

Модуль 2. Паралельне програмування з використанням бібліотеки TPL та PLINQ

Лабораторна робота 2. Паралелізм задач (6 балів)

Мета роботи: вивчення основних методів класу **Task.Factory**.

1. Клас **Task.Factory**. Застосування методів очікування **Task.Wait()**.
2. Створення продовження задачі без параметрів.
3. Створення продовження задачі з параметрами.

Методичні вказівки.

Прочитайте лекцію 2 і проаналізуйте наведені в ній приклади. Знайдіть відповіді на наведені в кінці лекції запитання.

Усі проекти створіть в одному рішенні Lab2.

Програма **AsyncDownload** демонструє послідовне і паралельне завантаження сайтів. Дослідити і пояснити функції цієї програми.

```
//-----
//
// Copyright (c) Microsoft Corporation. All rights reserved.
//
// File: Program.cs
//
//-----

using System;
using System.Collections.Generic;
using System.Linq;
using System.Net;
using System.Threading.Tasks;

class Program
{
    static void Main()
    {
        // NOTE: Synchronous .Wait() calls added only for demo purposes

        // Single async request
        Download("http://www.microsoft.com").ContinueWith(CompletedDownloadData).Wait();

        // Single async request with timeout
        Download("http://www.microsoft.com").WithTimeout(new TimeSpan(0, 0, 0, 0, 1)).ContinueWith(CompletedDownloadData).Wait();

        // Serial async requests
        Task.Factory.TrackedSequence(
            () => Download("http://blogs.msdn.com/pfxteam"),
            () => Download("http://blogs.msdn.com/nativeconcurrency"),
            () => Download("http://exampleexampleexample.com"), // will fail
            () => Download("http://msdn.com/concurrency"),
            () => Download("http://bing.com")
        ).ContinueWith(SerialTasksCompleted).Wait();

        // Concurrent async requests
        Task.Factory.ContinueWhenAll(new []
        {
            Download("http://blogs.msdn.com/pfxteam"),
```

```

        Download("http://blogs.msdn.com/nativeconcurrency"),
        Download("http://exampleexampleexample.com"), // will fail
        Download("http://msdn.com/concurrency"),
        Download("http://bing.com")
    }, ConcurrentTasksCompleted).Wait();

    // Done
    Console.WriteLine();
    Console.WriteLine("Press <enter> to exit.");
    Console.ReadLine();
}

static Task<byte[]> Download(string url)
{
    return new WebClient().DownloadDataTask(url);
}

static void CompletedDownloadData(Task<byte[]> task)
{
    switch (task.Status)
    {
        case TaskStatus.RanToCompletion:
            Console.WriteLine("Request succeeded: {0}", task.Result.Length);
            break;
        case TaskStatus.Faulted:
            Console.WriteLine("Request failed: {0}", task.Exception.InnerException);
            break;
        case TaskStatus.Canceled:
            Console.WriteLine("Request was canceled");
            break;
    }
}

static void SerialTasksCompleted(Task<IList<Task>> tasks)
{
    int failures = tasks.Result.Where(t => t.Exception != null).Count();
    Console.WriteLine("Serial result: {0} successes and {1} failures",
tasks.Result.Count() - failures, failures);
}

static void ConcurrentTasksCompleted(Task<byte[]>[] tasks)
{
    int failures = tasks.Where(t => t.Exception != null).Count();
    Console.WriteLine("Concurrent result: {0} successes and {1} failures", tasks.Length
- failures, failures);
}
}

```

Варіанти індивідуальних завдань Група ЗПІ-81

Варіант	Завдання
1	1. Клас Task.Factory. Метод StartNew() Створити клас з двома методами. Перший метод створює одновимірний масив із 100 випадкових чисел і множить їх на число 2. Результат вивести на консоль. Другий метод створює масив з 30 елементів, знаходить їхню суму і середнє арифметичне. Результат вивести на консоль. Розробити паралельну версію програми (з використанням класу

	<p>Task.Factory). Для організації очікування завершення застосувати метод Task.WaitAll(). Звільнити ресурси після завершення (метод Dispose()).</p> <p>2. Створення продовження задачі Створення масивів та їх сортування. Написати 3 методи генерації масивів з 100 елементів. Створити три задачі для виклику цих методів. Створити продовження задачі для сортування цих масивів.</p> <p>3. Створення продовження задачі. Передача параметрів Замовлення піци. Створити Windows Forms Application для розрахунку замовлення піци. На формі представити <i>три сорти піци</i> ціну і кількість кожної. Розмістити кнопку Розрахувати. Створити три задачі з параметрами і результатом, що повертається (вартість замовлення одного сорту). В кожній задачі обчислюється одне своє замовлення. Загальний розрахунок зробити в методі продовження задачі.</p>
2	<p>1. Клас Task.Factory В класі Program створити 2 методи. Перший метод обчислює суму елементів масиву. Другий метод обчислює добуток елементів масиву. Результати вивести на консоль. Розробити послідовну і паралельну версії програми. Для паралельної версії в методі main створити 2 задачі для виклику методів (з використанням класу Task.Factory). Визначити час виконання послідовної і паралельної версії програми. Для організації очікування завершення застосувати метод Task.Wait(). Звільнити ресурси після завершення.</p> <p>2. Створення продовження задачі Створення масивів та робота з ними. Написати 3 методи генерації масивів з 100 елементів та обчислення суми елементів кожного з масивів. Створити три задачі для виклику цих методів. Створити продовження задачі для обчислення загальної суми елементів цих масивів.</p> <p>3. Створення продовження задачі. Передача параметрів Замовлення у МакДональд. Створити Windows Forms Application для розрахунку замовлення їжи. На формі представити <i>три види їжи</i>, ціну і кількість кожної. Розмістити кнопку Розрахувати. Створити три задачі з параметрами і результатом, що повертається (вартість замовлення одного виду їжи). В кожній задачі обчислюється одне своє замовлення. Загальний розрахунок зробити в методі продовження задачі.</p>

3	<p>1. Клас Task.Factory. Метод StartNew() Написати програму обчислення середньої температури за літо (за 3 місяці). Дані про температури зберігаються у трьох одновимірних масивах. Створити клас з одним методом обчислення середньої температури за місяць та полями для масивів та середніх значень. Написати послідовну і паралельну версію програми (з використанням класу Task.Factory). Визначити час виконання послідовної і паралельної версії програми. Звільнити ресурси після завершення.</p> <p>2. Створення продовження задачі Створення масивів та їх сортування. Написати 3 методи генерації масивів з 100 елементів. Створити три задачі для виклику цих методів. Створити продовження задачі для сортування цих масивів.</p> <p>3. Створення продовження задачі. Передача параметрів Замовлення піци. Створити Windows Forms Application для розрахунку замовлення піци. На формі представити <i>три сорти піци</i> ціну і кількість кожної. Розмістити кнопку Розрахувати. Створити три задачі з параметрами і результатом, що повертається (вартість замовлення одного сорту). В кожній задачі обчислюється одне своє замовлення. Загальний розрахунок зробити в методі продовження задачі.</p>
4	<p>1. Клас Task.Factory. Метод StartNew() Згенерувати масив з 25 випадкових чисел у діапазоні від 46 до 100 (оцінки за кредит). Створити 2 методи. Перший метод обчислює середній рейтинг групи. Другий метод підраховує кількість оцінок, які менші за 60 (ті, що не склали кредит). Розробити послідовну і паралельну версію програми і оцінити час роботи кожної версії. Для паралельної версії створити 2 задачі (з використанням класу Task.Factory). Для організації очікування завершення застосувати метод Task.Wait(). Звільнити ресурси після завершення.</p> <p>2. Створення продовження задачі Створення масивів та робота з ними. Написати 3 методи генерації масивів з 100 елементів та обчислення середнього арифметичного елементів кожного з масивів. Створити три задачі для виклику цих методів. Створити продовження задачі для обчислення середнього арифметичного елементів цих масивів.</p> <p>3. Створення продовження задачі. Передача параметрів Замовлення у морозива. Створити Windows Forms Application для розрахунку замовлення морозива. На формі представити <i>чотири</i></p>

	<p><i>види морозива, ціну і кількість кожного. Розмістити кнопку Розрахувати. Створити 4 задачі з параметрами і результатом, що повертається (вартість замовлення одного виду морозива). В кожній задачі обчислюється одне своє замовлення. Загальний розрахунок зробити в методі продовження задачі.</i></p>
5	<p>1. Клас Task.Factory. Метод StartNew() У двох одновимірних масивах зберігаються результати кредиту з двох дисциплін однієї групи. Визначити, з якої дисципліни успішність краща та кількість студентів, які не склали кредит з кожної дисципліни. Створити клас для зберігання потрібних полів та методів. Розробити послідовну і паралельну версію програми (з використанням класу Task.Factory). Оцінити час роботи кожної версії. Для організації очікування завершення застосувати метод Task.WaitAll (). Звільнити ресурси після завершення.</p> <p>2. Створення продовження задачі Створення масивів та робота з ними. Написати 3 методи генерації трьох масивів для зберігання даних про опади за три місяці (весна) та обчислення середньої кількості опадів за місяць. Створити три задачі для виклику цих методів. Створити продовження задачі для обчислення середньої кількості опадів за весну (за 3 місяці).</p> <p>3. Створення продовження задачі. Передача параметрів Замовлення кави. Створити Windows Forms Application для розрахунку замовлення кави. На формі представити 3 <i>види кави (експрессо, капуччіно, американо)</i>, ціну і кількість кожного виду. Розмістити кнопку Розрахувати. Створити 3 задачі з параметрами і результатом, що повертається (вартість замовлення одного виду кави). В кожній задачі обчислюється одне своє замовлення. Загальний розрахунок зробити в методі продовження задачі.</p>