

Міністерство освіти і науки України
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Кафедра математичного моделювання та штучного інтелекту (№ 304)

ЗАТВЕРДЖУЮ

Керівник проектної групи



(підпис)

Д. І. Чумаченко

(ініціали та прізвище)

« 20 » _____ 08 _____ 2023 р.

**РОБОЧА ПРОГРАМА ОBOB'ЯЗКОВОЇ
НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

Паралельні та розподілені обчислення

(шифр і назва навчальної дисципліни)

Галузь знань: 12 "Інформаційні технології "
(шифр і найменування галузі знань)

Спеціальність: 122 "Комп'ютерні науки"
(код та найменування напрямку підготовки)

Освітня програма: "Інтелектуальні системи та технології"
(найменування освітньої програми)

Форма навчання: денна

Рівень вищої освіти: перший (бакалаврський)

Харків 2023 рік

Робоча програма навчальної дисципліни

«Паралельні та розподілені обчислення»

(назва навчальної дисципліни)

для студентів за спеціальністю 122 «Комп'ютерні науки»

освітніми програмами "Інтелектуальні системи та технології"

«27» серпня 2023 р. – 12 с.

РОЗРОБНИК ПРОГРАМИ:

Скоб Ю. О., проф. кафедри 304, д. т. н., доцент

(прізвище та ініціали, посада, наукова ступень та вчене звання)

(підпис)

Робочу програму розглянуто на засіданні випускової кафедри Математичного
моделювання та штучного інтелекту (№ 304)

(назва кафедри)

Протокол № 1 від «30» серпня 2023 р.

Завідувач

кафедри № 304

д.т.н., професор

(науковий ступінь і вчене звання)

(підпис)

А. Г. Чухрай

(ініціали та прізвище)

Опис навчальної дисципліни

Найменування показників	Галузь знань, напрям підготовки, рівень вищої освіти	Характеристика навчальної дисципліни
		Денна форма навчання
Кількість кредитів – 4,5	<p>Галузь знань 12 "Інформаційні технології" (шифр та найменування)</p> <p>Спеціальність 122 "Комп'ютерні науки" (код та найменування)</p> <p>Освітня програма "Інтелектуальні системи та технології" (найменування)</p> <p>Рівень вищої освіти: <u>перший (бакалаврський)</u></p>	Обов'язкова
Модулів – 2		Рік підготовки:
Змістових модулів – 4		2023/ 2024
Індивідуальне науково-дослідне завдання: <u>розрахункова робота «Паралельний розрахунок на багатоядерному комп'ютері»</u>		Семестр
Загальна кількість годин – 135 денна – 72/135		7-й
		Лекції
Тижневих годин для денної форми навчання: аудиторних – 4,5 самостійної роботи студента - 3,9		32 год.
		Практичні
		–
		Лабораторні
	40 год.	
	Самостійна робота	
	63 год.	
Індивідуальна робота		
–		
Вид контролю		
модульний контроль, іспит		

Примітка

Співвідношення кількості годин аудиторних занять до самостійної і індивідуальної роботи становить:

для денної форми навчання – 72/63.

2. Мета та завдання навчальної дисципліни

Мета навчальної дисципліни: здобуття теоретичних знань та практичних навичок програмування складних, розподілених та навантажених систем за допомогою сучасних технологій та паралельних обчислень; формування у студентів уявлення про сучасні методи проектування і створення паралельного і розподіленого багатоярусного програмного забезпечення, що функціонує в рамках .NET Framework як на локальному багатопроецесорному комп'ютері або комп'ютері з багатоядерним процесором, так і в локальній мережі комп'ютерів, в Інтернеті.

Завдання навчальної дисципліни: отримання навичок розробки алгоритмів паралельних обчислень, програмування паралельних методів розв'язань рівнянь та перетворення

арифметичних виразів за допомогою сучасних технологій, виконання віддалених викликів процедур та застосування методів; вивчення принципів функціонування паралельних та розподілених застосувань, моделей і парадигм паралельних та розподілених обчислень, формування навичок створення сучасних об'єктно-орієнтованих комп'ютерних систем з паралельною архітектурою.

Згідно з вимогами освітньо-професійної програми студенти повинні досягти таких **загальних компетентностей:**

- Здатність застосовувати знання у практичних ситуаціях (ЗК 2)
- Знання та розуміння предметної області та професійної діяльності (ЗК 3).
- Здатність вчитися і оволодівати сучасними знаннями (ЗК 6).
- Базові знання науково-методичних основ і стандартів в області інформаційних технологій, уміння застосовувати їх під час розробки та інтеграції систем, продуктів і сервісів інформаційних технологій (ЗК17).
- Навички використання інформаційних і комунікаційних технологій (ЗК21).

Згідно з вимогами освітньо-професійної програми студенти повинні досягти таких **фахових компетентностей:**

- Ґрунтовна підготовка в області програмування, володіння алгоритмічним мисленням, методами програмної інженерії для реалізації програмного забезпечення з урахуванням вимог до його якості, надійності, виробничих характеристик (ФК 4).

Програмні результати навчання.

Відповідно до освітньої програми студент повинен досягти наступних програмних результатів:

- Знання теоретичних і практичних основ методології та технології моделювання у процесі дослідження, проектування та експлуатації інформаційних систем, продуктів, сервісів інформаційних технологій, інших об'єктів професійної діяльності; здатність реалізовувати алгоритми моделювання для дослідження характеристик і стану складних об'єктів (ПРН 18).

Міждисциплінарні зв'язки: для вивчення дисципліни «Паралельні та розподілені обчислення» необхідно володіти запасом знань таких дисциплін, як «Програмування та алгоритмічні мови», «Методи обчислень», «Операційні системи», «Організація та обробка електронної інформації», «Організація баз даних», «Ознайомча практика», а в подальшому знання з дисципліни «Об'єктно-орієнтоване програмування» стануть основою для вивчення таких дисциплін, як «Обчислювальна геометрія та комп'ютерна графіка», «Платформи корпоративних інформаційних систем» та ін.

3. Програма навчальної дисципліни

Модуль 1.

Змістовий модуль 1. Аналіз паралельного алгоритму, дослідження його ефективності, виявлення проблематики та розробка прототипу паралельної програми

Тема 1. Предмет вивчення і задачі дисципліни.

Предмет вивчення і задачі дисципліни «Паралельні та розподілені обчислення». Основні історичні етапи розвитку і становлення сучасної моделі проектування і програмування, перспективи подальшого розвитку. Паралельне програмування в. NET. Переваги та недоліки паралельного програмування. Облік накладних витрат. Координація даних. Масштабування додатків. Коли слід виконувати паралельно. Вирішуючи, коли виконувати послідовно.

Тема 2. Загальні паралельні алгоритми.

Сортування, пошук і кешування. Паралельний алгоритм сортування Quicksort. Перехід по паралельному дереву. Пошук у паралельному дереві. Використання паралельного кешу пам'яті. Використання паралельної карти і скорочень. Використання паралельного відображення. Використання паралельного скорочення. Використання паралельного «відображення-скорочення». Умоглядна обробка (Speculative Processing). Селекція.

Модульний контроль

Змістовий модуль 2. Реалізація паралельних алгоритмів на платформі передачі повідомлень

Тема 3. Програмування задач.

Ознайомлення з класом Task. Створення та запуск завдань. Створення простих завдань. Налаштування стану завдання. Як отримати результат. Завдання параметрів створення завдання. Ідентифікація завдань. Відміна одного завдання. Моніторинг скасування за допомогою опитування. Моніторинг скасування за допомогою делегата. Моніторинг скасування за допомогою дескриптора очікування. Відміна декількох завдань. Створення композитних маркерів скасування. Визначення, чи була задача скасована. В очікуванні часу продовження виконання. Використання дескриптора очікування маркера скасування. Використання класичного очікування. Використання спінового очікування. Очікування завдань. В очікуванні єдиного завдання. Очікування декількох завдань. В очікуванні однією з багатьох завдань. Ліниве виконання завдань. Загальні проблеми та їх причини. Тупик залежностей завдань. Оцінений локальних змінних. Надмірний спінінг.

Тема 4. Координація задач.

Додаткові можливості в роботі з завданнями. Використання продовжень завдань. Створення простих продовжень. Створення продовжень «один- до-багатьох». Створення вибіркового продовжень. Створення продовжень «багато-до-одного» і «будьякий-до-одного». Відміна продовжень. Очікування продовжень. Обробка виключень. Створення дочірніх завдань. Використання синхронізації для координації завдання. Бар'єри. Подія CountdownEvent. Клас ManualResetEventSlim. Клас AutoResetEvent. Клас SemaphoreSlim. Використання паралельного шаблону «виробник /споживач». Створення шаблону. Об'єднання множинних колекції. Використання користувацького планувальника завдання. Створення користувацького планувальника.

Модульний контроль

Змістовий модуль 3. Теоретичні та прикладні аспекти проектування розподілених застосувань

Тема 5. Спільне використання даних.

Проблеми з даними. Гонка за даними. Створення деякого порядку. Послідовне виконання. Виконання без зміни даних. Виконання в ізоляції. Синхронізуючи виконання. Визначення критичних ділянок. Визначення примітивів синхронізації. Використовуючи синхронізацію розумно. Використання основних примітивів синхронізації. Блокування та моніторинг. Використання заблокованих операцій. Використання спінової блокування. Використання дескрипторів очікування і класу Mutex. Конфігурування міжпроцесної синхронізації. Використання декларативної синхронізації. Використання блокувань доступу «читання-запис». Неочікувана мінливість. Множинні блокування. Порядок отримання блокування. Осиротілі блокування.

Тема 6. Паралельні цикли.

Паралельні цикли проти послідовних циклів. Клас Parallel. Виклик дій (Actions). Використання паралельних циклів. Створення основного паралельного циклу for. Створення основного паралельного циклу ForEach. Установка опцій налаштування паралельного циклу. Переривання

і зупинка паралельних циклів. Обробка виключень паралельних циклів. Отримання результатів циклів. Відміна паралельних циклів. Поточкова локальна пам'ять в паралельних циклах. Виконання паралельних циклів з залежностями. Вибір стратегії розбиття. Використовуючи chunking - стратегії («розбиття на шматки»). Використання впорядкованої стратегії розбиття за замовчуванням. Створення користувальницької стратегії розбиття. Написання контекстного роздільник. Написання впорядковує контекстного роздільник.

Модульний контроль

Змістовий модуль 4. Програмний інструментарій проведення експериментів на Грід-платформах

Тема 7. Грід-обчислення.

Концепція грід. Порівняння грід-систем і звичайних суперкомп'ютерів. Типи грід-систем. Історія розвитку грід-систем. Основні функціональні підсистеми глобального грід. Підсистема управління завантаженням. Підсистема управління даними. Підсистема інформаційного обслуговування і моніторингу грід. Підсистема безпеки та контролю прав доступу. Підсистема протоколювання. Підсистема обліку.

Тема 8. Тестування та налагодження паралельних застосувань.

Оцінка паралельної продуктивності. Використання хороших стратегій кодування. Використовуйте синхронізацію економно. Не уникайте синхронізації. Рівномірно розподіляйте роботу. Уникайте паралелізації малих робочих навантажень. Оцініть різні ступені паралелізму. Виконання простих порівнянь продуктивності.

Модульний контроль

4. Структура навчальної дисципліни

Назви змістових модулів і тем	Кількість годин					
	денна форма					
	усього	у тому числі				
л		п	лаб	інд	с.р.	
1	2	3	4	5	6	7
Модуль 1.						
Змістовий модуль 1. Аналіз паралельного алгоритму, дослідження його ефективності, виявлення проблематики та розробка прототипу паралельної програми						
Тема 1. Предмет вивчення і задачі дисципліни	8	2				6
Тема 2. Загальні паралельні алгоритми	9	3				6
Модульний контроль.	1	1				
Разом за змістовим модулем 1	18	6				12
Змістовий модуль 2. Реалізація паралельних алгоритмів на платформі передачі повідомлень						
Тема 3. Програмування задач	18	4		8		6
Тема 4. Координація задач	20	4		10		6
Модульний контроль	2			2		
Разом за змістовим модулем 2	40	8		20		12
Змістовий модуль 3. Теоретичні та прикладні аспекти проектування розподілених застосувань						
Тема 5. Спільне використання даних	19	4		9		6
Тема 6. Паралельні цикли	19	4		9		6
Модульний контроль.	2			2		
Разом за змістовим модулем 3	40	8		20		12

Змістовий модуль 4. Програмний інструментарій проведення експериментів на Грід-платформах						
Тема 7. Грід-обчислення	10	4				6
Тема 8. Тестування та налагодження паралельних застосувань	10	4				6
Модульний контроль	1	1				
Разом за змістовим модулем 4	21	9				12
Модуль 2.						
РР. Паралельний розрахунок на багатоядерному комп'ютері	15					15
Модульний контроль	1	1				
Усього годин (модуль 2)	16	1				15
Контрольний захід – залік						
Усього годин навчальної дисципліни	135	32		40		63

5. Теми семінарських занять

№ з/п	Назва теми	Кількість годин
1		
2		
...		

6. Теми практичних занять

№ з/п	Назва теми	Кількість годин
1.	Програмування задач	8
2.	Спільне використання даних	10
3.	Модульний контроль	2
4.	Координація задач	9
5.	Паралельні цикли	9
6.	Модульний контроль	2
	Разом	40

7. Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
1		
2		
...		

8. Самостійна робота

№ з/п	Назва теми	Кількість годин
1.	Предмет вивчення і задачі дисципліни	6
2.	Загальні паралельні алгоритми	6
3.	Програмування задач	6
4.	Координація задач	6
5.	Спільне використання даних	6
6.	Паралельні цикли	6

7.	Грід-обчислення	6
8.	Тестування та налагодження паралельних застосувань	6
9.	РР. Паралельний розрахунок на багатоядерному комп'ютері	15
	Разом	63

9. Індивідуальні завдання

1. Виконання розрахункової роботи на тему «Паралельний розрахунок на багатоядерному комп'ютері».

Спроекувати схему паралельного розрахунку поставленої проблеми. Розробити об'єктно-орієнтовані класи на боці серверу і клієнта на базі ієрархії класів, що домішують функціональність за допомогою підтримки низки інтерфейсів. Проект програми розробити на основі консольного шаблону мовою C# в середовищі Microsoft Visual Studio.Net. Проектування класів і інтерфейсів виконати засобами інструментів «Діаграми класів». До пояснювальної записки долучити постанову задачі, тексти програми, копію вікна з результатами розрахунку.

Обсяг роботи – 15 сторінок.

10. Методи навчання

1. Пояснювально-ілюстративний (інформаційно-рецептивний) метод та метод проблемного виконання (лекційні заняття).
2. Репродуктивний (лабораторні роботи).
3. Частково-пошуковий (евристичний) та дослідницький (самостійна робота та виконання розрахункової та розрахунково-графічної робіт).
4. Дисципліна «Паралельні та розподілені обчислення» передбачає лекційні (в т.ч. з використанням мультимедійного обладнання) і лабораторні заняття під керівництвом викладача та самостійну роботу студента за підручниками і матеріалами, опублікованими кафедрою (методичні посібники і мережеві ресурси), що забезпечує закріплення теоретичних знань, сприяє набуттю практичних навичок і розвитку самостійного наукового мислення. Передбачено регулярні індивідуальні консультації.

11. Методи контролю

Визначення рівня засвоєння студентом навчального матеріалу дисципліни здійснюється шляхом проведення поточних (захист лаб. робіт, поточні контрольні з теоретичного матеріалу) і підсумкових контролів (захист змістовного модуля, залік).

У завдання поточного контролю входить систематична перевірка розуміння й засвоєння студентом програмного матеріалу, виконання практичних і лабораторних робіт, уміння самостійно проробляти тексти складання конспектів, написання звітів, здатності усно або письмово представляти певний матеріал.

Перед підсумковим контролем ставиться завдання перевірки глибини засвоєння студентом програмного матеріалу дисципліни, логіки й взаємозв'язки між її окремими розділами, здатності творчо використати придбані знання, уміння сформулювати своє відношення до проблеми, що впливає зі змісту дисципліни.

12. Розподіл балів, які отримують студенти (іспит)

12.1. Розподіл балів, які отримують студенти (кількісні критерії оцінювання)

Складові навчальної роботи	Бали за одне завдання	Кількість завдань	Сумарна кількість балів
Модуль 1			
Виконання і захист лабораторних робіт (змістовний модуль 1)	3...5	4	12...20
Модульний контроль	3...5	1	3...5
Виконання і захист лабораторних робіт(змістовний модуль 2)	3...5	4	12...20
Модульний контроль	3...5	1	3...5
Виконання і захист лабораторних робіт(змістовний модуль 3)	3...5	3	9...15
Модульний контроль	3...5	1	3...5
Виконання і захист лабораторних робіт(змістовний модуль 4)	3...5	3	9...15
Модульний контроль	3...5	1	3...5
Модуль 2			
Виконання і захист РР	3...5	1	3...5
Модульний контроль	3...5	1	3...5
Усього за семестр			60...100

Семестровий контроль (іспит) проводиться у разі відмови студента від балів поточного тестування й за наявності допуску до іспиту. Під час складання семестрового іспиту студент має можливість отримати максимум 100 балів.

Білет для іспиту складається з одного теоретичного і двох практичних питань. В першому питанні студент повинен продемонструвати теоретичні знання. У другому питанні – показати навички складання і виконання програми з використанням об'єктів задач. У третьому пункті – продемонструвати знання зі створення і застосування об'єктів паралельного доступу до спільних даних.

Складові білету	Складові оцінки	Бали за одне питання	Сумарна кількість балів
Пункт 1.	дано визначення наведеного поняття	5	20
	наведено приклади	5	
	наведено фрагменти тексту	10	
Пункт 2.	складено алгоритм розрахунку (блок-схема)	10	30
	створено проект консольного застосування з використанням задач	10	
	отримано результати в	10	

	КОНСОЛЬНОМУ ВІКНІ		
--	-------------------	--	--

Пункт 3.	складено алгоритм розрахунку (блок-схема)	10	50
	створено проект застосунку з паралельним доступом до спільних даних	10	
	отримано і проаналізовано результати у консольному вікні	30	
Ітогова оцінка за іспит			100

12.2. Якісні критерії оцінювання

Необхідний обсяг знань для одержання позитивної оцінки:

студент повинен знати:

- принципи програмування задач з паралельним обчисленням;
- як створити проект консольного застосунку;
- як створити проект застосунку з графічним інтерфейсом;
- як використовувати вбудований у середовище інструмент відлагодження застосунку;
- способи декларації, визначення і утворення об'єктів класів паралельних обчислень;
- як утворювати алгоритм розрахунку у вигляді блок-схеми;
- як використовувати принципи захисту спільних даних від некоректного доступу;
- як застосовувати паралельні алгоритми;
- як застосовувати паралельні цикли.

Необхідний обсяг вмінь для одержання позитивної оцінки:

повинен вміти:

- виконувати поставлене завдання у вигляді консольної або графічної аплікації з паралельним доступом до спільних даних;
- закладувати об'єкт задачі різними засобами;
- повернути результати розрахунку з об'єкта задачі;
- використовувати можливості інтегрованого середовища програмування для відлагодження паралельних застосунків;
- алгоритмізувати поставлене завдання у вигляді алгоритму розрахунку (блок-схеми);
- використовувати стандартні простори імен для доступу до стандартних бібліотек паралельних класів;
- застосовувати механізми захисту спільних даних під час одночасного доступу до них з різних об'єктів задач;
- застосовувати паралельні алгоритми для розв'язання прикладних задач.

12.3 Критерії оцінювання роботи студента протягом семестру.

Приклад.

Задовільно (60-74). Показати мінімум теоретичних знань та практичних умінь. Виконати всі лабораторні роботи за індивідуальним варіантом із застосуванням консольної аплікації. Захистити всі теми індивідуальних завдань та модулів. Знати як утворити алгоритм розрахунку у вигляді блок-схеми. Виконати розрахункову роботу у вигляді консольного застосунку.

Добре (75-89). Показати знання основних теоретичних питань та практичних умінь. Виконати всі лабораторні роботи за індивідуальним варіантом із застосуванням аплікації з графічним інтерфейсом і використанням паралельних класів. Захистити всі теми індивідуальних завдань та модулів на оцінку «добре». Знати як утворити алгоритм розрахунку у вигляді блок-схеми і створювати каркас класу за допомогою діаграми класів. Виконати

розрахункову роботу у вигляді консольного застосунку але з використанням класів і інтерфейсів, і захистити отримані результати розрахунку.

Відмінно (90-100). Показати тверде і досконале знання всіх теоретичних питань та практичних умінь. Виконати всі лабораторні роботи за індивідуальним варіантом із застосуванням аплікації з графічним інтерфейсом і використанням паралельних класів. Захистити всі теми індивідуальних завдань та модулів на оцінку «відмінно». Знати як утворити алгоритм розрахунку у вигляді блок-схеми і створювати каркас класу за допомогою діаграми класів. Вчасно виконати розрахункову роботу у вигляді застосунку з графічним інтерфейсом, з використанням паралельних класів і інтерфейсів, і захистити отримані результати розрахунку з оцінкою «відмінно».

Шкала оцінювання: бальна і традиційна

Сума балів	Оцінка за традиційною шкалою	
	Іспит, диференційований залік	Залік
90 – 100	Відмінно	Зараховано
75 – 89	Добре	
60 – 74	Задовільно	
0 – 59	Незадовільно	Не зараховано

13. Методичне забезпечення

1. Скоб Ю.О. Сучасні технології програмування / Ю.О. Скоб. – Навч. посібник до лаб. практикуму. – Харків: Нац. аерокосм. ун-т «Харк. авіац. ін-т», 2006. – 100 с.
2. Теорія програмування / Ю.О. Скоб, О.В. Патокіна, В.О. Халтурін. – Навч. посібник до лаб. практикуму. – Харків: Нац. аерокосм. ун-т «Харк. авіац. ін-т», 2005. – 73 с.
3. Об'єктно-орієнтоване програмування мовою С# / Ю.О. Скоб, М.Л. Угрюмов, К.П. Коробчинський. – Навч. посібник до лаб. робіт. – Харків: Нац. аерокосм. ун-т «Харк. авіац. ін-т», 2011. – 108 с. (електроний вигляд, каф. 304)
4. Програмування елементів управління ActiveX засобами бібліотеки активних шаблонів ATL / Ю.О. Скоб, М.Л. Угрюмов, К.П. Коробчинський. – Навч. посібник до кур. проекту. – Харків: Нац. аерокосм. ун-т «Харк. авіац. ін-т», 2006. – 108 с.
5. Основи програмування Windows мовою С# / Ю.О. Скоб, М.Л. Угрюмов, К.П. Коробчинський. – Навч. посібник до лаб. робіт. – Харків: Нац. аерокосм. ун-т «Харк. авіац. ін-т», 2010. – 108 с.

14. Рекомендована література

Базова література

1. Основы объектно-ориентированного программирования на языке С# / Ю.А. Скоб, М.Л. Угрюмов, К.П. Коробчинский. – Конспект лекций. – Харьков: Нац. аэрокосм. ун-т «Харк. авіац. ін-т», 2011. – 196 с.
2. Решение инженерных задач экологии средствами прикладных пакетов Excel, MathCad, Delphi, Visual C++ / В.Н. Кобрин, Ю.А. Скоб, М.Л. Угрюмов. – Учеб. пособие по лаб. практикуму. – Харьков: Нац. аэрокосм. ун-т «Харьк. авіац. ін-т», 2007. – 109 с.
3. Информатика / А.В. Карташов, Ю.А. Скоб, В.А. Халтурин, И.А. Трофимова, Ю.К. Чернишов, Л.И. Черноштан, О.В. Яровая. – Учеб. пособие по лаб. практикуму. – Харьков: ХАИ, 2005. – 177 с.
4. Информатика / А.Ю. Соколов, М.Л. Угрюмов, В.А. Халтурин, Ю.К. Чернышев. – Конспект лекций. – Харьков: ХАИ им. Н. Е. Жуковского, 2003. – 191 с.

5. Adam Freeman. Pro .NET 4 Parallel Programming in C#: Apress, 2011. – 329 p.
6. Просиз Дж. Программирование для Microsoft .NET: Пер. с англ. – М: Русская редакция, 2003. – 704 с.
7. Либерти Д. Программирование на C#: Пер. с англ. – СПб: Символ-Плюс, 2003. – 688 с.
8. Буч Г. Объектно-ориентированный анализ и проектирование с примерами приложений на C++: Пер с англ.- М.: Издательство Бином, СПб.: Невский диалект, 1998.- 560 с.
9. Microsoft Corporation. Разработка Windows-приложений на Microsoft Visual Basic.NET и Microsoft Visual C#.NET. Учебный курс MCAD/MCSD: Пер. с англ. – М: Русская Редакция, 2003. – 512 с.
10. Скоб Ю.О. Сучасні технології програмування: Навч. посібник до лаб. практикуму. – Харків: Нац. аерокосм. ун-т «Харк. авіац. ін-т», 2011 . – 100 с. (абонемент бібліотеки гол.корп.)

Допоміжна література

1. Рофаэл Э., Шохауд Я. СОМ и СОМ+. Полное руководство: Пер. с англ. – К.: ВЕК+; К.: НТИ; М.: Энтроп, 2000. – 560 с. Либерти Д. Программирование на C#: Пер. с англ. – СПб: Символ-Плюс, 2003. – 688 с.
2. Microsoft Corporation. Разработка Windows-приложений на Microsoft Visual Basic.NET и Microsoft Visual C#.NET. Учебный курс MCAD/MCSD: Пер. с англ. – М: Русская Редакция, 2003. – 512 с.
3. Черосвитов А. Visual C++ 6 и MFC. Курс MCSD для профессионалов. – СПб.: Питер, 2000. – 544 с.

15. Інформаційні ресурси

1. сайт кафедри №304 - <http://k304.khai.edu>
2. <https://sites.google.com/site/parcomp2012/lectures> - Лекції з паралельних обчислень.
3. <http://ics.opu.ua/rus/chairs/cs/literature1> - Учбові матеріали з паралельних обчислень.
4. http://ru.wikibooks.org/wiki/Объектно-ориентированное_программирование - Объектно-ориентированное программирование
5. <http://programming.in.ua/>- Програмування українською.