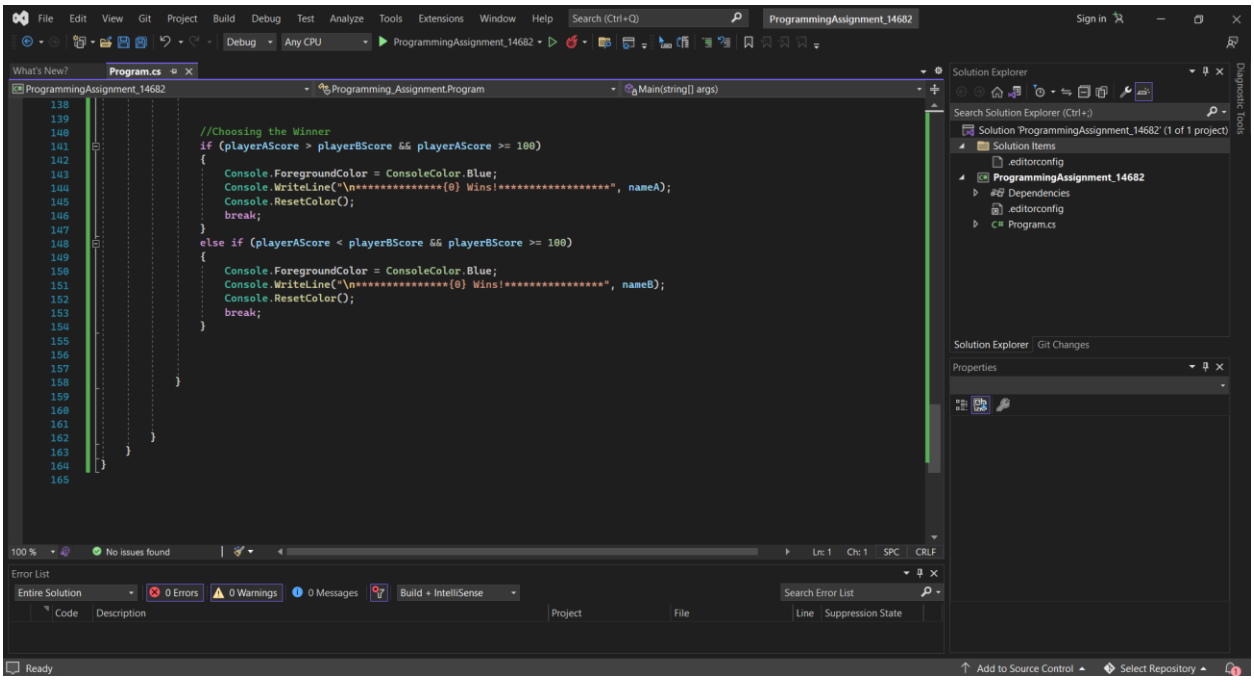
```
Console.ReadKey();

//Generating random numbers
//Player A
playerARandDice1 = randm.Next(1, 7);
playerARandDice2 = randm.Next(1, 7);
Console.WriteLine("{0} has rolled {1} {2}", nameA, playerARandDice1, playerARandDice2);

//Player B
playerBRandDice1 = randm.Next(1, 7);
playerBRandDice2 = randm.Next(1, 7);
Console.WriteLine("{0} has rolled {1} {2}", nameB, playerBRandDice1, playerBRandDice2);


playerADiceTotal = playerARandDice1 + playerARandDice2;
playerAScore = playerAScore + playerADiceTotal;
Console.WriteLine("{0} Score is {1}", nameA, playerAScore);

playerBDiceTotal = playerBRandDice1 + playerBRandDice2;
playerBScore = playerBScore + playerBDiceTotal;
Console.WriteLine("{0} Score is {1}", nameB, playerBScore);



//Game Rules
if (playerARandDice1 == playerARandDice2 && playerARandDice1 != 1 && playerARandDice2 != 1)
{

    Console.ForegroundColor = ConsoleColor.Yellow;
    Console.WriteLine("\n{0} will get another chance to roll the dice!", nameA);
    Console.ResetColor();
```



```
//Game Rules
if (playerARandDice1 == playerARandDice2 && playerARandDice1 != 1 && playerARandDice2 != 1)
{

    Console.ForegroundColor = ConsoleColor.Yellow;
    Console.WriteLine("\n{0} will get another chance to roll the dice!", nameA);
    Console.ResetColor();

    playerARandDice1 = randm.Next(1, 7);
    playerARandDice2 = randm.Next(1, 7);
    Console.WriteLine("{0} has rolled {1} {2}", nameA, playerBRandDice1, playerBRandDice2);
    playerADiceTotal = playerARandDice1 + playerARandDice2;
    playerAScore = playerAScore + playerADiceTotal;
    Console.WriteLine("{0} Score is {1}", nameA, playerAScore);

}
else if (playerBRandDice1 == playerBRandDice2 && playerBRandDice1 != 1 && playerBRandDice2 != 1)
{

    Console.ForegroundColor = ConsoleColor.Yellow;
    Console.WriteLine("\n{0} will get another chance to roll the dice!", nameB);
    Console.ResetColor();

    playerBRandDice1 = randm.Next(1, 7);
    playerBRandDice2 = randm.Next(1, 7);
    Console.WriteLine("{0} has rolled {1} {2}", nameB, playerBRandDice1, playerBRandDice2);
    playerBDiceTotal = playerBRandDice1 + playerBRandDice2;
    playerBScore = playerBScore + playerBDiceTotal;
    Console.WriteLine("{0} Score is {1}", nameB, playerBScore);

}
```



```
        }
        else if (playerARandDice1 == playerARandDice2 && playerARandDice1 == 1 && playerARandDice2 == 1)
        {
            playerAScore = 0;

            Console.ForegroundColor = ConsoleColor.Red;
            Console.WriteLine("\nBoth dices give the value 1. Marks of {0} reduced to '0'", nameA);
            Console.ResetColor();
            Console.WriteLine("{0} Score is {1}", nameA, playerAScore);

        }
        else if (playerBRandDice1 == playerBRandDice2 && playerBRandDice1 == 1 && playerBRandDice2 == 1)
        {
            playerBScore = 0;

            Console.ForegroundColor = ConsoleColor.Red;
            Console.WriteLine("\nBoth dices give the value 1. Marks of {0} reduced to '0'", nameB);
            Console.ResetColor();
            Console.WriteLine("{0} Score is {1}", nameB, playerBScore);

        }


        //Choosing the Winner
        if (playerAScore > playerBScore && playerAScore >= 100)
        {
            Console.ForegroundColor = ConsoleColor.Blue;
            Console.WriteLine("\n***************{0} Wins!******************", nameA);
            Console.ResetColor();
            break;
        }
        else if (playerAScore < playerBScore && playerBScore >= 100)
```



```
        //Choosing the Winner
        if (playerAScore > playerBScore && playerAScore >= 100)
        {
            Console.ForegroundColor = ConsoleColor.Blue;
            Console.WriteLine("\n***************{0} Wins!******************", nameA);
            Console.ResetColor();
            break;
        }
        else if (playerAScore < playerBScore && playerBScore >= 100)
        {
            Console.ForegroundColor = ConsoleColor.Blue;
            Console.WriteLine("\n***************{0} Wins!***************", nameB);
            Console.ResetColor();
            break;
        }



    }


}
}
}
```

Complete code of the Dice Game

```
using System;

namespace Programming_Assignment
{
    class Program
    {
        static void Main(string[] args)
        {

            //Game Starts
            Console.ForegroundColor = ConsoleColor.Blue;
            Console.WriteLine("*****************************************");
            Console.WriteLine("\n         Welcome to the Dice Game!\n");
            Console.WriteLine("*****************************************");
            Console.ResetColor();

            //Get the player's names as console inputs
            Console.WriteLine("\n>>Enter the name of Player A : ");
            string? nameA = Console.ReadLine();
            Console.WriteLine("\n>>Enter the name of Player B : ");
            string? nameB = Console.ReadLine();


            string[] players = new string[2];
            players[0] = nameA;
            players[1] = nameB;


            //Variables to store random numbers of rolling dices
            //Player A
            int playerARandDice1;
            int playerARandDice2;

            //Player B
            int playerBRandDice1;
            int playerBRandDice2;

            //Variables to store Scores
            //Player A
            int playerADiceTotal = 0;
            int playerAScore = 0;

            //Player B
            int playerBDiceTotal = 0;
            int playerBScore = 0;


            Random randm = new Random();


            for (int i = 0; i < 100; i++)
            {
```

```
                Console.WriteLine("\n");
                Console.ForegroundColor = ConsoleColor.Green;
                Console.WriteLine("Roll the dice by pressing any key. Good Luck!");
                Console.ResetColor();

                Console.ReadKey();

                //Generating random numbers
                //Player A
                playerARandDice1 = randm.Next(1, 7);
                playerARandDice2 = randm.Next(1, 7);
                Console.WriteLine("{0} has rolled {1} {2}", nameA, playerARandDice1,
playerARandDice2);

                //Player B
                playerBRandDice1 = randm.Next(1, 7);
                playerBRandDice2 = randm.Next(1, 7);
                Console.WriteLine("{0} has rolled {1} {2}", nameB, playerBRandDice1,
playerBRandDice2);


                playerADiceTotal = playerARandDice1 + playerARandDice2;
                playerAScore = playerAScore + playerADiceTotal;
                Console.WriteLine("{0} Score is {1}", nameA, playerAScore);

                playerBDiceTotal = playerBRandDice1 + playerBRandDice2;
                playerBScore = playerBScore + playerBDiceTotal;
                Console.WriteLine("{0} Score is {1}", nameB, playerBScore);



                //Game Rules
                if (playerARandDice1 == playerARandDice2 && playerARandDice1 != 1 &&
playerARandDice2 != 1)
                {

                    Console.ForegroundColor = ConsoleColor.Yellow;
                    Console.WriteLine("\n{0} will get another chance to roll the
dice!", nameA);
                    Console.ResetColor();

                    playerARandDice1 = randm.Next(1, 7);
                    playerARandDice2 = randm.Next(1, 7);
                    Console.WriteLine("{0} has rolled {1} {2}", nameA,
playerBRandDice1, playerBRandDice2);
                    playerADiceTotal = playerARandDice1 + playerARandDice2;
                    playerAScore = playerAScore + playerADiceTotal;
                    Console.WriteLine("{0} Score is {1}", nameA, playerAScore);

                }
                else if (playerBRandDice1 == playerBRandDice2 && playerBRandDice1 !=
1 && playerBRandDice2 != 1)
                {

                    Console.ForegroundColor = ConsoleColor.Yellow;
                    Console.WriteLine("\n{0} will get another chance to roll the
dice!", nameB);
```

```
                    Console.ResetColor();

                    playerBRandDice1 = randm.Next(1, 7);
                    playerBRandDice2 = randm.Next(1, 7);
                    Console.WriteLine("{0} has rolled {1} {2}", nameB,
playerBRandDice1, playerBRandDice2);
                    playerBDiceTotal = playerBRandDice1 + playerBRandDice2;
                    playerBScore = playerBScore + playerBDiceTotal;
                    Console.WriteLine("{0} Score is {1}", nameB, playerBScore);

                }
                else if (playerARandDice1 == playerARandDice2 && playerARandDice1 ==
1 && playerARandDice2 == 1)
                {
                    playerAScore = 0;

                    Console.ForegroundColor = ConsoleColor.Red;
                    Console.WriteLine("\nBoth dices give the value 1. Marks of {0}
reduced to '0'", nameA);
                    Console.ResetColor();
                    Console.WriteLine("{0} Score is {1}", nameA, playerAScore);

                }
                else if (playerBRandDice1 == playerBRandDice2 && playerBRandDice1 ==
1 && playerBRandDice2 == 1)
                {
                    playerBScore = 0;

                    Console.ForegroundColor = ConsoleColor.Red;
                    Console.WriteLine("\nBoth dices give the value 1. Marks of {0}
reduced to '0'", nameB);
                    Console.ResetColor();
                    Console.WriteLine("{0} Score is {1}", nameB, playerBScore);

                }


                //Choosing the Winner
                if (playerAScore > playerBScore && playerAScore >= 100)
                {
                    Console.ForegroundColor = ConsoleColor.Blue;
                    Console.WriteLine("\n***************{0} Wins!******************",
nameA);
                    Console.ResetColor();
                    break;
                }
                else if (playerAScore < playerBScore && playerBScore >= 100)
                {
                    Console.ForegroundColor = ConsoleColor.Blue;
                    Console.WriteLine("\n****************{0} Wins!****************",
nameB);
                    Console.ResetColor();
                    break;
                }
```

```
            }


        }
    }
}
```

The approach for the Dice Game

A Roll a Dice Game has developed using C# (console application) according to the given game rules.

According to the specified game rules;

- This dice game can be played by two players using two dices.
- Each player will get a chance to roll two dices at the same time.
- The winning score is 100 and the game automatically terminates if player A or player B get 100 or more than 100 marks.
- The player, whose winning score is greater than or equal to 100 will be the winner.
- If 02 dices give the same result (except the dice value 1), the same player will get another chance to roll the dice. If not, the other player will get the chance to roll 02 dices.
- When calculating the score, if both dices give the value 1, the individual marks of the respective player will be reduced to 0. If not, sum of the two dice values will be added to the individual scores of the respective player.

The explanation of the code and the game logic

When the player run the program, the console output will give a welcome message as “Welcome to the Dice Game!”. It will appear in blue colour font.

```
using System;

namespace Programming_Assignment
{
    class Program
    {
        static void Main(string[] args)
        {

            //Game Starts
            Console.ForegroundColor = ConsoleColor.Blue;
            Console.WriteLine("*****************************************");
            Console.WriteLine("\n         Welcome to the Dice Game!\n");
            Console.WriteLine("*****************************************");
            Console.ResetColor();
```

Name of the players can be given as console inputs through the following code.

Console output will appear as “Enter the name of Player A : ”. Player A should type his/her name and press the enter key. Then the Console output will appear as “Enter the name of Player B : ”. Player B should type his/her name and press the enter key.

The names that are given as the console inputs will be assigned to the respective players.

```
//Get the player's names as console inputs
            Console.WriteLine("\n>>Enter the name of Player A : ");
            string? nameA = Console.ReadLine();
            Console.WriteLine("\n>>Enter the name of Player B : ");
            string? nameB = Console.ReadLine();

string[] players = new string[2];
            players[0] = nameA;
            players[1] = nameB;
```

Following are the variable that are used to store different values.

The player A rolled the two dices and the generated random numbers are stored in two integer type variables called; ‘playerARandDice1’ and ‘playerARandDice2’.

The player B rolled the two dices and the generated random numbers are stored in two integer type variables called; ‘playerBRandDice1’ and ‘playerBRandDice2’.

Score of player A is stored in ‘playerADiceTotal’ and ‘playerAScore’ integer type variables.

Score of player B is stored in ‘playerBDiceTotal’ and ‘playerBScore’ integer type variables.

```
//Variables to store random numbers of rolling dices
//Player A
int playerARandDice1;
int playerARandDice2;

//Player B
int playerBRandDice1;
int playerBRandDice2;

//Variables to store Scores
//Player A
int playerADiceTotal = 0;
int playerAScore = 0;

//Player B
int playerBDiceTotal = 0;
int playerBScore = 0;
```

Following code is used to initialize the instances of the Random class.

```
Random randm = new Random();
```

All the game rules are included within a for loop with 100 iterations. 100 iterations are more than enough to complete the game as mostly the game will terminate with 15 or 20 iterations. Modifications can be done using other types of loops as well. E.g. while loop.

```
for (int i = 0; i < 100; i++){}
```

After the Player B entered his/her name to the console input, A message will be appeared as “Roll the dice by pressing any key. Good Luck!”. The game can be started by pressing any key on the keyboard.

```
Console.WriteLine("\n");
Console.ForegroundColor = ConsoleColor.Green;
Console.WriteLine("Roll the dice by pressing any key. Good Luck!");
Console.ResetColor();

Console.ReadKey();
```

Following code block is used to generate random numbers from two dices. Random numbers including one and excluding 7 will be generated. Each player will get a chance to roll two dices at the same time.

```
                //Generating random numbers
                //Player A
                playerARandDice1 = randm.Next(1, 7);
                playerARandDice2 = randm.Next(1, 7);
                Console.WriteLine("{0} has rolled {1} {2}", nameA, playerARandDice1,
playerARandDice2);

                //Player B
                playerBRandDice1 = randm.Next(1, 7);
                playerBRandDice2 = randm.Next(1, 7);
                Console.WriteLine("{0} has rolled {1} {2}", nameB, playerBRandDice1,
playerBRandDice2);
```

In each iteration the score is printed on the console through following code block.

```
    playerADiceTotal = playerARandDice1 + playerARandDice2;
    playerAScore = playerAScore + playerADiceTotal;
    Console.WriteLine("{0} Score is {1}", nameA, playerAScore);

    playerBDiceTotal = playerBRandDice1 + playerBRandDice2;
    playerBScore = playerBScore + playerBDiceTotal;
    Console.WriteLine("{0} Score is {1}", nameB, playerBScore);
```

The most important part of any game is its rules.

One of the rules of this game is “If 02 dices give the same result (except the dice value 1), the same player will get another chance to roll the dice. Else other player gets the chance to roll 02 dices.”

If clause is used to imitate that behaviour of this game. As && operator is used, all the following conditions must be true for the execution of the code block inside the if clause.

- The generated random numbers of two dices should be equal.
- But the random numbers of two dices should not be equal to ‘1’.

The if block will be executed only if the above both conditions are true. Therefore, if above conditions are true, the if clause will be executed and will give another chance to the same player. If both conditions are false or one condition is false, the other player will get a chance to roll the dices as in the case of normal game behaviour.

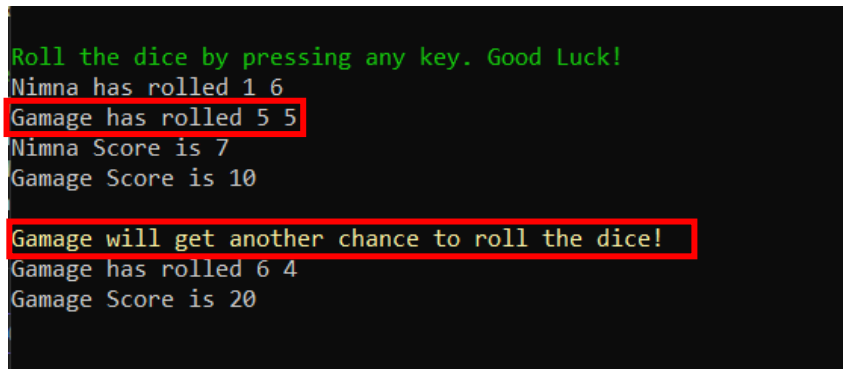
Player A

```
//Game Rules
                if (playerARandDice1 == playerARandDice2 && playerARandDice1 != 1 &&
playerARandDice2 != 1)
                {

                    Console.ForegroundColor = ConsoleColor.Yellow;
                    Console.WriteLine("\n{0} will get another chance to roll the
dice!", nameA);
                    Console.ResetColor();

                    playerARandDice1 = randm.Next(1, 7);
                    playerARandDice2 = randm.Next(1, 7);
                    Console.WriteLine("{0} has rolled {1} {2}", nameA,
playerBRandDice1, playerBRandDice2);
                    playerADiceTotal = playerARandDice1 + playerARandDice2;
                    playerAScore = playerAScore + playerADiceTotal;
                    Console.WriteLine("{0} Score is {1}", nameA, playerAScore);

                }
```

Player B

```
else if (playerBRandDice1 == playerBRandDice2 && playerBRandDice1 != 1 &&
playerBRandDice2 != 1)
                {

                    Console.ForegroundColor = ConsoleColor.Yellow;
                    Console.WriteLine("\n{0} will get another chance to roll the
dice!", nameB);
                    Console.ResetColor();

                    playerBRandDice1 = randm.Next(1, 7);
                    playerBRandDice2 = randm.Next(1, 7);
                    Console.WriteLine("{0} has rolled {1} {2}", nameB,
playerBRandDice1, playerBRandDice2);
                    playerBDiceTotal = playerBRandDice1 + playerBRandDice2;
                    playerBScore = playerBScore + playerBDiceTotal;
                    Console.WriteLine("{0} Score is {1}", nameB, playerBScore);

                }
```

An example console output is given below;



Another rule of this game is;

“Calculation of score: If both dices give the value 1, the individual marks of the respective player will be reduced to 0. Else sum of the two dice values will be added to individual scores of the respective player.”

Following conditions must be true to execute this else if code block.

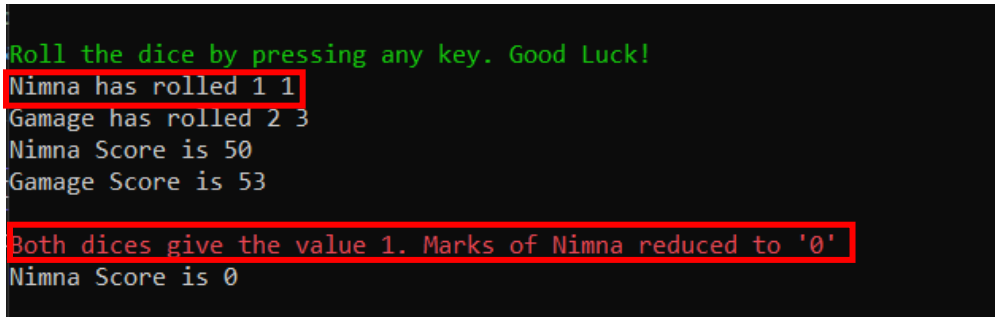
- Both dices should give the same value
- That value must be equal to ‘1’.

As && operator is used in the middle, both conditions must be true. If not (if only one condition is true or both are false), the following code block will not be executed. If both conditions are true, the total mark of the respective player will be reduced to ‘0’. Unless the sum of the two dice values will be added to total scores of the respective player as in the case of normal scenario.

Player A

```
else if (playerARandDice1 == playerARandDice2 && playerARandDice1 == 1 &&
playerARandDice2 == 1)
                {
                    playerAScore = 0;

                    Console.ForegroundColor = ConsoleColor.Red;
                    Console.WriteLine("\nBoth dices give the value 1. Marks of {0}
reduced to '0'", nameA);
                    Console.ResetColor();
                    Console.WriteLine("{0} Score is {1}", nameA, playerAScore);

                }

Player B

                else if (playerBRandDice1 == playerBRandDice2 && playerBRandDice1 ==
1 && playerBRandDice2 == 1)
                {
                    playerBScore = 0;

                    Console.ForegroundColor = ConsoleColor.Red;
                    Console.WriteLine("\nBoth dices give the value 1. Marks of {0}
reduced to '0'", nameB);
                    Console.ResetColor();
                    Console.WriteLine("{0} Score is {1}", nameB, playerBScore);

                }
```

An example console output is given below;

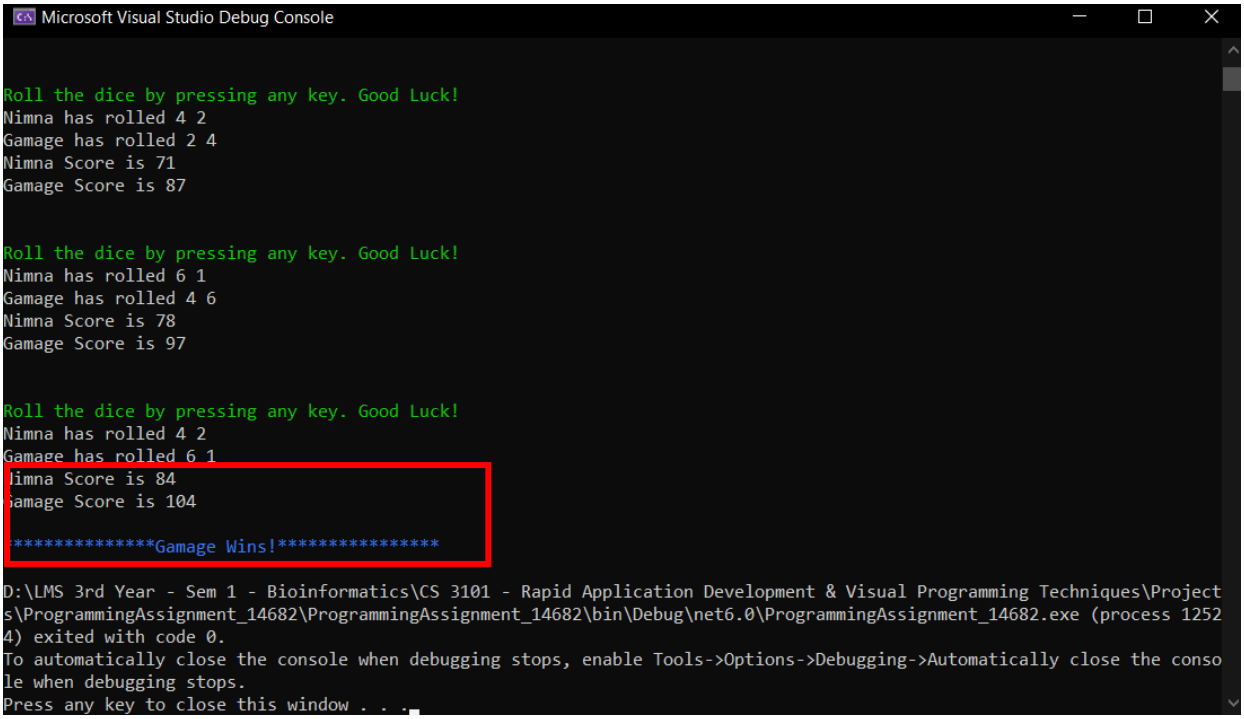


The winning score of this game is '100'. The player who first reaches the winning score will be the winner of the game. When one player reaches the winning score, the game will automatically terminate and give the name of the winner.

Another if clause is used here. If the score of the player A is greater that the score of the player B and if the score of the player A is greater than or equal to winning score 100; the player A will be the winner. If the score of the player B is greater that the score of the player A and if the score of the player B is greater than or equal to the winning score 100; the player B will be the winner.

```
//Choosing the Winner
                if (playerAScore > playerBScore && playerAScore >= 100)
                {
                    Console.ForegroundColor = ConsoleColor.Blue;
                    Console.WriteLine("\n***************{0} Wins!******************",
nameA);
                    Console.ResetColor();
                    break;
                }
                else if (playerAScore < playerBScore && playerBScore >= 100)
                {
                    Console.ForegroundColor = ConsoleColor.Blue;
                    Console.WriteLine("\n***************{0} Wins!***************",
nameB);
                    Console.ResetColor();
                    break;
                }
```

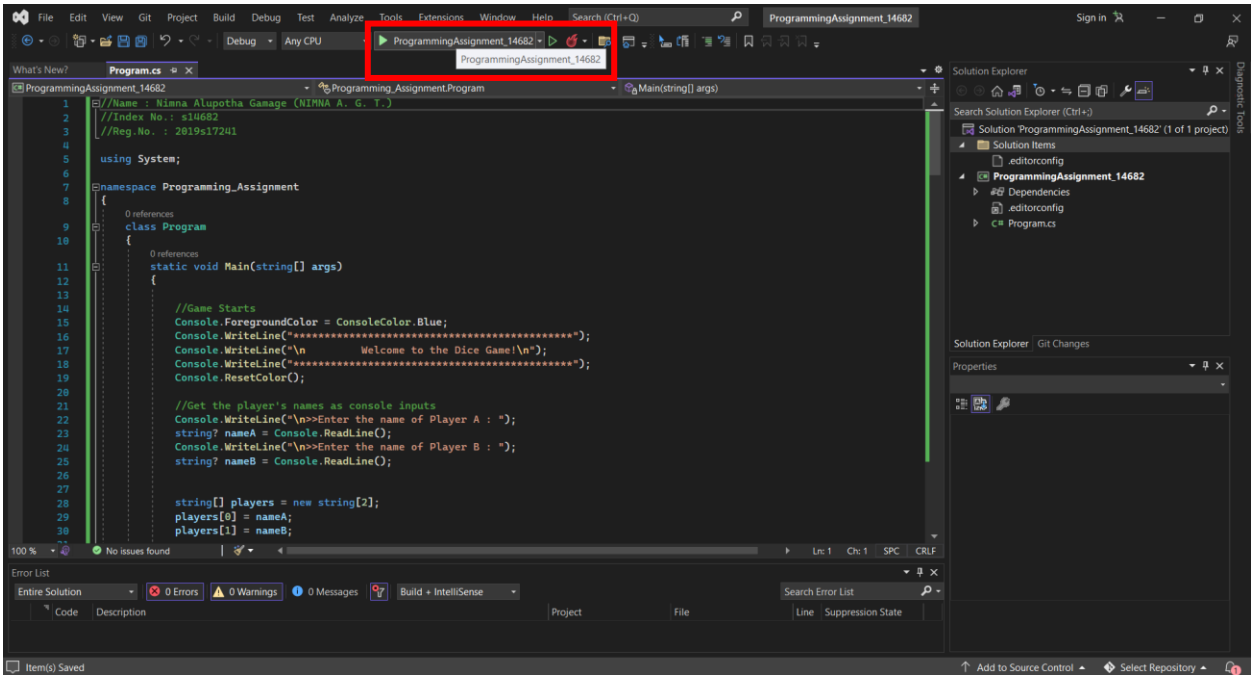
An example console output is given below;



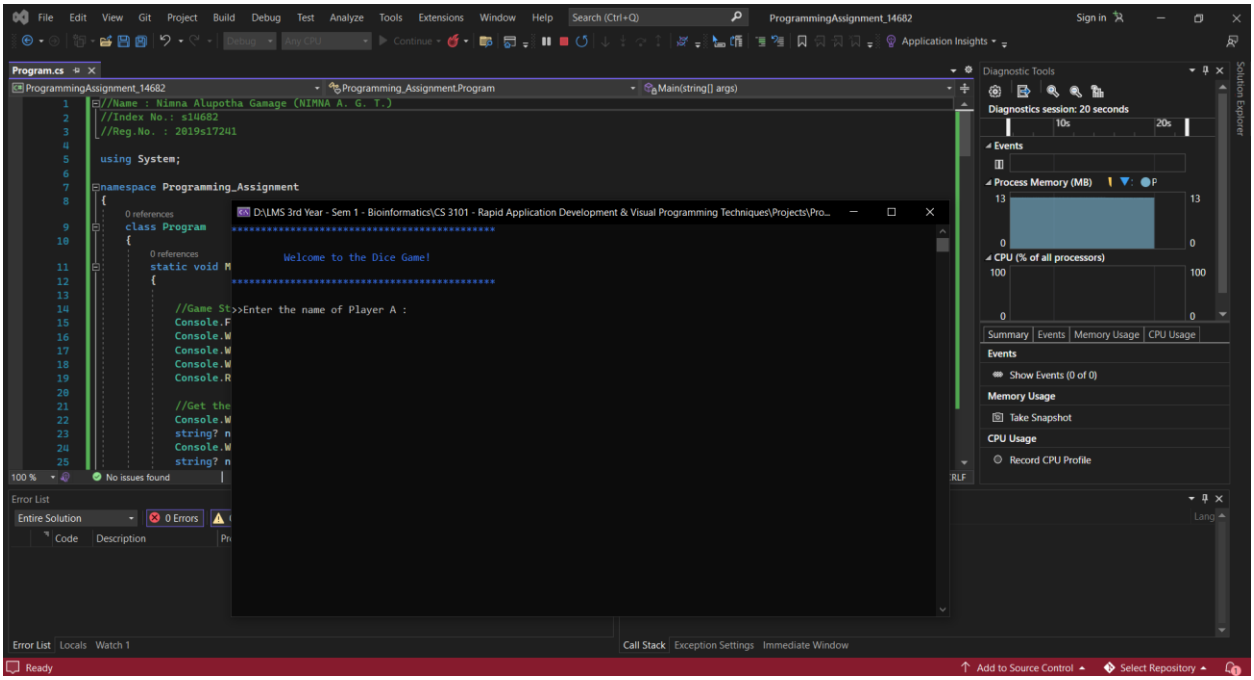
Instructions to play the game with appropriate screenshots

The steps to follow to play the dice game will explained with screenshots of one round of the game.

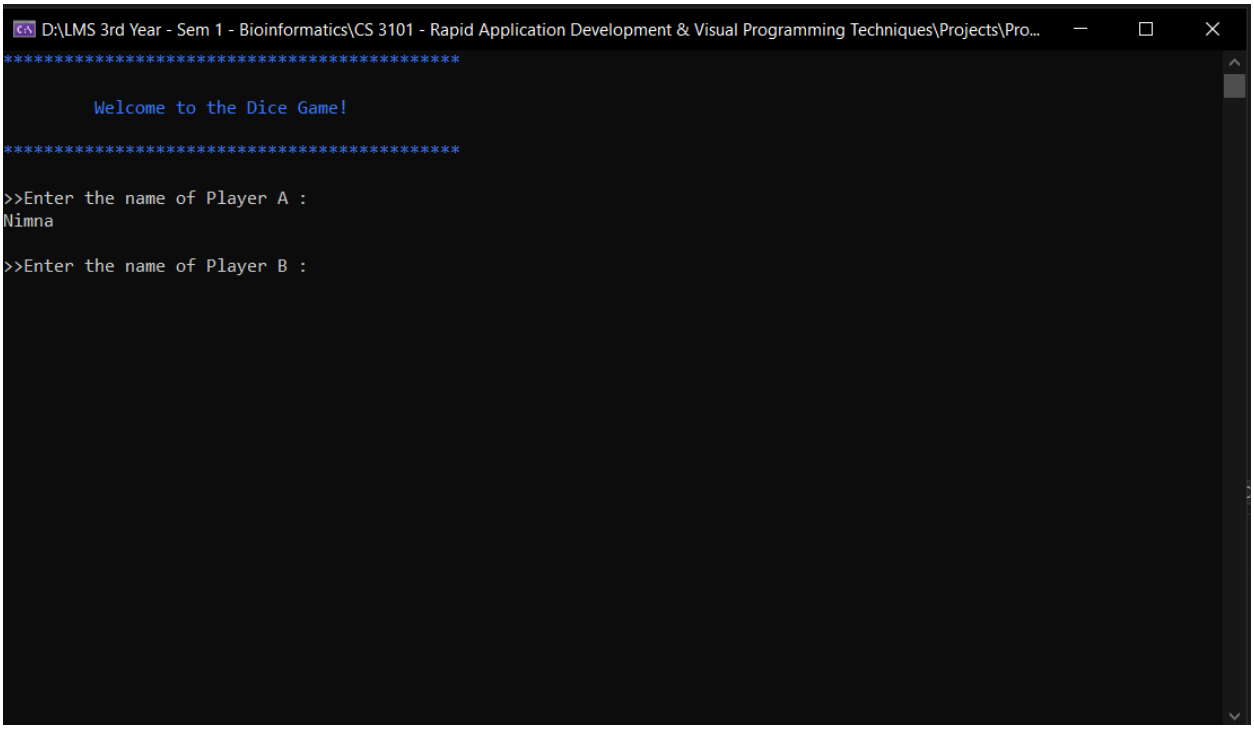
Step 1: The players should run the console application.

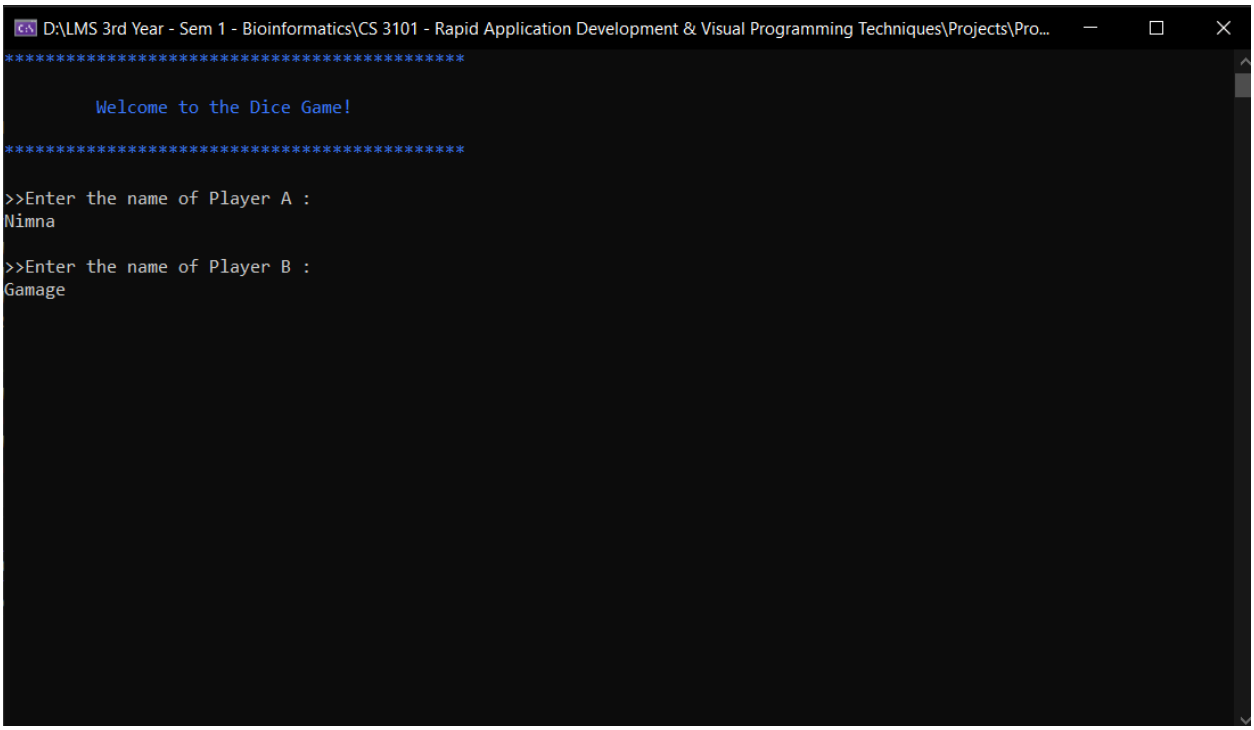


Step 2: The name of the player A should be entered.

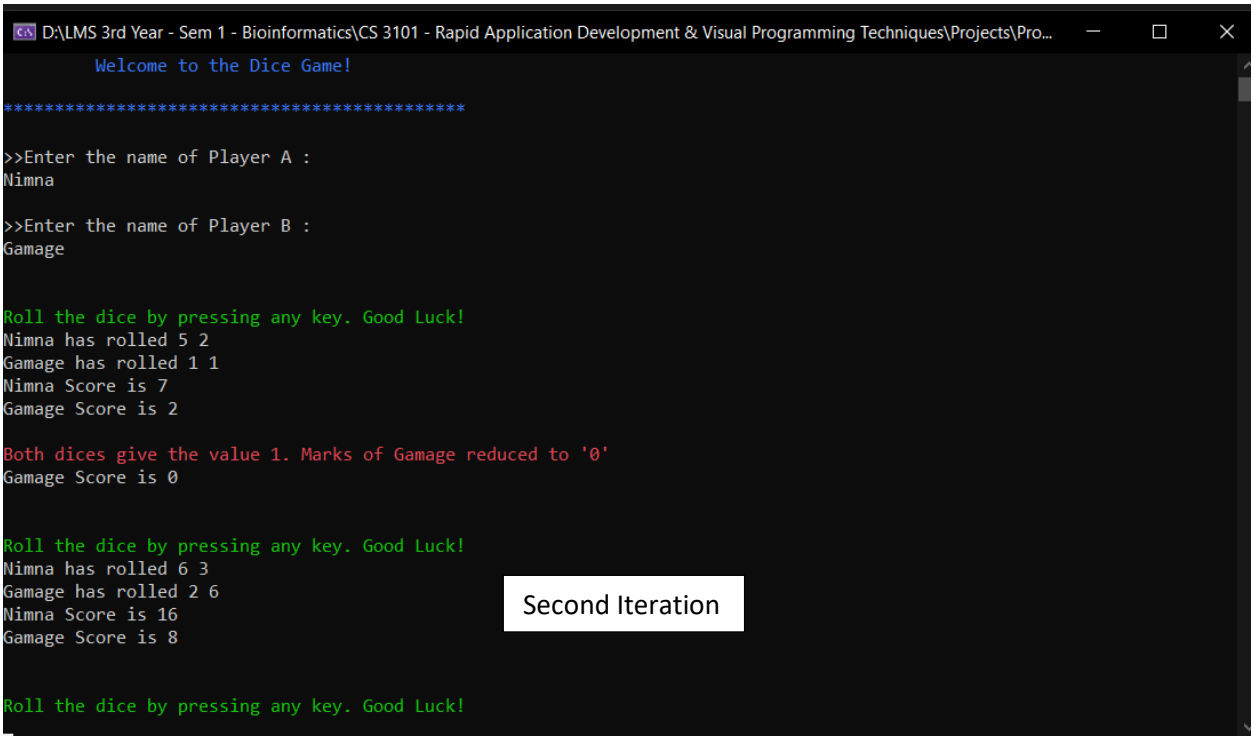


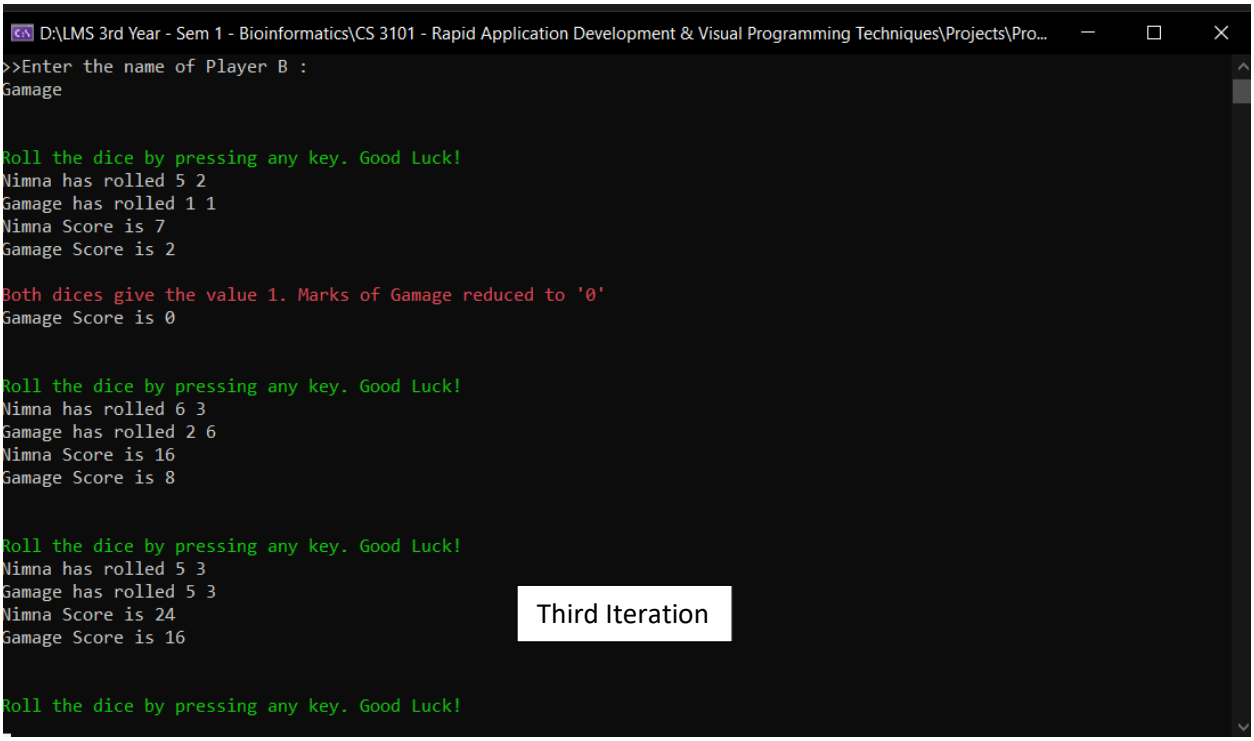
Step 3: The name of the player B should be entered.



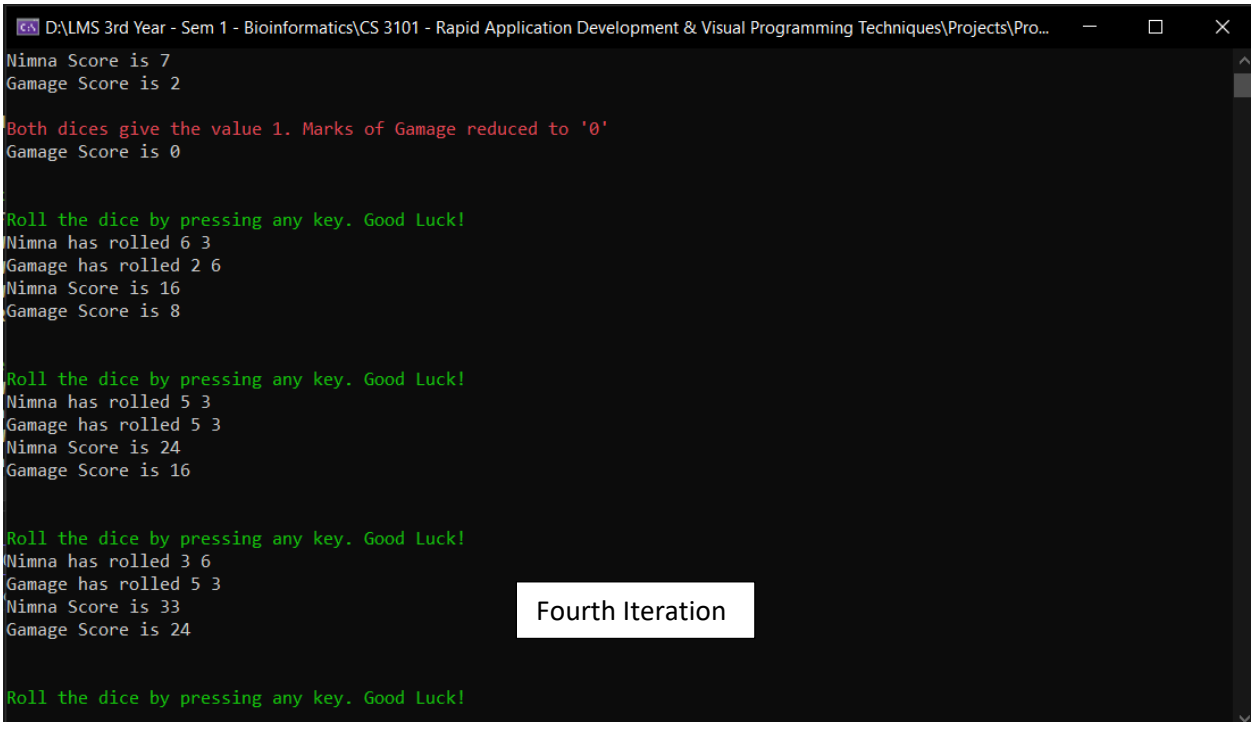


Step 4: Roll the dices by pressing any key in the keyboard. This will be processed until one player reaches the winning score.

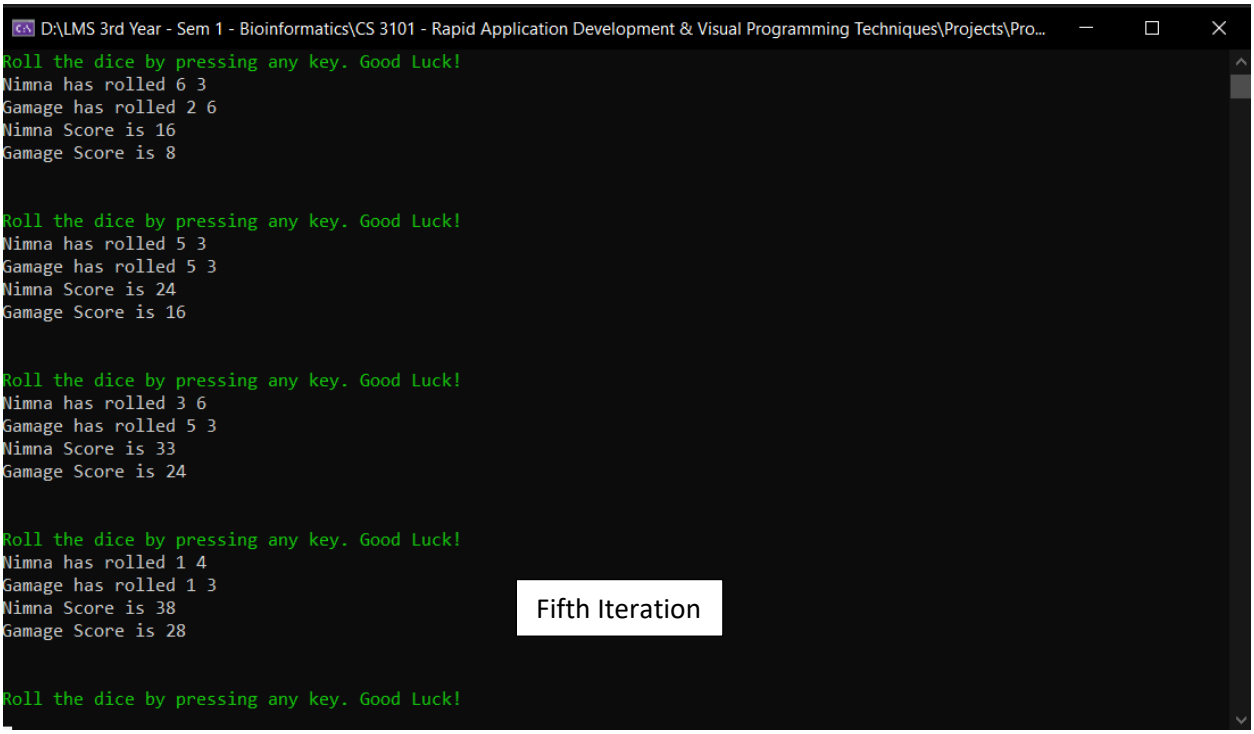


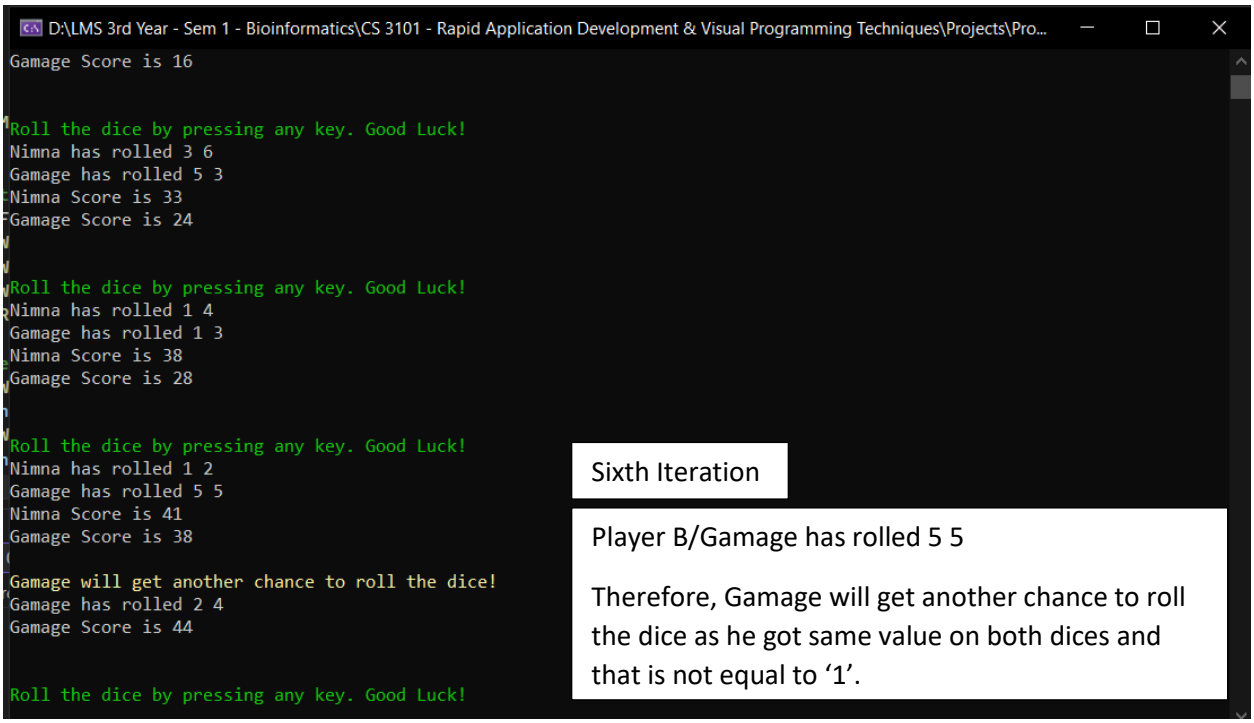


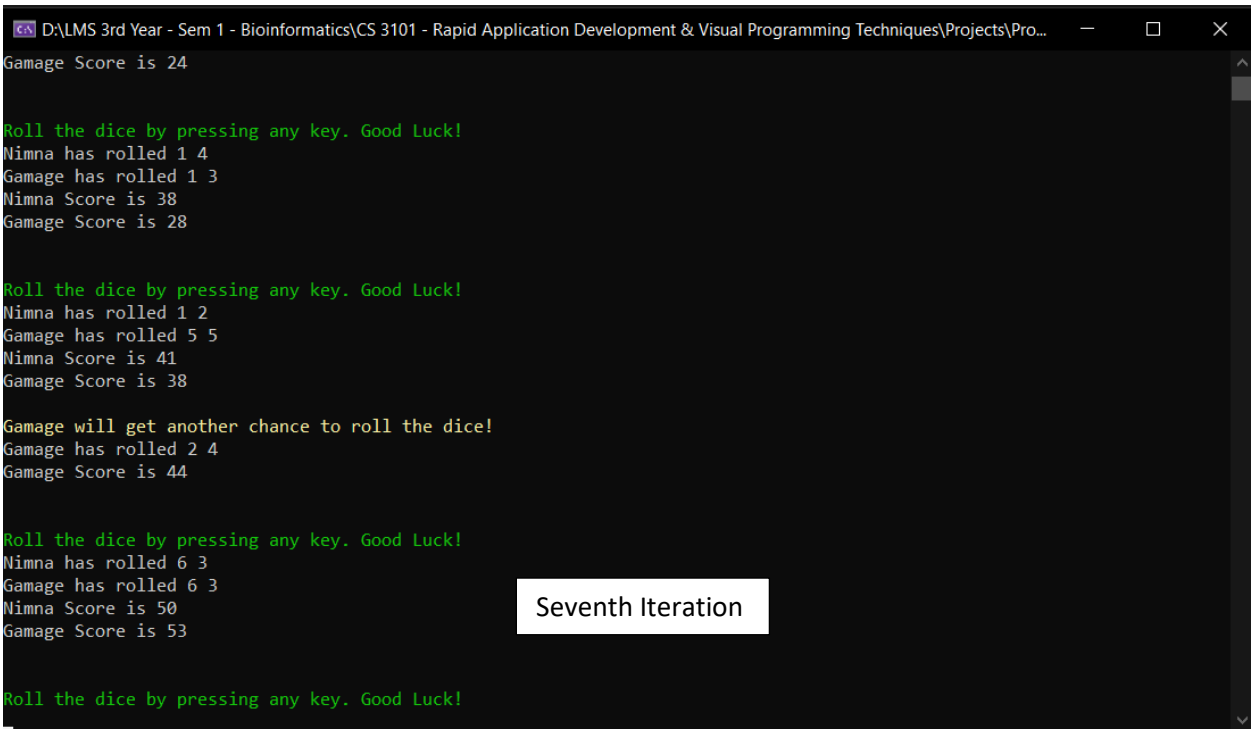
Third Iteration



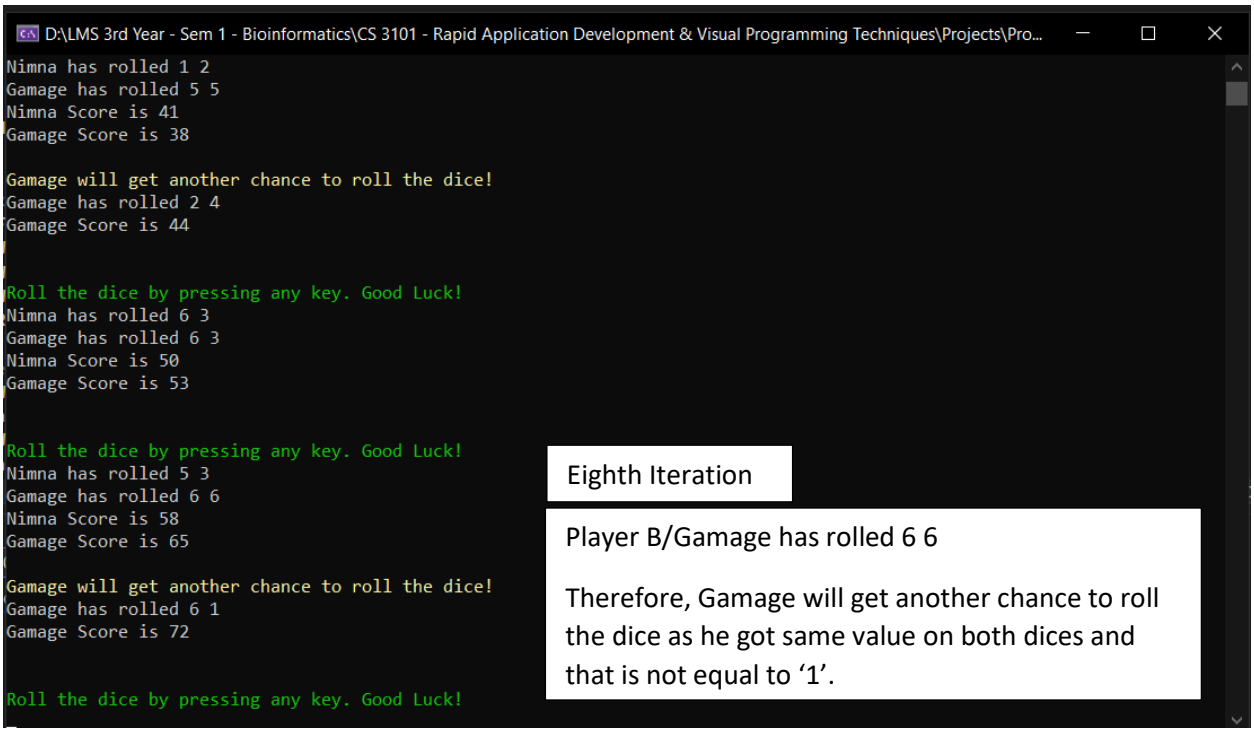
Fourth Iteration


D:\LMS 3rd Year - Sem 1 - Bioinformatics\CS 3101 - Rapid Application Development & Visual Programming Techniques\Projects\Pro...
Roll the dice by pressing any key. Good Luck!
Nimna has rolled 6 3
Gamage has rolled 2 6
Nimna Score is 16
Gamage Score is 8
Roll the dice by pressing any key. Good Luck!
Nimna has rolled 5 3
Gamage has rolled 5 3
Nimna Score is 24
Gamage Score is 16
Roll the dice by pressing any key. Good Luck!
Nimna has rolled 3 6
Gamage has rolled 5 3
Nimna Score is 33
Gamage Score is 24
Roll the dice by pressing any key. Good Luck!
Nimna has rolled 1 4
Gamage has rolled 1 3
Nimna Score is 38
Gamage Score is 28
Roll the dice by pressing any key. Good Luck!
Fifth Iteration


D:\LMS 3rd Year - Sem 1 - Bioinformatics\CS 3101 - Rapid Application Development & Visual Programming Techniques\Projects\Pro...
Gamage Score is 16
Roll the dice by pressing any key. Good Luck!
Nimna has rolled 3 6
Gamage has rolled 5 3
Nimna Score is 33
Gamage Score is 24
Roll the dice by pressing any key. Good Luck!
Nimna has rolled 1 4
Gamage has rolled 1 3
Nimna Score is 38
Gamage Score is 28
Roll the dice by pressing any key. Good Luck!
Nimna has rolled 1 2
Gamage has rolled 5 5
Nimna Score is 41
Gamage Score is 38
Gamage will get another chance to roll the dice!
Gamage has rolled 2 4
Gamage Score is 44
Roll the dice by pressing any key. Good Luck!
Sixth Iteration
Player B/Gamage has rolled 5 5
Therefore, Gamage will get another chance to roll the dice as he got same value on both dices and that is not equal to ‘1’.



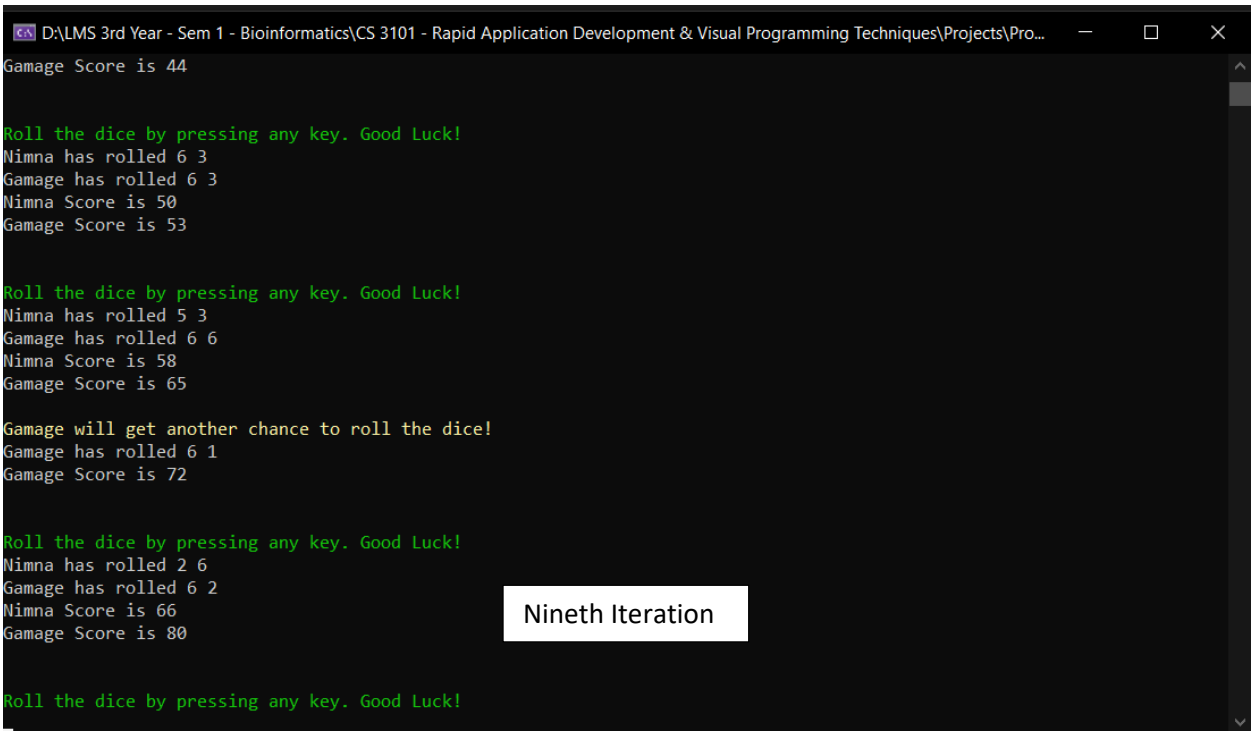
Seventh Iteration

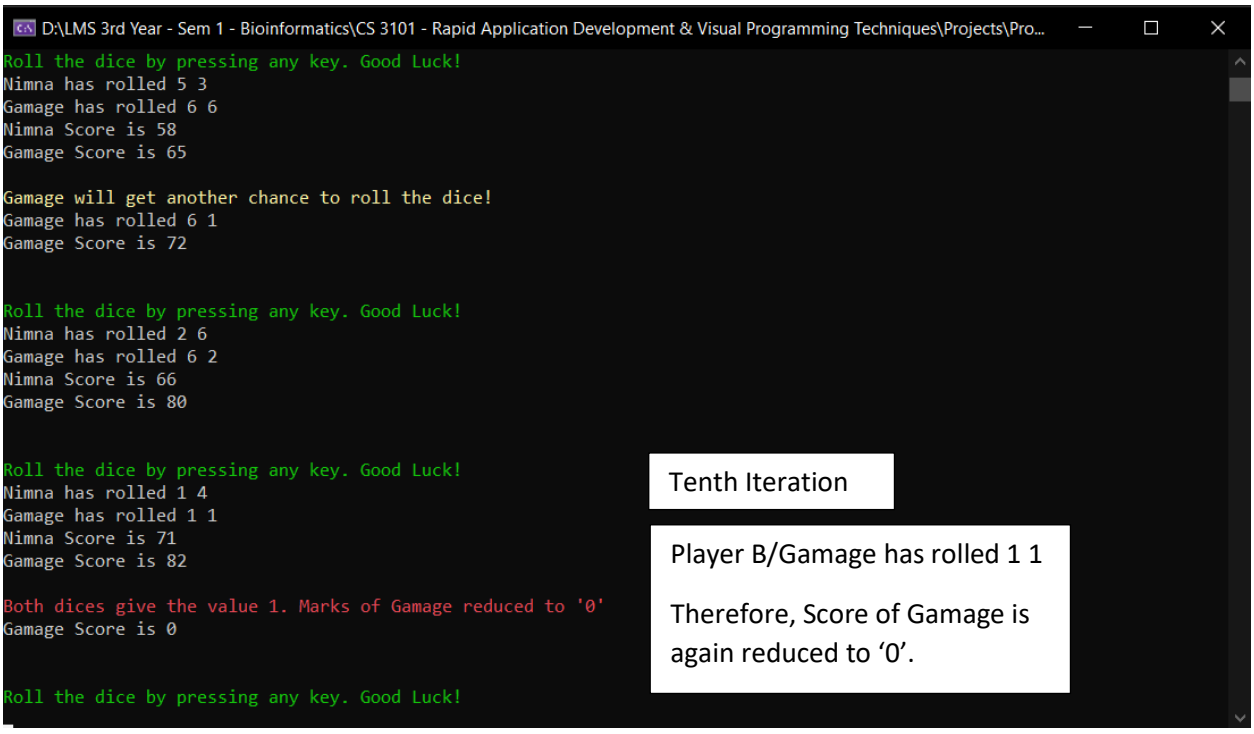


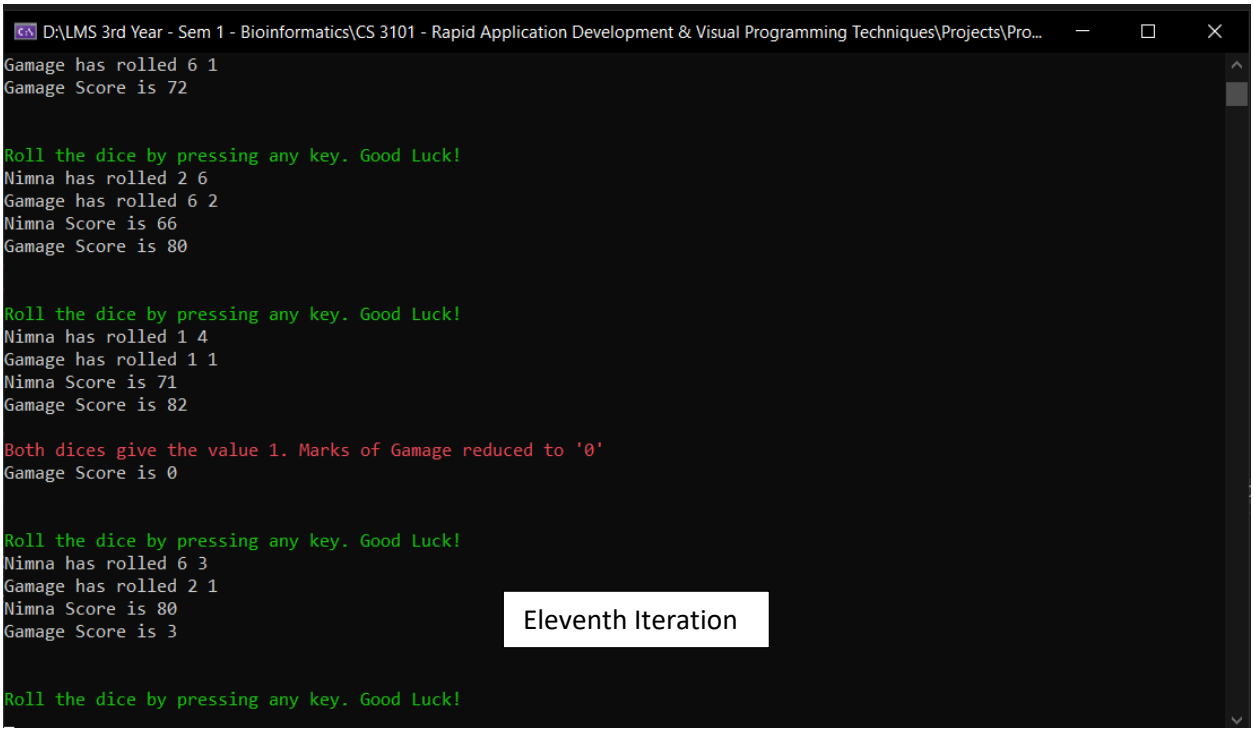
Eighth Iteration

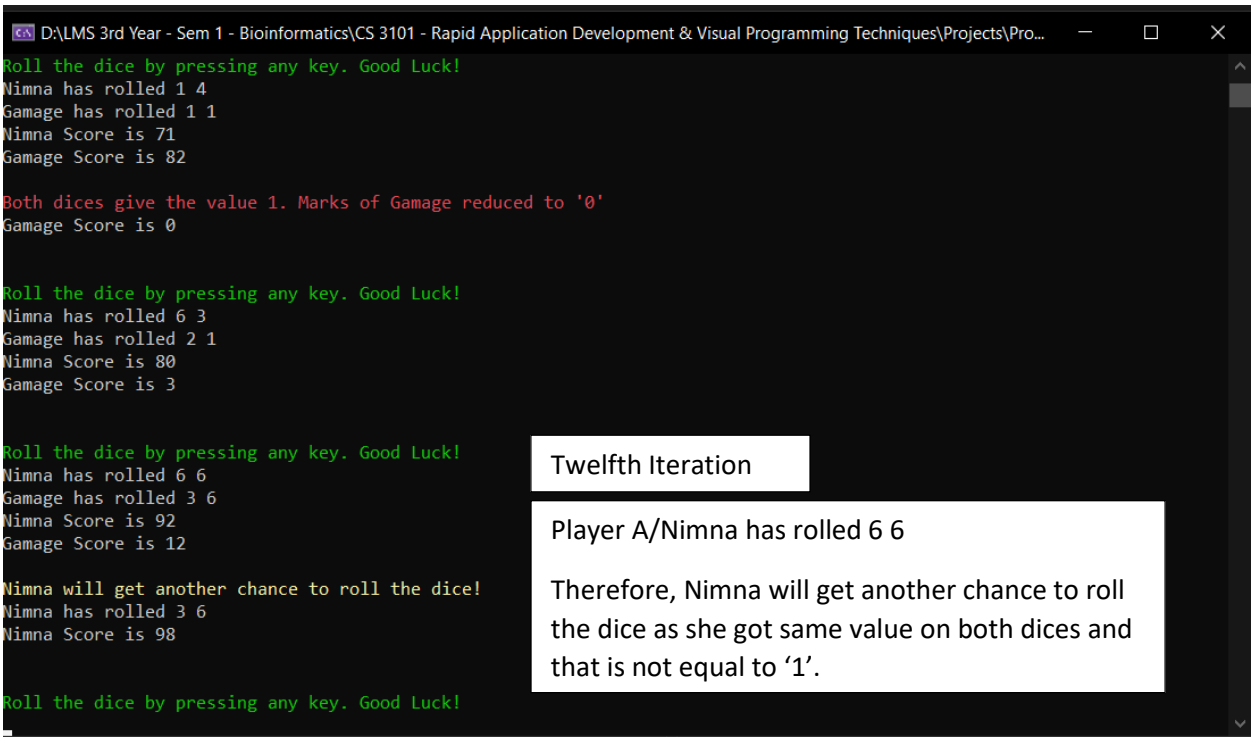
Player B/Gamage has rolled 6 6

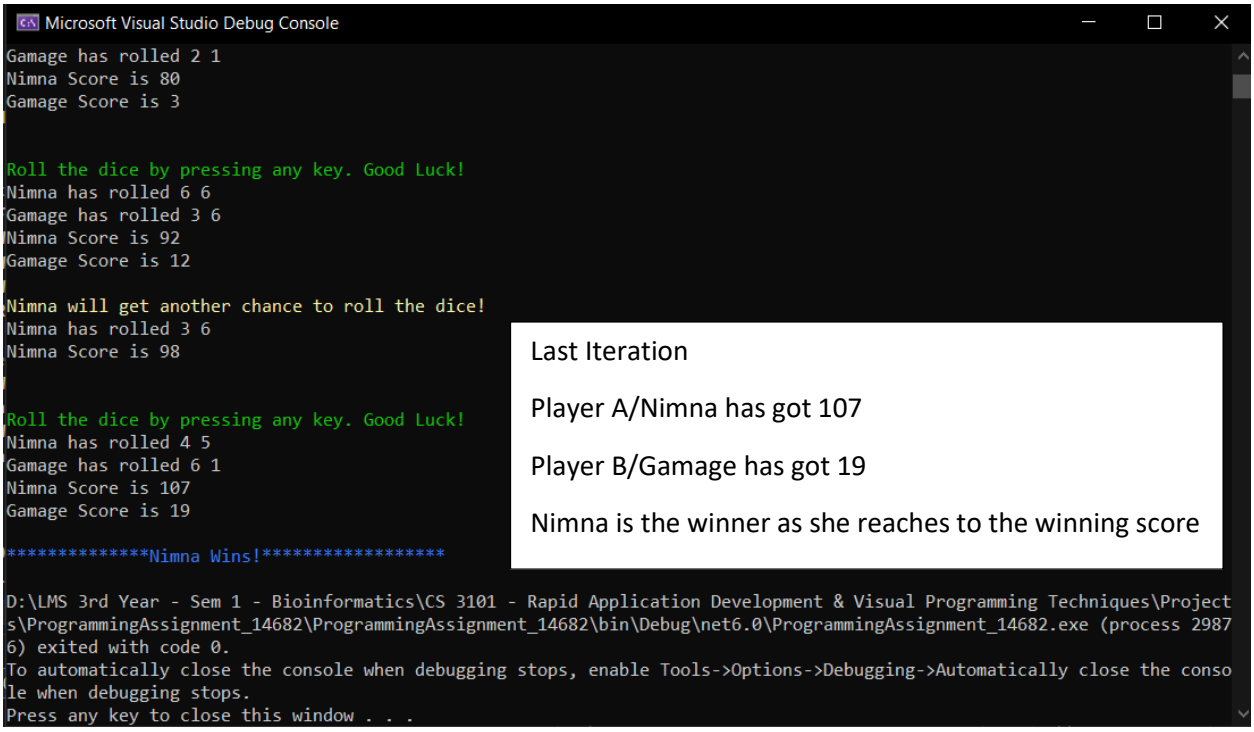
Therefore, Gamage will get another chance to roll the dice as he got same value on both dices and that is not equal to ‘1’.


D:\LMS 3rd Year - Sem 1 - Bioinformatics\CS 3101 - Rapid Application Development & Visual Programming Techniques\Projects\Pro...
Gamage Score is 44
Roll the dice by pressing any key. Good Luck!
Nimna has rolled 6 3
Gamage has rolled 6 3
Nimna Score is 50
Gamage Score is 53
Roll the dice by pressing any key. Good Luck!
Nimna has rolled 5 3
Gamage has rolled 6 6
Nimna Score is 58
Gamage Score is 65
Gamage will get another chance to roll the dice!
Gamage has rolled 6 1
Gamage Score is 72
Roll the dice by pressing any key. Good Luck!
Nimna has rolled 2 6
Gamage has rolled 6 2
Nimna Score is 66
Gamage Score is 80
Roll the dice by pressing any key. Good Luck!
Nineth Iteration


D:\LMS 3rd Year - Sem 1 - Bioinformatics\CS 3101 - Rapid Application Development & Visual Programming Techniques\Projects\Pro...
Roll the dice by pressing any key. Good Luck!
Nimna has rolled 5 3
Gamage has rolled 6 6
Nimna Score is 58
Gamage Score is 65
Gamage will get another chance to roll the dice!
Gamage has rolled 6 1
Gamage Score is 72
Roll the dice by pressing any key. Good Luck!
Nimna has rolled 2 6
Gamage has rolled 6 2
Nimna Score is 66
Gamage Score is 80
Roll the dice by pressing any key. Good Luck!
Nimna has rolled 1 4
Gamage has rolled 1 1
Nimna Score is 71
Gamage Score is 82
Both dices give the value 1. Marks of Gamage reduced to '0'
Gamage Score is 0
Roll the dice by pressing any key. Good Luck!
Tenth Iteration
Player B/Gamage has rolled 1 1
Therefore, Score of Gamage is again reduced to ‘0’.


D:\LMS 3rd Year - Sem 1 - Bioinformatics\CS 3101 - Rapid Application Development & Visual Programming Techniques\Projects\Pro...
Gamage has rolled 6 1
Gamage Score is 72
Roll the dice by pressing any key. Good Luck!
Nimna has rolled 2 6
Gamage has rolled 6 2
Nimna Score is 66
Gamage Score is 80
Roll the dice by pressing any key. Good Luck!
Nimna has rolled 1 4
Gamage has rolled 1 1
Nimna Score is 71
Gamage Score is 82
Both dices give the value 1. Marks of Gamage reduced to '0'
Gamage Score is 0
Roll the dice by pressing any key. Good Luck!
Nimna has rolled 6 3
Gamage has rolled 2 1
Nimna Score is 80
Gamage Score is 3
Roll the dice by pressing any key. Good Luck!
Eleventh Iteration


D:\LMS 3rd Year - Sem 1 - Bioinformatics\CS 3101 - Rapid Application Development & Visual Programming Techniques\Projects\Pro...
Roll the dice by pressing any key. Good Luck!
Nimna has rolled 1 4
Gamage has rolled 1 1
Nimna Score is 71
Gamage Score is 82
Both dices give the value 1. Marks of Gamage reduced to '0'
Gamage Score is 0
Roll the dice by pressing any key. Good Luck!
Nimna has rolled 6 3
Gamage has rolled 2 1
Nimna Score is 80
Gamage Score is 3
Roll the dice by pressing any key. Good Luck!
Nimna has rolled 6 6
Gamage has rolled 3 6
Nimna Score is 92
Gamage Score is 12
Nimna will get another chance to roll the dice!
Nimna has rolled 3 6
Nimna Score is 98
Roll the dice by pressing any key. Good Luck!
Twelfth Iteration
Player A/Nimna has rolled 6 6
Therefore, Nimna will get another chance to roll the dice as she got same value on both dices and that is not equal to ‘1’.



The game is terminated after 13 iterations. This game can be further modified in different ways by adding game levels and different logics.