

**ГБОУ ВПО Нижегородский государственный технический
университет им. Р. Е. Алексеева**
**Институт радиоэлектроники и информационных
технологий, кафедра "Вычислительные системы и
технологии"**

СОГЛАСОВАНО

Доцент каф. ВСТ

_____ Гай В. Е.

“ ____ ” _____

**ТЕХНОЛОГИИ РАСПРЕДЕЛЁННОЙ ОБРАБОТКИ
ДАННЫХ**

Отчет к лабораторной работе №3

**РАСПАРАЛЛЕЛИВАНИЕ АЛГОРИТМА С ПОМОЩЬЮ
БИБЛИОТЕКИ CSR**

Ив. подл.	Подп. и дата	Подп. и дата	Взам. инв.	Ив. дубл.	Подп. и дата

Студент гр. 13-В-1

_____ Корегин А. Д.

“ ____ ” _____

СОДЕРЖАНИЕ

1	Цель и порядок выполнения работы	3
2	Теоретические сведения	4
2.1	Библиотека Concurrent and Coordination Runtime	4
2.2	Создание проекта	5
2.3	Оценка времени выполнения	5
3	Выполнение лабораторной работы	6
3.1	Вариант задания	6
3.2	Листинг программы	6
3.3	Результат работы программы	10
4	Вывод	11

Инов. подл.	Подп. и дата		Взам. инв.		Инв. дубл.		Подп. и дата			
		Изм	Лист	докум.	Подп.	Дата	Распараллеливание алгоритма с помощью библиотеки CCR			
		Разраб.	Корегин А. Д				Технологии			
		Пров.	Гай В. Е.				распределённой			
							обработки данных			
		Н. контр.					Отчет к лабораторной			
		Утв.					работе №3			
								Лит.	Лист	Листов
									2	11

Копировал

Формат А4

1 ЦЕЛЬ И ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

Цель работы: получить представления о возможности библиотеки Concurrent and Coordination Runtime для организации параллельных вычислений.

Порядок выполнения работы:

- а) Разработка последовательного алгоритма, решающего одну из приведённых задач в соответствии с выданным вариантом задания;
- б) Разработка параллельного алгоритма, соответствующий варианту последовательного алгоритма;
- в) Выполнение сравнения времени выполнения последовательного и параллельного алгоритмов обработки данных при различных размерностях исходных данных.

Инов. подл.	Подп. и дата	Взам. инв.	Инов. дубл.	Подп. и дата	<div>Распараллеливание алгоритма с помощью библиотеки CCR</div>					Лист
										3
Изм	Лист	докум.	Подп.	Дата						

Копировал

Формат А4

2 ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

2.1 Библиотека Concurrent and Coordination Runtime

Библиотека Concurrent and Coordination Runtime (CCR) предназначена для организации обработки данных с помощью параллельно и асинхронно выполняющихся методов. Взаимодействие между такими методами организуется на основе сообщений. Рассылка сообщений основана на использовании портов. Основные понятия CCR:

- а) Сообщение – экземпляр любого типа данных;
- б) Порт – очередь сообщений типа FIFO (First-In-First-Out), сообщение остаётся в порте пока не будут извлечено из очереди порта получателем.

Определение порта:

```
Port<int> p = new Port<int>();
```

Отправка сообщения в порт:

```
p.Post(1);
```

- в) получатель – структура, которая выполняет обработку сообщений. Данная структура объединяет:

- один или несколько портов, в которые отправляются сообщения;
- метод (или методы), которые используются для обработки сообщений (такой метод называется задачей);
- логическое условие, определяющее ситуации, в которых активизируется тот или иной получатель.

Делегат, входящий в получатель, выполнится, когда в порт `intPort` придёт сообщение. Получатели сообщений бывают двух типов: временные и постоянные (в примере получатель – временный). Временный получатель, обработав сообщение (или несколько сообщений), удаляется из списка получателей сообщений данного порта.

Изм.	Лист	докум.	Подп.	Дата	Распараллеливание алгоритма с помощью библиотеки CCR	Лист
						4
Изм.	Лист	докум.	Подп.	Дата		

г) процессом запуска задач управляет диспетчер. После выполнения условий активации задачи (одним из условий активации может быть получение портом сообщения) диспетчер назначает задаче поток из пула потоков, в котором она будет выполняться. Описание диспетчера с двумя потоками в пуле:

```
Dispatcher d = new Dispatcher(2, "MyPool");
```

Описание очереди диспетчера, в которую задачи ставятся на выполнение:

```
DispatcherQueue dq = new DispatcherQueue("MyQueue d);
```

2.2 Создание проекта

Нужно выполнить следующие действия:

- а) Установить библиотеку CCR (CCR входит в состав Microsoft Robotics Developer Studio);
- б) Создать проект консольного приложения и добавьте к проекту библиотеку Microsoft.Ccr.Core.dll.

2.3 Оценка времени выполнения

Время выполнения вычислений будем определять с помощью класса

Stopwatch:

```
Stopwatch sWatch = new Stopwatch();
```

```
sWatch.Start();
```

<выполняемый код>

```
sWatch.Stop();
```

```
Console.WriteLine(sWatch.ElapsedMilliseconds.ToString());
```

Инов. подл.	Подп. и дата	Взам. инв.	Инов. дубл.	Подп. и дата	<div>Распараллеливание алгоритма с помощью библиотеки CCR</div>				Лист
									5
Изм	Лист	докум.	Подп.	Дата					

3 ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ

3.1 Вариант задания

Вариант 13:

- Разработать алгоритм сортировки массива чисел методом пузырька

3.2 Листинг программы

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Microsoft.Ccr.Core;

namespace CCR_Test
{
    class Data
    {
        public int row = 0;
    }

    class Program
    {
        public static int[,] array;
        public static int[,] arrayLinear;

        const int arraySize = 1000;

        static public void SortBubble(int row)
        {
```

Изм.	Лист	докум.	Подп.	Дата	Распараллеливание алгоритма с помощью библиотеки CCR	Лист
						6

```

int c = 0;
for (int i = 0; i < array.GetLength(1) - 1; i++)
{
    for (int j = 0; j < array.GetLength(1) - i - 1; j++)
    {
        if (array[row, j] > array[row, j + 1])
        {
            c = array[row, j + 1];
            array[row, j + 1] = array[row, j];
            array[row, j] = c;
        }
    }
}

```

```

static void Task(Data d, Port<int> resp)
{
    SortBubble(d.row);
    resp.Post(1);
}

```

```

static void ParallelBubbleSort()
{
    Dispatcher dispatcher = new Dispatcher(arraySize, "
        THREAD_POOL");
    DispatcherQueue dsipQueue = new DispatcherQueue("Dispatcher"
        , dispatcher);

    Port<int> port = new Port<int>();

    for (int i = 0; i < arraySize; ++i)
    {
        Data d = new Data();
        d.row = i;
        Arbiter.Activate(dsipQueue, new Microsoft.Ccr.Core.Task<
            Data, Port<int>>(d, port, Task));
    }

    return;
}

```

Инов. подл.	Подп. и дата	Инов. дубл.	Подп. и дата	Инов. инв.	Подп. и дата	<div>Распараллеливание алгоритма с помощью библиотеки CCR</div>		Лист
								7
Изм	Лист	докум.	Подп.	Дата				

Инв. подл.	Подп. и дата	Взам. инв.	Инв. дубл.	Подп. и дата

8


```

sw1.Start();
c = 0;
for (int k = 0; k < arraySize; ++k)
{
    for (int i = 0; i < arrayLinear.GetLength(1) - 1; i++)
    {
        for (int j = 0; j < arrayLinear.GetLength(1) - i - 1; j++)
        {
            if (arrayLinear[k, j] > arrayLinear[k, j + 1])
            {
                c = arrayLinear[k, j + 1];
                arrayLinear[k, j + 1] = arrayLinear[k, j];
                arrayLinear[k, j] = c;
            }
        }
    }
}
sw1.Stop();

```

```

Console.WriteLine("Planar_sorting_ended._Time:_{0}", sw1.
    Elapsed.Milliseconds);

```

```

s = new System.IO.StreamWriter("out.txt");
for (int i = 0; i < arraySize; ++i)
{
    for (int j = 0; j < arraySize; ++j)
    {
        s.Write("_{0}_", array[i, j]);
    }
    s.WriteLine();
}
s.Close();

```

```

Console.WriteLine("Writing_to_file_ended.");

```

```

Console.ReadLine();
return;

```

```

    }
}

```

Инов. подл.	Подп. и дата	Взам. инв.	Инов. дубл.	Подп. и дата	<div>Распараллеливание алгоритма с помощью библиотеки CSR</div>					Лист
										9
Изм	Лист	докум.	Подп.	Дата						

}

3.3 Результат работы программы

Скриншот работы программы представлен на Рис.

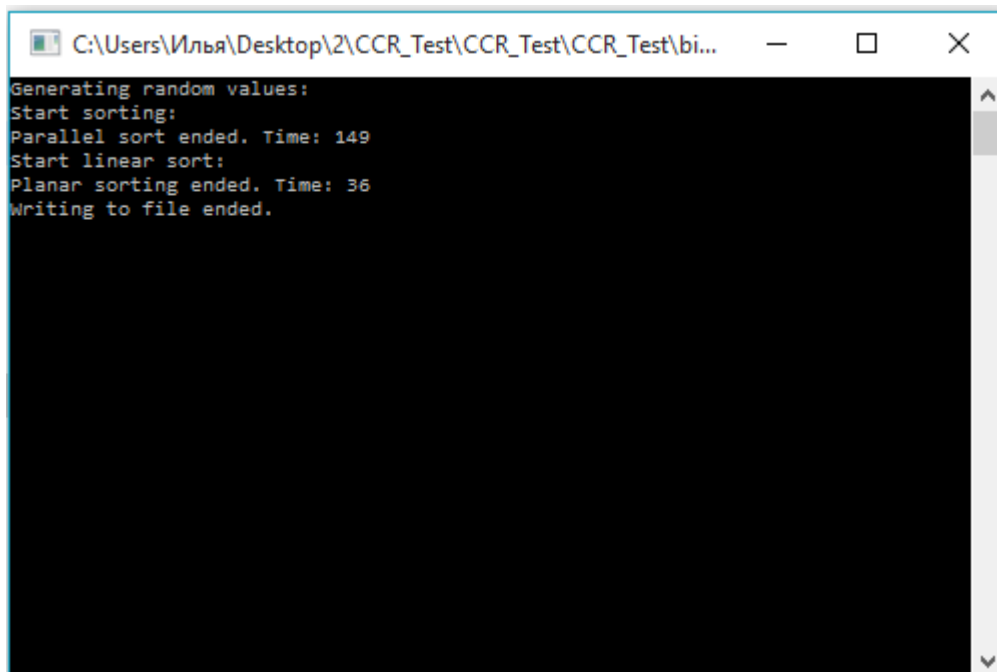


Рисунок 1

Инов. подл.	Подп. и дата	Взам. инв.	Инов. дубл.	Подп. и дата
Изм	Лист	докум.	Подп.	Дата

4 ВЫВОД

В результате выполнения лабораторной работы мы получили представление о возможности библиотеки Concurrent and Coordination Runtime для организации параллельных вычислений. Мы выяснили, что скорость работы параллельного алгоритма превосходит скорость работы последовательного алгоритма. Быстродействие параллельного алгоритма напрямую зависит от числа используемых ядер.

Инов. подл.	Подп. и дата	Взам. инв.	Инов. дубл.	Подп. и дата	<div>Распараллеливание алгоритма с помощью библиотеки CCR</div>					Лист
										11
Изм	Лист	докум.	Подп.	Дата						

Копировал

Формат А4