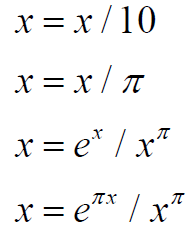
Лабораторна робота №3

Тема 1: ПАРАЛЕЛІЗМ ДАНИХ ТА ПАРАЛЕЛІЗМ ЗАДАЧ (Ч. 2)

Завдання:

1. Створити програму, що використовує для розпаралелювання метод Parallel.For(). Провести ряд обчислювальних експериментів із різним типом елементів масиву (int, double); різною кількістю елементів масиву; різною складністю обчислень:



Виміряти час, що витрачається паралельним способом обробки та послідовним. Результати експериментів оформити у вигляді таблиці. Зробити висновки.

1. Модифікувати попередню програму (створивши новий проект у рішенні) таким чином, щоб відбувався вихід з паралельного циклу за умови входження значення елемента у деякий окіл деякого числа (число та відхилення задаються константами).
2. Повторити приклади розпаралелювання за допомогою ForEach().
3. Модифікувати попередню програму (створивши новий проект у рішенні) таким чином, щоб тіло паралельного циклу задавалося лямбда-виразом.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Спосіб обробки  Складність обчислень | Послідовний | | Паралельний | |
| Час обробки (секунди) | | | |
| Integer | Double | Integer | Double |
|  | 1,4069356 | 1,453328 | 0,4103983 | 0,469245 |
|  | 0,5927225 | 0,5581479 | 0,1841632 | 0,2072709 |
|  | 1,5521812 | 1,5694029 | 1,2786856 | 0,3854418 |
|  | 4,3786736 | 1,6985488 | 1,3490834 | 0,5092929 |

Дослідження часу обробки для послідовного та паралельного виконання операцій з цілочисельними (int) та дійсними (double) числами показує, що паралельне виконання є значно швидшим у всіх випадках, особливо ефективним виявляється для складних математичних обчислень, таких як та . Це підтверджує переваги використання багатопоточності у багатоядерних системах, демонструючи її значний потенціал для покращення продуктивності в обчислювальних задачах.

**Висновки**: у ході лабораторної роботи було детальніше розглянуто термін паралелізм. Було досліджено створення паралельних задач, використовуючи методи Parallel.For() та ForEach(). Також було проведено експеримент на дослідження витрати часу послідовно оброблюваних даних та паралельно оброблюваних даних. У результаті дійдено висновків, що не завжди паралельний підхід обробки даних є ефективнішим за послідовний.