



СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ



«ПАРАЛЕЛЬНІ ТА РОЗПОДІЛЕНІ ОБЧИСЛЕННЯ»

Шифр та назва спеціальності	124 – Системний аналіз	Факультет / Інститут	<u>Комп'ютерних наук і програмної інженерії</u>
Назва освітньо-наукової програми	Системний аналіз і управління	Кафедра	Системного аналізу та інформаційно-аналітичних технологій

ВИКЛАДАЧ



Прокопенков Володимир Пилипович, Volodymyr.Prokopenko@khi.edu.ua

Старший викладач кафедри Системного аналізу та інформаційно-аналітичних технологій НТУ «ХПІ». Досвід роботи – 30 років. Автор 45 наукових та навчально-методичних праць.

ЗАГАЛЬНА ІНФОРМАЦІЯ ПРО ДИСЦИПЛІНУ

Анотація	Дисципліна спрямована на освоєння методів та засобів паралельного програмування на основі систем автоматизації програмування Visual Studio.Net
Мета та цілі	Вивчення та освоєння студентами загальних проблем, наявних засобів і технології паралельного програмування в потоках для організації різних обчислень на персональних обчислювальних системах з багатоядерними процесорами
Формат	Лекції, лабораторні роботи, консультації. Підсумковий контроль - іспит
Результати навчання	Теоретичні та практичні навички розробки програмних застосувань мовою C# # в середовищі Visual Studio.Net В результаті вивчення дисципліни студент повинен вміти: Створювати й аналізувати паралельні алгоритми. Оцінювати ефективність паралельних алгоритмів. Створювати паралельні програми засобами бібліотек паралельних обчислень на рівні потоків.
Обсяг	Загальний обсяг дисципліни 120 год.: лекції – 32 год., лабораторні роботи – 32 год., самостійна робота – 56 год.
Пререквізити	«Програмування та алгоритмічні мови», «Архітектура обчислювальних систем», «Об'єктно-орієнтоване програмування», «Алгоритми та структури даних», «Технології програмування»
Вимоги викладача	Студент зобов'язаний відвідувати всі заняття згідно розкладу, не спізнюватися. Дотримуватися етики поведінки. Для проходження дисципліни необхідно мати: бажання, можливості, терпіння, сумління та персональний комп'ютер. Працювати з навчальною та додатковою літературою, з літературою на електронних носіях і в Інтернеті. З метою оволодіння необхідною якістю освіти з дисципліни потрібно відвідуваність і регулярна підготовленість до занять.

СТРУКТУРА ДИСЦИПЛІНИ

Лекція 1	Делегати та події як основа паралелізму. Події. Механізм подій.	Лабораторна робота 1	Використання делегатів для асинхронного виконання методів класу.	Самостійна робота	Створення Windows додатку з використанням об'єкта класу BackgroundWorker.
Лекція 2	Використання механізму подій для контролю завершення асинхронної операції.	Лабораторна робота 2	Події. Механізм подій.		
Лекція 3	Виконання програм в ОС Windows. Процеси. Потоки. Програмування в потоках. Багатозадачність. Паралельність. Пріоритети	Лабораторна робота 3	Використання механізму подій для контролю завершення асинхронної операції.		
Лекція 4	Організація потоку. Робота з класом Thread простору System.Threading.	Лабораторна робота 4	Організація потоку.		
Лекція 5	Створення об'єкта потоку з робочим методом з параметром.	Лабораторна робота 5	Робота з класом Thread.		
Лекція 6	Умови реалізації паралельних обчислень. Оцінка ефективності паралельних обчислень. Огляд засобів паралельної розробки програм.	Лабораторна робота 6	Створення об'єкта потоку з робочим методом з параметром.		
Лекція 7	Організація багатопотокової програми.	Лабораторна робота 7	Організація багатопотокової програми.		
Лекція 8	Властивості потоку Thread.	Лабораторна робота 8	Властивості потоку Thread.		Асинхронні завдання і можливості бібліотеки TPL.
Лекція 9	Управління потоками.	Лабораторна робота 9	Управління потоками.		
Лекція 10	Локальні та глобальні дані потоку.	Лабораторна робота 10	Локальні та глобальні дані потоку.		
Лекція 11	Пул потоків.	Лабораторна робота 11	Використання пулу потоків для паралельної обробки даних.		Засоби налагодження програм у середовищі Microsoft Visual Studio з використанням .NET Framework
Лекція 12	Синхронізація потоків.	Лабораторна робота 12	Синхронізація потоків.		
Лекція 13	Засоби і методи синхронізації потоків.	Лабораторна робота 13	Засоби блокування критичних секцій.		
Лекція 14	Клінчі. Боротьба з клінчами.	Лабораторна робота 14	Клінчі. Боротьба з клінчами.		
Лекція 15	Паралелізм завдань. Створення програми з розпаралелюванням завдань.	Лабораторна робота 15	Створення програми з розпаралелюванням завдань.		
Лекція 16	Паралелізм даних. Створення програми з розпаралелюванням даних.	Лабораторна робота 16	Створення програми з розпаралелюванням даних.		

ЛІТЕРАТУРА ТА НАВЧАЛЬНІ МАТЕРІАЛИ

1. Рейли Д. Создание приложений Microsoft ASP.Net/ Пер.с англ.- М.: Изд.-торговый дом Русская редакция, 2002.-480с. : ил.
2. Петцольд Ч. Программирование для Microsoft Windows на C#. В 2-х томах. Том 1./Пер. с англ. — М.: Издательско-торговый дом Русская Редакция, 2002.- 576 с.: ил.
3. Петцольд Ч. Программирование для Microsoft Windows на C#. В 2-х томах. Том 2./Пер. с англ. — М.: Издательско-торговый дом Русская Редакция, 2002.— 624 с.: ил.
4. Лабор В. В. Си Шарп: Создание приложений для Windows/ В. В. Лабор.— Мн.: Харвест, 2003. -384 с.
5. Рихтер Дж. Программирование на платформе Microsoft .NET Framework /Пер. с англ. — 2-е изд., испр. — М.: Издательско-торговый дом Русская Редакция, 2003- — 512 стр.: ил.
6. Разработка Windows-приложений на Microsoft Visual Basic .NET и Microsoft Visual C# -NET. Учебный курс MCAD/MCSD/Пер. с англ. — М.: Издательско-торговый дом Русская Редакция, 2003. — 512 стр.: ил.
7. Троелсен Э. C# и платформа NET. Библиотека программиста. — Спб.: Питер, 2003.-800с.:ил.
8. Анализ требований и создание архитектуры решений на основе Microsoft .NET. Учебный курс MCSD/Пер. с англ. -- М.: Издательско-торговый дом Русская Редакция, 2004.— 416 стр.: ил.
9. Шилдг, Герберт. Полный справочник по C#. : Пер. с англ. — М. : Издательский дом "Вильямс", 2004. — 752 с. : ил.
10. Бишоп Дж. C# в кратком изложении/ Дж. Бишоп, Н.Хорспул; Пер. с англ.=М.: Бином, Лаборатория знаний, 2005.-472с., ил.
11. Методичні вказівки до лабораторних робіт "Основи програмування на мові C#" для студентів напрямів 6.040302 "Інформатика", 6.040303 "Системний аналіз" з курсу "Технологія програмування" / Упоряд. Ю. Н.Кожин, О. Н.Малих, В. Ф. Прокопенков – Харків: НТУ "ХПІ", 2011. – 122 с.
12. Методичні вказівки до лабораторних робіт "Робота з двійковими файлами в мові C#" з дисципліни "Технологія програмування" для студентів напрямів 6.040302 "Інформатика", 6.040303 "Системний аналіз" /упоряд. Ю. Н. Кожин, О. Н. Малих, В. Ф. Прокопенков – Х.: НТУ "ХПІ", 2016.– 32 с.
13. Методичні вказівки до лабораторних робіт "Обробка виняткових ситуацій у мові C#" з дисципліни "Технологія програмування" для студентів напрямів 6.040302 "Інформатика", 6.040303 "Системний аналіз" /упоряд. Ю. Н. Кожин, О. Н. Малих, В. Ф. Прокопенков – Х.: НТУ "ХПІ", 2013.– 36 с.

1. Науково-технічна бібліотека НТУ «ХПІ» [Електронний ресурс]. Режим доступу: <http://library.kpi.kharkov.ua/>
2. Русский MSDN [Електронний ресурс].-Режим доступу: <https://msdn.microsoft.com/ru-ru>
3. Англомовний MSDN [Електронний ресурс].- Режим доступу: <https://msdn.microsoft.com/en-us/>
4. Язык программирования C# 5.0 и платформа .NET 4.5, Эндрю Троелсен, 6-е издание, 1312 стр., ISBN 978-5-8459-1814-7, «ВИЛЬЯМС», март 2013
5. C# 5.0. Справочник. Полное описание языка, Джозеф Албахари, Бен Албахари, 5-е издание, 1054 стр., ISBN 978-5-8459-1819-2, «ВИЛЬЯМС», 2013

ПЕРЕЛІК ЗАПИТАНЬ ДЛЯ ПІДГОТОВКИ ДО ЗАЛІКУ

Архітектура обчислювальної системи. Класи архітектур обчислювальних систем. Алгоритми. Швидкість, продуктивність і ефективність обчислювальних систем. Цілі паралельних обчислень. Умови реалізації паралельних обчислень. Паралелізм. Архітектурний паралелізм. Алгоритмічний паралелізм. Паралельні обчислення як основа високопродуктивних обчислень. Засоби розробки паралельних програм. Оцінка ефективності паралельних обчислень. Прискорення паралельного алгоритму. Ефективність паралельного алгоритму. Максимальний теоретичний вииграш в продуктивності від паралельного алгоритму - Закон Амдала. Максимально допустиме прискорення від паралельної програми - Закон Густафсона-Барсиса. Огляд засобів паралельної розробки програм. Середовище Microsoft Visual Studio з використанням .NET Framework. Бібліотеки для паралельного програмування .NET Framework. Виконання програм в ОС Windows. Процеси. Потіки. Багатозадачність. Паралельність. Пріоритети. Делегати та події як основа паралелізму. Функціональний тип і клас делегат. Анонімні функції і лямбда вирази. Базовий клас MulticastDelegate. Синхронне виконання делегата. Асинхронне виконання делегата. Інтерфейс IAsyncResult. Контроль завершення асинхронної операції, властивості IsCompleted, AsyncWaitHandle. Отримання результату виконання асинхронної операції. Події. Механізм подій. Визначення внутрішнього і зовнішнього обробників подій. Аналіз здійснення події і викид події одержувачу. Визначення обробника події в одержувача події. Включення події на прослуховування. Використання механізму подій для контролю завершення асинхронної операції. Визначення події. Використання події для контролю завершення операції. Програмування в потоках. Організація потоку. Структура потоку. Стан потоку. Перемикання контексту. Основні етапи роботи з потоками. Робота з класом Thread простору System.Threading. Створення потоку. Робочий метод потоку. Створення об'єкта потоку з робочим методом без параметрів. Делегат ThreadStart. Створення потоку з робочим методом без параметрів у вигляді анонімної функції. Створення потоку з робочим методом без параметрів у вигляді лямбда виразу. Запуск потоку на виконання, метод Thread.Start(). Створення об'єкта потоку з робочим методом з параметром. Делегат Parameterized-ThreadStart. Створення потоку з робочим методом з параметром у вигляді анонімної функції. Створення потоку з робочим методом з параметром у вигляді лямбда виразу. Запуск потоку на виконання, метод Thread.Start(object). Передача параметрів у потік за допомогою параметра. Організація багатопотокової програми. Первинний потік програми і вторинні потоки. Запуск вторинних потоків в основному потоці програми. Способи очікування моменту завершення вторинного потоку в основному потоці: метод Thread.Join(), організація свого завершення потоку, використання класу AutoResetEvent. Властивості потоку Thread. Контекст потоку Thread.Context. Отримання посилання на потік Thread.CurrentThread. Стан активності потоку Thread.IsAlive. Власне ім'я потоку Thread.Name. Пріоритет потоку Thread.Priority. Стан потоку Thread.State. Отримання статистики про потік. Управління потоками. Переклад потоку в стан очікування методом Thread.Sleep(). Рекомендаційний пріоритет потоку для обробки в ОС. Основні та фонові потоки, властивість Thread.IsBackground. Призупинення роботи потоку Thread.Suspend(). Переривання роботи потоку Thread.Interrupt(). Припинення роботи потоку Thread.Abort(). Відновлення роботи потоку Thread.Resume(). Локальні та глобальні дані потоку. Локальні дані потоку за місцем оголошення. Оголошення локальної змінної потоку поза області визначення потоку, атрибут ThreadStatic. Використання об'єктів ThreadLocal<T> для оголошення локальних даних потоку поза області визначення потоку. Використання локальних слотів потоку. Отримання слоту потоком, метод Thread.GetNamedDataSlot(). Збереження даних у слоті потоку методом Thread.SetData(). Витяг даних з слота потоку методом Thread.GetData(). Пул потоків. Клас ThreadPool. Розміщення методу для виконання в асинхронному режимі в пул потоків. Засоби і методи синхронізації потоків. Загальні ресурси потоків. Поняття гонки умов (даних). Критична секція. Проблема синхронізації потоків при використанні загальних ресурсів. Механізм подій як засіб блокування критичних секцій. Синхронізація потоків як спосіб завдання правильного порядку виконання потоків. Пасивне блокування. Активне блокування. Гібридні засоби блокування. Клінчі. Боротьба з клінчами. Поняття клінчу. Модель клінчу. Боротьба з клінчем на двох потоках. Боротьба з клінчем на багатьох потоках. Асинхронні завдання і можливості бібліотеки TPL. Паралелізм завдань. Створення програми з розпаралелюванням завдань. Паралелізм даних. Створення програми з розпаралелюванням даних. Створення Windows додатку з використанням об'єкта класу BackgroundWorker. Засоби налагодження програм у середовищі Microsoft Visual Studio з використанням .NET Framework.

ПЕРЕЛІК ОБЛАДНАННЯ

Лабораторний практикум укомплектовано наступним устаткуванням:

сучасні ПЕОМ з багатоядерними процесорами

з операційною системою Windows 10,

середовищем розробки програмних застосувань Visual Studio.Net 2017

СИСТЕМА ОЦІНЮВАННЯ

Розподіл балів для оцінювання успішності аспіранта		Оцінка ECTS	Оцінка за національною шкалою	Нарахування балів
	90-100	A	відмінно	
	82-89	B	добре	
	74-81	C		
	64-73	D	задовільно	
	60-63	E		
	35-59	FX	незадовільно з можливістю повторного складання	
	0-34	F	незадовільно з обов'язковим повторним вивченням дисципліни	

Підсумкову залікову оцінку студент може отримати як рейтингову – інтегровану оцінку з накопичених оцінок під час семестрового навчання.

Якщо студент не отримав залік за результатами поточного контролю, то він виставляється за результатами виконання ним залікової контрольної роботи.

НОРМИ АКАДЕМІЧНОЇ ЕТИКИ

Студент повинен дотримуватися «Кодексу етики академічних взаємовідносин та доброчесності НТУ «ХПІ»»: виявляти дисциплінованість, вихованість, доброзичливість, чесність, відповідальність. Конфліктні ситуації повинні відкрито обговорюватися в навчальних групах з викладачем, а при нерозв'язності конфлікту доводиться до завідуючого кафедри, декану.

Силабус за змістом повністю відповідає робочій програмі навчальної дисципліни