

14 МНОГОПОТОЧНЫЕ ПРИЛОЖЕНИЯ

Задание 1. Создать консольное приложение, в котором реализовано 3 потока. Первый выводит числа от 0 до 9, второй – от 10 до 19, третий – от 20 до 29. Вывод значений на экран происходит следующим образом: сначала своё число выводит первый поток, а затем второй и третий. Организовать возможность использования методов Start и Sleep, а также изменение свойства Priority, позволяющее изменить приоритет потока.

Листинг программы:

```
class Pogram
{
    static void mythread1()
    {
        for (int i = 0; i < 10; i++)
        {
            Console.WriteLine("Поток 1 выводит " + i);
        }
    }
    static void mythread2()
    {
        for (int i = 10; i < 20; i++)
        {
            Console.WriteLine("Поток 2 выводит " + i);
        }
    }
    static void mythread3()
    {
        for (int i = 20; i < 30; i++)
        {
            Console.WriteLine("Поток 3 выводит " + i);
        }
    }
}
```

					УП 2-40 01 01.31 ТП.2471.22.14			
Изм.	Лист	№ докум	Подп.	Дата				
Разраб.		Мушинский М.С			МНОГОПОТОЧНЫЕ ПРИЛОЖЕНИЯ			
Пров.		Толочко П.С.						
Н.контр.								
Утв.								
					Лит			Лист
								Листов
					Гродненский ГКТТид			

```

static void Main(string[] args)
{
    Thread thread1 = new Thread(mythread1);
    Thread thread2 = new Thread(mythread2);
    Thread thread3 = new Thread(mythread3);
    thread1.Priority = ThreadPriority.Highest;
    thread2.Priority = ThreadPriority.AboveNormal;
    thread3.Priority = ThreadPriority.Lowest;
    thread1.Start();
    thread2.Start();
    Thread.Sleep(1000);
    thread3.Start();
    Console.ReadLine();
}
}

```

Таблица 14.1 – Входные и выходные данные

Входные данные	Выходные данные
	0 – 9, 10 – 19, 20 – 29

Анализ результатов:

```

Поток 1 выводит 9
Поток 2 выводит 10
Поток 2 выводит 11
Поток 2 выводит 12
Поток 2 выводит 13
Поток 2 выводит 14
Поток 2 выводит 15
Поток 2 выводит 16
Поток 2 выводит 17
Поток 2 выводит 18
Поток 2 выводит 19
Поток 3 выводит 20
Поток 3 выводит 21
Поток 3 выводит 22

```

Рисунок 14.1 – Результат работы программы
Источник: собственная разработка

Задание 2. Создать консольное приложение, в котором несколько потоков будут выполнять один и тот же метод. (Количество потоков: 2, Метод: Сумма чисел от 1 до 10). Произвести расчет затраченного времени в миллисекундах на выполнение потока и вывести его на экран.

Листинг программы:

```

using System.Threading.Tasks;
using System.Diagnostics;
static void method()
{
    int sum = 0;
    for (int i = 1; i <= 10; i++)
    {
        sum += i;
        Console.WriteLine("task Прибавка к переменной sum значения = " + i);
    }
    Console.WriteLine($"Result task = {sum}");
}
Stopwatch sw = new Stopwatch();
Stopwatch sw2 = new Stopwatch();
Task task1 = new Task(method);
Task task2 = new Task(method);
sw.Start();
task1.Start();
task1.Wait();
sw.Stop();
sw2.Start();
task2.Start();
task2.Wait();
sw2.Stop();
TimeSpan ts = sw.Elapsed;
TimeSpan ts2 = sw2.Elapsed;
string elapsedTime = String.Format("{0:00}:{1:00}:{2:00}.{3:00}", ts.Hours,
ts.Minutes, ts.Seconds, ts.Milliseconds / 10);
Console.WriteLine("RunTime " + elapsedTime);
string elapsedTime2 = String.Format("{0:00}:{1:00}:{2:00}.{3:00}",
ts2.Hours, ts2.Minutes, ts2.Seconds, ts2.Milliseconds / 10);
Console.WriteLine("RunTime2 " + elapsedTime);
Console.ReadLine();

```

Таблица 14.2 – Входные и выходные данные

Входные данные	Выходные данные
	1 – 10, 55, 00:00:00.19

Анализ результатов:

```
task Прибавка к переменной sum значения = 1
task Прибавка к переменной sum значения = 2
task Прибавка к переменной sum значения = 3
task Прибавка к переменной sum значения = 4
task Прибавка к переменной sum значения = 5
task Прибавка к переменной sum значения = 6
task Прибавка к переменной sum значения = 7
task Прибавка к переменной sum значения = 8
task Прибавка к переменной sum значения = 9
task Прибавка к переменной sum значения = 10
Result task = 55
task Прибавка к переменной sum значения = 1
task Прибавка к переменной sum значения = 2
task Прибавка к переменной sum значения = 3
task Прибавка к переменной sum значения = 4
task Прибавка к переменной sum значения = 5
task Прибавка к переменной sum значения = 6
task Прибавка к переменной sum значения = 7
task Прибавка к переменной sum значения = 8
task Прибавка к переменной sum значения = 9
task Прибавка к переменной sum значения = 10
Result task = 55
RunTime 00:00:00.19
RunTime2 00:00:00.19
```

Рисунок 14.2 – Результат работы программы
Источник: собственная разработка