**14 Многопоточные приложения**

**Задание №4:** Дана последовательность натуральных чисел {a 0 …a n–1}. Создать многопоточное приложение для поиска суммы ∑a i, где a i – четные числа.

Листинг программы:

using System;

using System.Threading;

class Program

{

static readonly object lockObject = new object();

static int A, N;

static void Main()

{

Console.WriteLine("Введите значение A:");

tA = int.Parse(Console.ReadLine());

Console.WriteLine("Введите значение N:");

N = int.Parse(Console.ReadLine());

Thread sumThread1 = new Thread(CalculateSum);

Thread sumThread2 = new Thread(CalculateSum);

Thread productThread1 = new Thread(CalculateProduct);

Thread productThread2 = new Thread(CalculateProduct);

sumThread1.Start("Сумма поток 1");

sumThread2.Start("Сумма поток 2");

productThread1.Start("Произведение поток 1");

productThread2.Start("Произведение поток 2");

sumThread1.Join();

sumThread2.Join();

productThread1.Join();

productThread2.Join();

Console.WriteLine("Все потоки завершили работу.");

}

static void CalculateSum(object threadName)

{

int sum = 0;

for (int i = 1; i <= N; i++)

{

sum += A \* i;

Thread.Sleep(50); // Имитация работы

}

Console.WriteLine($"{threadName}: A\*1 + A\*2 + ... + A\*{N} = {sum}");

}

static void CalculateProduct(object threadName)

{

lock (lockObject) // Блокировка для синхронизации

{

int product = 1;

for (int i = 1; i <= N; i++)

{

product \*= A \* i;

Thread.Sleep(50); // Имитация работы

}

Console.WriteLine($"{threadName}: A\*1 \* A\*2 \* ... \* A\*{N} = {product}");

}

}

}

Таблица 14.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 10  2 5 8 3 12 7 6 9 4 11  3 | 10  12  10  32 |

Анализ результатов:

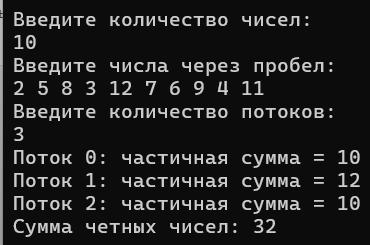


Рисунок 14.1 – Результат работы программы