

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра «Интеллектуальные информационные технологии»

Лабораторная работа №2

За четвертый семестр

По дисциплине «Модели решения задач в интеллектуальных системах»

**Тема: «Синхронизация процессов и потоков»**

Выполнили:

студенты 2 курса

группы ИИ-17(2)

А. И. Хмурец

К.В. Попов

Д.Е. Савчук

Проверил:

С.В. Анфилец

Брест 2021

Цель: Необходимо разработать программу, реализующую взаимодействие нескольких потоков (нитей) в зависимости от варианта задания.

Вариант: Игра в крестики-нолики. Играют две нити. Провести n партий, окончательный вид поля для каждой партии вывести в файл, результаты (счет, например, 10:15) вывести в окне. Взаимодействие нитей через синхронизацию. Ходы делаются не случайным образом, а в соответствии с правилами и стратегией на не-проигрыш.

### Ход работы:

#### Листинг:

```
public partial class MRZvIS_2 : Form
{
    public MRZvIS_2()
    {
        InitializeComponent();
    }

    private void buttonStart_Click(object sender, EventArgs e)
    {
        int firstPIWin = 0, secPIWin = 0;
        if(tbNumOfGames.Text == "")
        {
            return;
        }
        int num;
        if(!int.TryParse(tbNumOfGames.Text, out num))
        {
            tbNumOfGames.Text = "";
            return;
        }

        for(int count = 0; count < num; count++)
        {
            Player firstPlayer = new Player();
            Player secondPlayer = new Player();
            Table table = new Table();
            firstPlayer.setHand("x", true);
            secondPlayer.setHand("o", false);
            SimpleThread fPIForThread = new SimpleThread(ref firstPlayer, ref table, 5);
            SimpleThread sPIForThread = new SimpleThread(ref secondPlayer, ref table, 4);
            Thread fThread = new Thread(new ThreadStart(fPIForThread.WorkProcess));
            Thread sThread = new Thread(new ThreadStart(sPIForThread.WorkProcess));

            fThread.Start();
            sThread.Start();

            sThread.Join();
            fThread.Join();

            table.fillTheFile();
            if (table.getResults() == "x")
            {

```

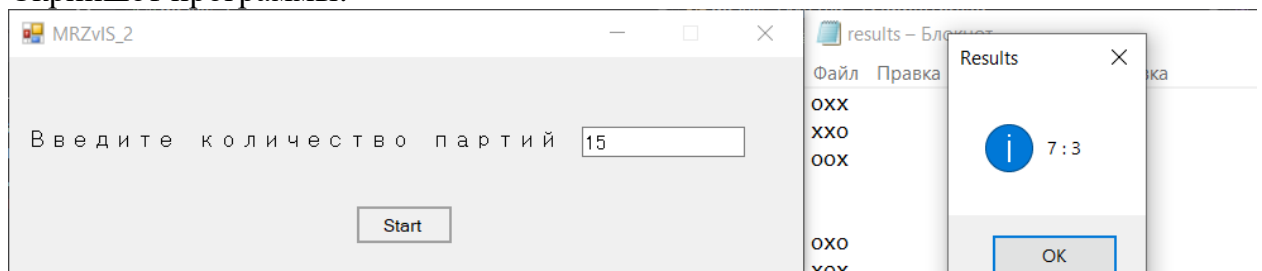
```

        firstPIWin++;
    }
    else if (table.getResults() == "o")
    {
        secPIWin++;
    }
}
string results = firstPIWin.ToString() + " : " + secPIWin;
MessageBox.Show(results, "Results", MessageBoxButtons.OK, MessageBoxIcon.Information);
}

public class SimpleThread
{
    Player player;
    Table table;
    int numOfIter;
    static Mutex cycleMutex = new Mutex();
    public SimpleThread(ref Player pl, ref Table tb, int count)
    {
        player = pl;
        table = tb;
        numOfIter = count;
    }
    public void WorkProcess()
    {
        for (int i = 0; i < numOfIter; i++)
        {
            cycleMutex.WaitOne();
            player.getCoordinates(ref table);
            Thread.Sleep(1);
            cycleMutex.ReleaseMutex();
        }
    }
}
}

```

Скриншот программы:



Вывод: Разработали программу, реализующую взаимодействие нескольких потоков (нитей) в зависимости от варианта задания.