

**Міністерство освіти і науки України
Національний технічний університет
«Дніпровська політехніка»**

Кафедра програмного забезпечення комп'ютерних систем

«ЗАТВЕРДЖЕНО»

завідувач кафедри
Алексєєв М.О.

_____ 
« ____ » _____ 2024 року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

**«Моделі та методи створення програмних систем паралельної та розподіленої
обробки даних»**

Галузь знань	12 Інформаційні технології
Спеціальність	121 «Інженерія програмного забезпечення»
Рівень вищої освіти	третій (освітньо-науковий)
Статус.....	фахова
Загальний обсяг	6 кредити ЄКТС (180 годин)
Форма підсумкового контролю	диференційований залік, іспит
Термін викладання	1,2 -й семестр (1, 2, 3, 4 чв.)
Мова викладання	українська

Викладач: д.т.н., проф. каф. ПЗКС Геннадій ШВАЧИЧ

Пролонговано: на 20__/20__ н.р. _____ (_____) «__»__ 20__р.
(підпис, ПІБ, дата)

на 20__/20__ н.р. _____ (_____) «__»__ 20__р.
(підпис, ПІБ, дата)

Дніпро
НТУ «ДП»
2024

Робоча програма навчальної дисципліни «Моделі та методи створення програмних систем паралельної та розподіленої обробки даних» для докторів філософії спеціальності 121 «Інженерія програмного забезпечення» / Нац. техн. ун-т. «Дніпровська політехніка», Дніпро: НТУ «ДП», 2024. – 20 с.

Розробник: д.т.н., проф. каф. ПЗКС Геннадій ШВАЧИЧ

Робоча програма регламентує:

- мету дисципліни;
- дисциплінарні результати навчання, сформовані на основі трансформації очікуваних результатів навчання освітньої програми;
- базові дисципліни;
- обсяг і розподіл за формами організації освітнього процесу та видами навчальних занять;
- програму дисципліни (тематичний план за видами навчальних занять);
- алгоритм оцінювання рівня досягнення дисциплінарних результатів навчання (шкали, засоби, процедури та критерії оцінювання);
- інструменти, обладнання та програмне забезпечення;
- рекомендовані джерела інформації.

Робоча програма призначена для реалізації компетентнісного підходу під час планування освітнього процесу, викладання дисципліни, підготовки аспірантів до контрольних заходів, контролю провадження освітньої діяльності, внутрішнього та зовнішнього контролю забезпечення якості вищої освіти, акредитації освітніх програм у межах спеціальності.

ЗМІСТ

1 МЕТА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ	4
2 ОЧІКУВАНІ ДИСЦИПЛІНАРНІ РЕЗУЛЬТАТИ НАВЧАННЯ.....	5
3 БАЗОВІ ДИСЦИПЛІНИ	5
4 ОБСЯГ І РОЗПОДІЛ ЗА ФОРМАМИ ОРГАНІЗАЦІЇ ОСВІТНЬОГО ПРОЦЕСУ ТА ВИДАМИ НАВЧАЛЬНИХ ЗАНЯТЬ	5
5 ПРОГРАМА ДИСЦИПЛІНИ ЗА ВИДАМИ НАВЧАЛЬНИХ ЗАНЯТЬ.....	5
6 ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ	9
6.1 Шкали	9
6.2 Засоби та процедури.....	9
6.3 Критерії.....	11
7 ІНСТРУМЕНТИ, ОБЛАДНАННЯ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ	14
8 РЕКОМЕНДОВАНІ ДЖЕРЕЛА ІНФОРМАЦІЇ	12

1 МЕТА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

В освітньо-науковій програмі Національного технічного університету «Дніпровська політехніка» спеціальності 121 «Інженерія програмного забезпечення» здійснено розподіл програмних результатів навчання (ПРН) за організаційними формами освітнього процесу. Зокрема, до обов'язкової дисципліни «Моделі та методи створення програмних систем паралельної та розподіленої обробки даних» віднесено такі результати навчання:

РН03	Пропонувати нові ефективні методи і моделі розроблення, впровадження, супроводу та забезпечення якості програмного забезпечення та управління відповідними процесами на всіх етапах життєвого циклу.
РН07	Розробляти та досліджувати концептуальні, математичні і комп'ютерні моделі процесів і систем для отримання нових знань та/або створення інноваційних продуктів у інженерії програмного забезпечення та дотичних міждисциплінарних напрямках.
РН08	Глибоко розуміти загальні принципи та методи інженерії програмного забезпечення, а також методологію наукових досліджень, застосовувати їх у власних дослідженнях та у викладацькій практиці.
РН14	Розробляти, досліджувати та вдосконалювати програмні системи на основі моделей та засобів інтелектуальної обробки даних розподілених і масштабованих систем з урахуванням вимог до надійності їх структурно-алгоритмічного забезпечення.

Мета дисципліни – розвинути систему знань, умінь і навичок у здобувачів вищої освіти в області розвитку систем паралельної та розподіленої обробки даних, поглибити знання про моделі, методи та засоби паралельного програмування, що сприятиме формуванню здатності застосовувати знання з паралельного програмування на практиці, критично переосмислювати наявні інформаційні технології, відстежувати тенденції їх розвитку, розвивати й реалізовувати нові конкурентоздатні ідеї в галузі інформаційних технологій.

Завдання курсу:

- опанування теоретико-понятійної бази курсу;
- ознайомлення здобувачів з теоретичними основами математичних методів створення програмних систем паралельної та розподіленої обробки даних;
- засвоєння здобувачами технології розробки паралельних програм;
- засвоєння здобувачами понять та визначень теорії методів моделювання та оптимізація програмних систем;
- засвоєння здобувачами критеріїв ефективності паралельних алгоритмів;
- засвоєння здобувачами понять про методи моделювання, оцінки та оптимізації надійності технічних систем;
- отримання досвіду розробки нових та існуючих архітектурних рішень для розробки мультипроцесорних програмних систем;
- формування у здобувачів цілісної системи теоретичних знань з курсу «Моделі та методи створення програмних систем паралельної та розподіленої обробки даних».

2 ОЧІКУВАНІ ДИСЦИПЛІНАРНІ РЕЗУЛЬТАТИ НАВЧАННЯ

Шифр РН	Дисциплінарні результати навчання (ДРН)	
	шифр ДРН	зміст
РН03	РН03-Ф1	Пропонувати нові ефективні методи і моделі розроблення, впровадження, супроводу та забезпечення якості програмного забезпечення та управління відповідними процесами на всіх етапах життєвого циклу.
РН07	РН07-Ф1	Розробляти та досліджувати концептуальні, математичні і комп'ютерні моделі процесів і систем для отримання нових знань та/або створення інноваційних продуктів у інженерії програмного забезпечення та дотичних міждисциплінарних напрямках.
РН08	РН08-Ф1	Глибоко розуміти загальні принципи та методи інженерії програмного забезпечення, а також методологію наукових досліджень, застосовувати їх у власних дослідженнях та у викладацькій практиці.
РН14	РН14-Ф1	Розробляти, досліджувати та вдосконалювати програмні системи на основі моделей та засобів інтелектуальної обробки даних розподілених і масштабованих систем з урахуванням вимог до надійності їх структурно-алгоритмічного забезпечення.

3 БАЗОВІ ДИСЦИПЛІНИ

Дисципліна не потребує додаткових вимог до базових дисциплін. Міждисциплінарні зв'язки: курс ґрунтується на знаннях, отриманих з вивчених дисциплін за попереднім рівнем освіти.

4 ОБСЯГ І РОЗПОДІЛ ЗА ФОРМАМИ ОРГАНІЗАЦІЇ ОСВІТНЬОГО ПРОЦЕСУ ТА ВИДАМИ НАВЧАЛЬНИХ ЗАНЯТЬ

Вид навчальних занять	Обсяг, години	Розподіл за формами навчання, години			
		денна		заочна	
		аудиторні заняття	самостійна робота	аудиторні заняття	самостійна робота
лекційні	106	26	80		
практичні	66	26	40		
контрольні заходи	8	8	-		
РАЗОМ	180	60	120		

5 ПРОГРАМА ДИСЦИПЛІНИ ЗА ВИДАМИ НАВЧАЛЬНИХ ЗАНЯТЬ

Шифри ДРН	Види та тематика навчальних занять	Обсяг складових, години
	ЛЕКЦІЇ	106
РН03-Ф1 РН07-Ф1 РН08-Ф1 РН14-Ф1	ТЕМА 1. ПАРАЛЕЛЬНІ І РОЗПОДІЛЕНІ ОБЧИСЛЕННЯ Поняття, класифікація архітектур, еволюція парадигм. Сучасний науковий підхід і рішення у даному питанні. 1. Вступ (поняття та концепції паралельних і розподілених обчислень; відмінності між паралельними та розподіленими обчисленнями). 2. Класифікація архітектур (архітектури паралельних обчислень; SIMD (Single Instruction, Multiple Data); MIMD (Multiple Instruction, Multiple Data); Класифікація Флінна; архітектури розподілених обчислень (кластери, ґрид-обчислення, хмарні обчислення)). 3. Еволюція парадигм (розвиток паралельних обчислень (від перших суперкомп'ютерів до сучасних багатоядерних процесорів); розвиток розподілених обчислень (від локальних мереж до глобальних розподілених систем)). 4. Сучасний науковий підхід і рішення (сучасні методи та інструменти (використання мов програмування (MPI, OpenMP для паралельних обчислень); інструменти для розподілених обчислень (Apache Hadoop, Spark)); актуальні дослідження та інновації (огляд сучасних наукових статей та досліджень). 5. Висновки (переваги використання паралельних і розподілених обчислень у різних галузях; основні виклики та проблеми, з якими стикаються дослідники)	8
РН03-Ф1 РН07-Ф1 РН08-Ф1 РН14-Ф1	ТЕМА 2. МОДЕЛІ ПАРАЛЕЛЬНИХ ОБЧИСЛЕНЬ PRAM, BSP, Dataflow, SIMD/MIMD, MapReduce. 1. Вступ (поняття та концепції моделей паралельних обчислень, важливість моделей паралельних обчислень у сучасних системах). 2. PRAM (Parallel Random Access Machine) (принципи PRAM; модель обчислень з спільною пам'яттю; типи PRAM: EREW, CREW, CRCW; приклади використання, алгоритми для PRAM). 3. BSP (Bulk Synchronous Parallel) (принципи BSP; модель обчислень з синхронізацією; фази обчислень: обчислення, комунікація, синхронізація; приклади використання; алгоритми для BSP). 4. Dataflow (принципи Dataflow; модель обчислень, заснована на потоках даних; вузли та ребра в графах обчислень; приклади використання; реалізація Dataflow в сучасних системах). 5. SIMD/MIMD (принципи SIMD (Single Instruction, Multiple Data); виконання однієї інструкції над множиною даних; приклади використання: графічні процесори; принципи MIMD (Multiple Instruction, Multiple Data); виконання множини інструкцій над множиною даних; приклади використання: багатоядерні процесори). 6. MapReduce (принципи MapReduce; модель обчислень для обробки великих обсягів даних; фази обчислень: Map, Shuffle, Reduce; приклади використання; реалізація MapReduce в хмарних обчисленнях). 7. Висновки (переваги використання різних моделей паралельних обчислень; основні виклики та проблеми, з якими стикаються дослідники).	8
РН03-Ф1 РН07-Ф1 РН08-Ф1 РН14-Ф1	ТЕМА 3. МОДЕЛІ РОЗПОДІЛЕНИХ СИСТЕМ Модель клієнт-сервер, агентно-орієнтована модель, мікросервіси. 1. Вступ (поняття та концепції моделей розподілених систем; важливість розподілених систем у сучасних інформаційних технологіях). 2. Модель клієнт-сервер (принципи моделі клієнт-сервер; взаємодія між клієнтом і сервером; типи клієнт-серверних архітектур:	8

	товстий клієнт, тонкий клієнт; приклади використання; використання CDN для підвищення ефективності клієнт-серверних систем). 3. Агентно-орієнтована модель (принципи агентно-орієнтованої моделі; взаємодія автономних агентів; структурні елементи та етапи побудови агентно-орієнтованих моделей; приклади використання; агентно-орієнтоване моделювання в різних системах). 4. Мікросервіси (принципи мікросервісної архітектури; декомпозиція монолітних додатків на мікросервіси; використання контейнеризації (Docker) для розгортання мікросервісів; приклади використання; організація мікросервісів на серверних групах Kubernetes). 5. Висновки (переваги використання різних моделей розподілених систем; виклики та проблеми, з якими стикаються дослідники)	
PH03-Ф1 PH07-Ф1 PH08-Ф1 PH14-Ф1	ТЕМА 4. МЕТОДИ СИНХРОНІЗАЦІЇ ТА ВЗАЄМОДІЇ В ПАРАЛЕЛЬНИХ СИСТЕМАХ Взаємовиключення, бар'єри, семафори, монітори. 1. Вступ (синхронізація та взаємодія в паралельних системах; визначення та основні концепції; важливість синхронізації для забезпечення коректності та ефективності паралельних обчислень). 2. Взаємовиключення (принципи взаємовиключення; взаємовиключення як метод запобігання одночасному доступу до спільних ресурсів; використання замків (locks) та м'ютексів (mutexes); приклади використання; реалізація взаємовиключення в різних мовах програмування). 3. Бар'єри (принципи бар'єрів; бар'єри як метод синхронізації, що забезпечує одночасне завершення певних етапів обчислень; використання бар'єрів у багатопроцесорних системах; приклади використання; реалізація бар'єрів у бібліотеках паралельного програмування). 4. Семафори (принципи семафорів; семафори як метод управління доступом до ресурсів; типи семафорів: бінарні та рахункові; приклади використання; реалізація семафорів у різних мовах програмування). 5. Монітори (принципи моніторів; монітори як метод синхронізації, що забезпечує взаємовиключення та умови синхронізації; використання моніторів у об'єктно-орієнтованих мовах програмування; приклади використання; реалізація моніторів у різних мовах програмування). 6. Висновки (переваги використання різних методів синхронізації; основні виклики та проблеми, з якими стикаються дослідники).	8
PH03-Ф1 PH07-Ф1 PH08-Ф1 PH14-Ф1	ТЕМА 5. АЛГОРИТМИ ТА МОДЕЛІ УЗГОДЖЕНОСТІ В РОЗПОДІЛЕНИХ СИСТЕМАХ CAP-теорема, моделі узгодженості (strong/eventual/causal consistency). 1. Вступ (поняття узгодженості в розподілених системах; визначення та основні концепції; важливість узгодженості для забезпечення коректності та надійності розподілених систем). 2. CAP-теорема (принципи CAP-теорема; визначення та пояснення властивостей: узгодженість (Consistency), доступність (Availability), стійкість до розділення (Partition Tolerance); компроміси між цими властивостями в розподілених системах; приклади використання; реалізація CAP-теорема в сучасних розподілених системах). 3. Моделі узгодженості (Strong Consistency (сильна узгодженість), визначення та основні принципи, переваги та недоліки використання сильної узгодженості; Eventual Consistency (кінцева узгодженість), визначення та основні принципи, переваги та недоліки використання кінцевої узгодженості; Causal Consistency (причинно-наслідкова узгодженість), визначення та основні принципи, переваги та недоліки використання причинно-	8

	наслідкової узгодженості). 4. Алгоритми узгодженості (алгоритми для забезпечення сильної узгодженості: Paxos, Raft; алгоритми для забезпечення кінцевої узгодженості: Dynamo, Riak; алгоритми для забезпечення причиннонаслідкової узгодженості: COPS, Eiger). 5. Висновки (переваги використання різних моделей узгодженості; основні виклики та проблеми, з якими стикаються дослідники)	
PH03-Ф1 PH07-Ф1 PH08-Ф1 PH14-Ф1	ТЕМА 6. ІНСТРУМЕНТИ ПОБУДОВИ ПАРАЛЕЛЬНОГО ПЗ MPI, OpenMP, CUDA, багатопоточність у Java/.NET. 1. Вступ (поняття паралельного програмного забезпечення; визначення та основні концепції; важливість паралельного програмування у сучасних обчислювальних системах). 2. MPI (Message Passing Interface) (принципи MPI; модель передачі повідомлень для паралельних обчислень; функції та бібліотеки MPI; приклади використання; реалізація паралельних алгоритмів з використанням MPI). 3. OpenMP (Open Multi-Processing) (принципи OpenMP; модель паралельного програмування для багатоядерних процесорів; директиви та функції OpenMP; приклади використання; реалізація паралельних обчислень з використанням OpenMP). 4. CUDA (Compute Unified Device Architecture) (принципи CUDA; модель паралельного програмування для графічних процесорів (GPU); функції та бібліотеки CUDA; приклади використання; реалізація паралельних алгоритмів з використанням CUDA). 5. Багатопоточність у Java/.NET (принципи багатопоточності; моделі багатопоточності у Java та .NET; приклади використання; реалізація багатопоточних програм у Java та .NET). 6. Висновки (переваги використання різних інструментів для побудови паралельного ПЗ; основні виклики та проблеми, з якими стикаються розробники)	8
PH03-Ф1 PH07-Ф1 PH08-Ф1 PH14-Ф1	ТЕМА 7. ТЕХНОЛОГІЇ СТВОРЕННЯ РОЗПОДІЛЕНОГО ПЗ gRPC, Apache Kafka, REST, GraphQL, Docker, Kubernetes. 1. Вступ (поняття розподіленого програмного забезпечення; визначення та основні концепції; важливість розподіленого ПЗ у сучасних інформаційних технологіях). 2. gRPC (Google Remote Procedure Call) (принципи gRPC; високопродуктивний RPC фреймворк; використання Protocol Buffers для серіалізації даних; приклади використання; реалізація мікросервісів з використанням gRPC). 3. Apache Kafka (принципи Apache Kafka; платформа для обробки потоків даних у реальному часі; використання Kafka Streams для обробки даних; приклади використання; реалізація системи обробки даних у реальному часі з використанням Apache Kafka). 4. REST (Representational State Transfer) (принципи REST; архітектурний стиль для розробки веб-сервісів; використання HTTP методів для взаємодії з ресурсами; приклади використання; реалізація REST API для веб-додатків). 5. GraphQL (принципи GraphQL; запитова мова для API; використання схеми для визначення структури даних; приклади використання; реалізація API з використанням GraphQL). 6. Docker (принципи Docker; платформа для контейнеризації додатків; використання Docker для ізоляції та переносимості додатків; приклади використання; реалізація мікросервісів з використанням Docker). 7. Kubernetes (принципи Kubernetes; система оркестрації контейнерів; використання Kubernetes для автоматизації розгортання, масштабування та управління контейнерними додатками; приклади використання; реалізація кластерів контейнерів з використанням Kubernetes). 8. Висновки (переваги використання різних технологій	8

	для створення розподіленого ПЗ; основні виклики та проблеми, з якими стикаються розробники)	
PH03-Ф1 PH07-Ф1 PH08-Ф1 PH14-Ф1	ТЕМА 8. ПАРАЛЕЛЬНА ТА РОЗПОДІЛЕНА ОБРОБКА ВЕЛИКИХ ДАНИХ Hadoop, Spark, Flink: концепції, архітектура, приклади. 1. Вступ (поняття великих даних; визначення та основні концепції; важливість паралельної та розподіленої обробки великих даних у сучасних інформаційних технологіях). 2. Hadoop (принципи Hadoop; концепція та архітектура Hadoop; компоненти Hadoop: HDFS, MapReduce, YARN; приклади використання; реалізація обробки великих даних з використанням Hadoop). 3. Spark (принципи Spark; концепція та архітектура Spark; компоненти Spark: Spark Core, Spark SQL, Spark Streaming, MLlib, GraphX; приклади використання; реалізація обробки великих даних з використанням Spark). 4. Flink (принципи Flink; концепція та архітектура Flink; компоненти Flink: DataStream API, DataSet API, Flink ML; приклади використання; реалізація обробки великих даних з використанням Flink). 5. Порівняння Hadoop, Spark та Flink (переваги та недоліки; порівняння продуктивності та ефективності; вибір технології залежно від конкретних завдань). 6. Висновки (переваги використання паралельної та розподіленої обробки великих даних; основні виклики та проблеми, з якими стикаються розробники)..	10
PH03-Ф1 PH07-Ф1 PH08-Ф1 PH14-Ф1	ТЕМА 9. ЗАБЕЗПЕЧЕННЯ НАДІЙНОСТІ РОЗПОДІЛЕНИХ СИСТЕМ Fault tolerance, реплікація, checkpointing, Paxos, Raft. 1. Вступ (поняття надійності в розподілених системах; визначення та основні концепції; важливість надійності для забезпечення коректності та безперервності роботи розподілених систем). 2. Fault Tolerance (Стійкість до відмов) (основні принципи fault tolerance; визначення та методи забезпечення стійкості до відмов; використання резервування та дублювання компонентів; приклади використання; використання глибокого навчання для забезпечення fault tolerance). 3. Реплікація (принципи реплікації; визначення та методи реплікації даних; типи реплікації: синхронна та асинхронна; приклади використання; реалізація розподілених ACID транзакцій з використанням реплікації). 4. Checkpointing (принципи checkpointing; визначення та методи збереження стану системи; типи checkpointing: координоване, незалежне, квазі-синхронне; приклади використання; реалізація checkpointing у розподілених системах). 5. Paxos (принципи алгоритму Paxos; визначення та основні етапи алгоритму Paxos); використання Paxos для досягнення консенсусу в розподілених системах; приклади використання; реалізація Paxos у сучасних розподілених системах). 6. Raft (принципи алгоритму Raft; визначення та основні етапи алгоритму Raft; використання Raft для досягнення консенсусу в розподілених системах; приклади використання; оптимізація використання Raft у розподілених системах). 7. Висновки (переваги використання різних методів забезпечення надійності; основні виклики та проблеми, з якими стикаються розробники)	10
PH03-Ф1 PH07-Ф1 PH08-Ф1 PH14-Ф1	ТЕМА 10. ІНТЕЛЕКТУАЛЬНА ОБРОБКА ДАНИХ У РОЗПОДІЛЕНИХ СИСТЕМАХ Моделі машинного навчання в Spark/Flink, розподілені MLфреймворки. 1. Вступ (поняття інтелектуальної обробки даних; визначення та основні концепції; важливість інтелектуальної обробки даних у сучасних інформаційних технологіях). 2. Моделі машинного навчання	10

	в Spark (принципи Spark; концепція та архітектура Spark; компоненти Spark: Spark Core, Spark SQL, Spark Streaming, MLlib, GraphX; моделі машинного навчання в Spark; використання MLlib для реалізації моделей машинного навчання; приклади використання: класифікація, регресія, кластеризація). 3. Моделі машинного навчання в Flink (принципи Flink; концепція та архітектура Flink; компоненти Flink: DataStream API, DataSet API, Flink ML; моделі машинного навчання в Flink; використання Flink ML для реалізації моделей машинного навчання; приклади використання: обробка поточкових даних, аналіз великих даних). 4. Розподілені ML-фреймворки (принципи розподілених MLфреймворків; TensorFlow, PyTorch, Horovod; використання розподілених ML-фреймворків для масштабування моделей машинного навчання; приклади використання; реалізація розподілених моделей машинного навчання з використанням TensorFlow та PyTorch). 5. Порівняння моделей машинного навчання в Spark та Flink(переваги та недоліки; порівняння продуктивності та ефективності; вибір технології залежно від конкретних завдань). 6. Висновки (переваги використання інтелектуальної обробки даних у розподілених системах; основні виклики та проблеми, з якими стикаються розробники)	
PH03-Ф1 PH07-Ф1 PH08-Ф1 PH14-Ф1	ТЕМА 11. МОДЕЛЮВАННЯ ТА ВЕРИФІКАЦІЯ ПРОГРАМНИХ СИСТЕМ Формальні методи, Petri net, TLA+, моделювання поведінки. 1. Вступ (поняття моделювання та верифікації програмних систем; визначення та основні концепції; важливість моделювання та верифікації для забезпечення коректності та надійності програмних систем). 2. Формальні методи (принципи формальних методів; визначення та основні техніки формальних методів; використання математичних технік для специфікації, верифікації та доказу правильності програм; приклади використання; використання формальних методів у реальних проектах, таких як системи реального часу та безпечні атомні електростанції). 3. Petri net (основні принципи мереж Петрі; визначення та основні компоненти мереж Петрі: позиції, переходи, маркування; використання мереж Петрі для моделювання динамічних комп'ютерних систем; приклади використання; реалізація моделювання паралельних процесів за допомогою мереж Петрі). 4. TLA+ (Temporal Logic of Actions) (принципи TLA+; визначення та основні концепції TLA+; використання TLA+ для специфікації та верифікації розподілених систем; приклади використання; реалізація моделей поведінки та перевірка коректності програмних систем з використанням TLA+). 5. Моделювання поведінки (принципи моделювання поведінки; використання скінчених автоматів для моделювання поведінки об'єктів; використання UML для візуального моделювання поведінки систем; приклади використання; реалізація чат-ботів з поведінкою, що керується скінченим автоматом). 6. Висновки (переваги використання різних методів моделювання та верифікації; основні виклики та проблеми, з якими стикаються розробники)	10
PH03-Ф1 PH07-Ф1 PH08-Ф1 PH14-Ф1	ТЕМА 12. МЕТОДОЛОГІЯ НАУКОВИХ ДОСЛІДЖЕНЬ У ГАЛУЗІ РОЗПОДІЛЕНИХ ОБЧИСЛЕНЬ Постановка задач, огляд літератури, вибір методів, оцінювання результатів. 1. Вступ (поняття методології наукових досліджень галузі розподілених обчислень; визначення та основні концепції; важливість методології у наукових дослідженнях розподілених обчислень). 2.	10

	Постановка задач (основні принципи постановки задач; визначення проблеми та формулювання дослідницьких питань та приклади постановки задач у розподілених обчисленнях; методи постановки задач; використання динамічного програмування для оптимального розподілу завдань; планування завдань у розподілених системах з обмеженою інформацією про ресурси). 3. Огляд літератури (принципи огляду літератури; важливість систематичного огляду літератури; методи пошуку та аналізу наукових статей; приклади огляду літератури; порівняльний аналіз технологій розподілених обчислень). 4. Вибір методів (принципи вибору методів; визначення критеріїв вибору методів для дослідження; порівняння потокової та пакетної обробки даних; приклади вибору методів; використання статистичних алгоритмів та кластерного аналізу). 5. Оцінювання результатів (принципи оцінювання результатів; визначення метрик та критеріїв оцінювання; методи аналізу та інтерпретації результатів; приклади оцінювання результатів; аналіз ефективності хмарних обчислень для розподілених систем). 6. Висновки (переваги використання систематичного підходу до наукових досліджень; основні виклики та проблеми, з якими стикаються дослідники).	
	ПРАКТИЧНІ ЗАНЯТТЯ	66
PH03-Ф1 PH07-Ф1 PH08-Ф1	Практична робота №1 ПОБУДОВА БАГАТОПОТОЧНОГО ЗАСТОСУНКУ З OPENMP АБО STD-THREAD Аналіз продуктивності. Мета: Особливості функціонування багатопоточності та паралельного програмування. Використання OpenMP або std::thread для створення багатопоточних застосунків. Провести аналіз продуктивності створених застосунків.	12
PH03-Ф1 PH07-Ф1 PH08-Ф1	Практична робота №2 РОЗРОБКА КЛІЄНТ-СЕРВЕРНОГО ЗАСТОСУНКУ З ВИКОРИСТАННЯМ GRPC АБО REST Реалізація взаємодії та протоколів. Мета: Особливості використання gRPC або REST для розробки клієнт-серверних застосунків. Реалізувати взаємодію та протоколи між клієнтом і сервером. Провести аналіз продуктивності виконаних завдань	12
PH03-Ф1 PH07-Ф1 PH08-Ф1 PH14-Ф1	Практична робота №3 ВИКОРИСТАННЯ APACHE SPARK ДЛЯ ОБРОБКИ ВЕЛИКИХ ОБСЯГІВ ДАНИХ Завдання ETL, агрегація, обчислення. Мета: Ознайомитися з особливостями використання Apache Spark для обробки великих обсягів даних. Навчитися виконувати завдання ETL (Extract, Transform, Load), агрегацію та обчислення з використанням Apache Spark. Провести аналіз продуктивності виконаних завдань	12
PH07-Ф1 PH08-Ф1 PH14-Ф1	Практична робота №4 МОДЕЛЮВАННЯ РОЗПОДІЛЕНОГО ПРОТОКОЛУ УЗГОДЖЕННЯ (НАПРИКЛАД, RAFT) Імітаційне середовище або симулятор. Мета: Ознайомитися з особливостями функціонування розподілених протоколів узгодження. Навчитися моделювати протокол узгодження	10

	Raft. Використовувати імітаційне середовище або симулятор для перевірки коректності та ефективності протоколу.	
RH07-Ф1 RH08-Ф1 RH14-Ф1	Практична робота №5 ЗАСТОСУВАННЯ ІНСТРУМЕНТІВ РОЗПОДІЛЕНОГО МАШИННОГО НАВЧАННЯ (НАПРИКЛАД, MLLIB, TENSORFLOW ON KUBERNETES) Побудова та тренування моделі. Мета: Інструменти розподіленого машинного навчання. Навчитися використовувати MLib або TensorFlow на Kubernetes для побудови та тренування моделей машинного навчання. Провести аналіз продуктивності та ефективності тренуваних моделей.	10
RH07-Ф1 RH08-Ф1 RH14-Ф1	Практична робота №6 ФОРМАЛЬНЕ МОДЕЛЮВАННЯ ПОВЕДІНКИ ПАРАЛЕЛЬНОЇ СИСТЕМИ (НАПРИКЛАД, У TLA+ АБО PROMELA) Перевірка властивостей: безпека, завершеність, взаємовиключення. Мета: Ознайомитися з методами формального моделювання поведінки паралельних систем. Навчитися використовувати TLA+ або Promela для моделювання та перевірки властивостей паралельних систем. Перевірити властивості безпеки, завершеності та взаємовиключення.	10
Контрольний захід		8
ЗАГАЛЬНА КІЛЬКІСТЬ		180

6 ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ

Сертифікація досягнень здобувачів здійснюється за допомогою прозорих процедур, що ґрунтуються на об'єктивних критеріях відповідно до Положення університету «Про оцінювання результатів навчання здобувачів вищої освіти».

Досягнутий рівень компетентностей відносно очікуваних, що ідентифікований під час контрольних заходів, відображає реальний результат навчання здобувача за дисципліною.

6.1 Шкали

Оцінювання навчальних досягнень здобувачів НТУ «ДП» здійснюється за рейтинговою (100-бальною) та інституційною шкалами. Остання необхідна (за офіційною відсутністю національної шкали) для конвертації (переведення) оцінок мобільних здобувачів

Шкали оцінювання навчальних досягнень здобувачів вищої освіти НТУ «ДП»

Рейтингова	Інституційна
90...100	відмінно / Excellent
74...89	добре / Good
60...73	задовільно / Satisfactory
0...59	незадовільно / Fail

Кредити навчальної дисципліни зараховуються, якщо здобувач отримав підсумкову оцінку не менше 60-ти балів. Нижча оцінка вважається академічною

заборгованістю, що підлягає ліквідації відповідно до Положення про організацію освітнього процесу НТУ «ДП».

6.2 Засоби та процедури

Зміст засобів діагностики спрямовано на контроль рівня сформованості знань, умінь/навичок, комунікації, автономії та відповідальності здобувача за вимогами НРК до 8-го кваліфікаційного рівня (для третього (освітньо-наукового) рівня вищої освіти) під час демонстрації регламентованих робочою програмою результатів навчання.

Здобувач на контрольних заходах має виконувати завдання, орієнтовані виключно на демонстрацію дисциплінарних результатів навчання (розділ 2).

Засоби діагностики, що надаються здобувачам на контрольних заходах у вигляді завдань для поточного та підсумкового контролю, формуються шляхом конкретизації вихідних даних та способу демонстрації дисциплінарних результатів навчання.

Засоби діагностики (контрольні завдання) для поточного та підсумкового контролю дисципліни затверджуються кафедрою.

Види засобів діагностики та процедур оцінювання для поточного та підсумкового контролю дисципліни подано нижче.

Засоби діагностики та процедури оцінювання

ПОТОЧНИЙ КОНТРОЛЬ			ПІДСУМКОВИЙ КОНТРОЛЬ	
навчальне заняття	засоби діагностики	процедури	засоби діагностики	процедури
лекції	контрольні завдання за кожною темою	виконання завдання під час лекцій	комплексна контрольна робота (ККР)	визначення середньозваженого результату поточних контролів;
практичні	індивідуальне завдання	виконання завдань під час самостійної роботи		виконання ККР під час заліку за бажанням здобувача

Під час поточного контролю лекційні заняття оцінюються шляхом визначення якості виконання контрольних конкретизованих завдань. Практичні заняття оцінюються якістю виконання (згідно таблиці) індивідуального завдання.

Якщо зміст певного виду занять підпорядковано декільком складовим, то інтегральне значення оцінки може визначатися з урахуванням вагових коефіцієнтів, що встановлюються викладачем.

За наявності рівня результатів поточних контролів з усіх видів навчальних занять не менше 60 балів, підсумковий контроль здійснюється без участі здобувача шляхом визначення середньозваженого значення поточних оцінок.

Незалежно від результатів поточного контролю кожен здобувач під час заліку має право виконувати ККР, яка містить завдання, що охоплюють ключові дисциплінарні результати навчання.

Кількість конкретизованих завдань ККР повинна відповідати відведеному часу на виконання. Кількість варіантів ККР має забезпечити індивідуалізацію завдання.

Значення оцінки за виконання ККР визначається середньою оцінкою складових (конкретизованих завдань) і є остаточним.

Інтегральне значення оцінки виконання ККР може визначатися з урахуванням вагових коефіцієнтів, що встановлюється кафедрою для кожної складової опису кваліфікаційного рівня НРК.

6.3 Критерії

Реальні результати навчання здобувача ідентифікуються та вимірюються відносно очікуваних під час контрольних заходів за допомогою критеріїв, що описують дії здобувача для демонстрації досягнення результатів навчання.

Для оцінювання виконання контрольних завдань під час поточного контролю лекційних і практичних занять в якості критерію використовується коефіцієнт засвоєння, що автоматично адаптує показник оцінки до рейтингової шкали:

$$O_i = 100 a/m,$$

де a – число правильних відповідей або виконаних суттєвих операцій відповідно до еталону рішення; m – загальна кількість запитань або суттєвих операцій еталону.

Індивідуальні завдання та комплексні контрольні роботи оцінюються експертно за допомогою критеріїв, що характеризують співвідношення вимог до рівня компетентностей і показників оцінки за рейтинговою шкалою.

Зміст критеріїв спирається на компетентнісні характеристики, визначені НРК для освітньо-наукового рівня вищої освіти (подано нижче).

**Загальні критерії досягнення результатів навчання для 8-го
кваліфікаційного рівня за НРК
(доктор філософії)**

Опис кваліфікаційного рівня	Вимоги до знань, умінь/навичок, комунікації, відповідальності і автономії	Показник оцінки
Знання		
– Концептуальні та методологічні знання в галузі чи на межі галузей знань або професійної діяльності.	Відповідь відмінна – правильна, обґрунтована, осмислена. Характеризує наявність: 1. спеціалізованих концептуальних знань на рівні новітніх досягнень; 2. критичне осмислення проблем у навчанні та/або професійній діяльності та на межі предметних галузей	95-100
	Відповідь містить не грубі помилки або описки	90-94
	Відповідь правильна, але має певні неточності	85-89
	Відповідь правильна, але має певні неточності й недостатньо обґрунтована	80-84
	Відповідь правильна, але має певні неточності, недостатньо обґрунтована та осмислена	74-79
	Відповідь фрагментарна	70-73
	Відповідь демонструє нечіткі уявлення про об'єкт вивчення	65-69
	Рівень знань мінімально задовільний	60-64
	Рівень знань незадовільний	<60
Уміння/навички		
– Спеціалізовані уміння/навички і методи, необхідні для розв'язання значущих проблем у сфері професійної діяльності, науки та/або інновацій, розширення та переоцінки вже існуючих знань і професійної	Відповідь характеризує уміння: – виявляти проблеми; – формулювати гіпотези; – розв'язувати проблеми; – оновлювати знання; – інтегрувати знання; – провадити інноваційну діяльність; – провадити наукову діяльність	95-100
	Відповідь характеризує уміння/навички застосовувати знання в практичній діяльності з не грубими помилками	90-94
	Відповідь характеризує уміння/навички застосовувати знання в практичній діяльності, але має певні неточності	85-89

Опис кваліфікаційного рівня	Вимоги до знань, умінь/навичок, комунікації, відповідальності і автономії	Показник оцінки
практики; – започаткування, планування, реалізація та коригування послідовного процесу ґрунтового наукового дослідження з дотриманням належної академічної доброчесності; – критичний аналіз, оцінка і синтез нових та комплексних ідей.	при реалізації однієї вимоги	
	Відповідь характеризує уміння/навички застосовувати знання в практичній діяльності, але має певні неточності при реалізації двох вимог	80-84
	Відповідь характеризує уміння/навички застосовувати знання в практичній діяльності, але має певні неточності при реалізації трьох вимог	74-79
	Відповідь характеризує уміння/навички застосовувати знання в практичній діяльності, але має певні неточності при реалізації чотирьох вимог	70-73
	Відповідь характеризує уміння/навички застосовувати знання в практичній діяльності при виконанні завдань за зразком	65-69
	Відповідь характеризує уміння/навички застосовувати знання при виконанні завдань за зразком, але з неточностями	60-64
	Рівень умінь/навичок незадовільний	<60
Комунікація		
– Вільне спілкування з питань, що стосуються сфери наукових та експертних знань, з колегами, широкою науковою спільнотою, суспільством у цілому; – використання академічної української та іноземної мови у професійній діяльності та дослідженнях.	Зрозумілість відповіді (доповіді). <i>Мова:</i> правильна; чиста; ясна; точна; логічна; виразна; лаконічна. <i>Комунікаційна стратегія:</i> – послідовний і несуперечливий розвиток думки; – наявність логічних власних суджень; – доречна аргументації та її відповідність відстоюваним положенням; – правильна структура відповіді (доповіді); – правильність відповідей на запитання; – доречна техніка відповідей на запитання; – здатність робити висновки та формулювати пропозиції; – використання іноземних мов у професійній діяльності	95-100
	Достатня зрозумілість відповіді (доповіді) та доречна комунікаційна стратегія з незначними хибами	90-94
	Добра зрозумілість відповіді (доповіді) та доречна комунікаційна стратегія (сумарно не реалізовано три вимоги)	85-89
	Добра зрозумілість відповіді (доповіді) та доречна комунікаційна стратегія (сумарно не реалізовано чотири вимоги)	80-84
	Добра зрозумілість відповіді (доповіді) та доречна комунікаційна стратегія (сумарно не реалізовано п'ять вимог)	74-79
	Задовільна зрозумілість відповіді (доповіді) та доречна комунікаційна стратегія (сумарно не реалізовано сім вимог)	70-73
	Задовільна зрозумілість відповіді (доповіді) та комунікаційна стратегія з хибами (сумарно не реалізовано дев'ять вимог)	65-69

Опис кваліфікаційного рівня	Вимоги до знань, умінь/навичок, комунікації, відповідальності і автономії	Показник оцінки
	Задовільна зрозумілість відповіді (доповіді) та комунікаційна стратегія з хибамі (сумарно не реалізовано 10 вимог)	60-64
	Рівень комунікації незадовільний	<60
Відповідальність і автономія		
– Демонстрація значної авторитетності, інноваційності, високий ступінь самостійності, академічна та професійна доброчесність, постійна відданість розвитку нових ідей або процесів у передових контекстах професійної та наукової діяльності; – здатність до безперервного саморозвитку та самовдосконалення.	Відмінне володіння компетенціями: – використання принципів та методів організації діяльності команди; – ефективний розподіл повноважень в структурі команди; – підтримка врівноважених стосунків з членами команди (відповідальність за взаємовідносини); – стресовитривалість; – саморегуляція; – трудова активність в екстремальних ситуаціях; – високий рівень особистого ставлення до справи; – володіння всіма видами навчальної діяльності; – належний рівень фундаментальних знань; – належний рівень сформованості загальнонавчальних умінь і навичок	95-100
	Упевнене володіння компетенціями відповідальності і автономії з незначними хибамі	90-94
	Добре володіння компетенціями відповідальності і автономії (не реалізовано дві вимоги)	85-89
	Добре володіння компетенціями відповідальності і автономії (не реалізовано три вимоги)	80-84
	Добре володіння компетенціями відповідальності і автономії (не реалізовано чотири вимоги)	74-79
	Задовільне володіння компетенціями відповідальності і автономії (не реалізовано п'ять вимог)	70-73
	Задовільне володіння компетенціями відповідальності і автономії (не реалізовано шість вимог)	65-69
	Задовільне володіння компетенціями відповідальності і автономії (рівень фрагментарний)	60-64
	Рівень відповідальності і автономії незадовільний	<60

7 ІНСТРУМЕНТИ, ОБЛАДНАННЯ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ

Технічні засоби навчання: мультимедійні та комп'ютерні пристрої.

Засоби дистанційної освіти: Moodle, MS Teams.

Пакети приладних програм: MS Office, StatGraphics (навчальна безкоштовна версія), MathCad (навчальна безкоштовна версія). Інтерпритатор мови Python.

8 РЕКОМЕНДОВАНІ ДЖЕРЕЛА ІНФОРМАЦІЇ

Базова:

1. Минайленко Р.М. Паралельні та розподілені обчислення: навч. посіб. — Кропивницький: Видавець Лисенко В. Ф., 2021. — 153 с. <https://dspace.kntu.kr.ua/server/api/core/bitstreams/396e02d2-725b-47b5-a1c0-ae07a9bec326/content>
2. Семеренко, В. П. Технології паралельних обчислень: навчальний посібник / Семеренко В. П. – Вінниця: ВНТУ, 2018. – 104 с.
3. Коломійцев О., Рибальченко А., Голубничий Д., Третяк В., Воронін, В., Комаров, В., Пустоваров, В., Філіппенков, О., Любченко, О., & Рудаков, І. (2023). ПРОБЛЕМИ РОЗВИТКУ ПАРАЛЕЛЬНИХ ОБЧИСЛЮВАЛЬНИХ СИСТЕМ ТА ЇХ МАТЕМАТИЧНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ РІШЕННЯ ЗАДАЧ ДИСКРЕТНОЇ ОПТИМІЗАЦІЇ. Grail of Science, (32), 182–197. <https://doi.org/10.36074/grail-of-science.13.10.2023.032>
4. McCarty S. Docker really did change the world. InfoWorld. URL: <https://www.infoworld.com/article/3639596/docker-really-did-change-the-world.html>
5. Mardeni A. Why is Docker so Popular. Engineering Education (EngEd) Program | Section. URL: <https://www.section.io/engineering-education/why-is-docker-so-popular/>
6. Docker documentation. URL: <https://docs.docker.com/>
7. Kubernetes documentation. URL: <https://kubernetes.io/docs/home/>
8. Бараненко, Роман. “ТЕХНОЛОГІЇ РОЗПОДІЛЬНИХ СИСТЕМ ТА ПАРАЛЕЛЬНИХ ОБЧИСЛЕНЬ.” MEASURING AND COMPUTING DEVICES IN TECHNOLOGICAL PROCESSES, Khmelnytskyi National University.
9. Ghani, A., Ahmad, S., Khan, M. A., Khalid, S., Zohaib, S. (2021), "A Survey on Anomaly Detection Techniques for Microservices". IEEE Access. №9. P. 122766– 122805. DOI: <https://ieeexplore.ieee.org/document/10479425>
10. Li, Z., Zhang, W., Li, Z., Zhao, M. (2020), "Anomaly Detection for Microservices: A Survey". ACM Computing Surveys. №53. P. 1–42. DOI: <https://dl.acm.org/doi/10.1145/3639478.3643535>
11. Luo, J., Wu, J., Zhang, Y., Sun, L., Liu, Y. (2022), "Anomaly Detection for Microservices Based on Federated Learning". In: 2022 IEEE International Conference on Web Services (ICWS). IEEE, P. 582–589. DOI: <https://ieeexplore.ieee.org/document/9461272>
12. He, P., Ranjan, R., Nogueira, J., Veiga, L. M., Zhao, W. (2021), "A Survey on Anomaly Detection for Microservices in Cloud Computing". Journal of Systems and Software. №182. 318 p. DOI: <https://journalofcloudcomputing.springeropen.com/articles/10.1186/s13677-022-00296-4>
13. Patel, A. “Error Management in Data Source Connections: The Role of DSN”. Journal of Computing and Information Technology, 2023, 31(1), 75-88, DOI: 10.1109/CIT.2023.03168
14. Chen, M. “Optimizing Connection Settings for Enhanced Performance: A DSN Approach”. Database Systems Review, 2022, 33(3), 167-182, DOI: 10.1007/s11042-022-12568-0
15. Юрій Клушин. Підвищення ефективності в системах клієнт-сервер через використання мереж CDN. CSN. 2024; Випуск 6, Номер 2. : сс. 93-106.

<https://doi.org/10.23939/csn2024.02.093>

16. <https://www.webscale.com/engineering-education/>

17. Дивак М., Кіндзерський О. "Дослідження ефективності паралельної обчислювальної схеми ідентифікації інтервальних дискретних моделей на основі ройового інтелекту". Herald of Khmelnytskyi National University. Technical Sciences, 2024, 331(1), pp. 29-37. <https://doi.org/10.31891/2307-5732-2024-331-3>

18. Duchenchuk, V. і Boublik, V. 2020. Моделі узгодженості розподілених систем віртуальної реальності. *Наукові записки НаУКМА. Комп'ютерні науки*. 3, (Груд 2020), 7–11. DOI:<https://doi.org/10.18523/2617-3808.2020.3.7-11>

19. IEEE Article: The Ongoing Evolution of OpenMP. URL: <https://www.openmp.org/updates/ieee-article-the-ongoing-evolution-of-openmp/>

20. OpenMP in a Modern World: From Multi-device Support to Meta Programming. 18th International Workshop on OpenMP, IWOMP 2022, Chattanooga, TN, USA, September 27–30, 2022, Proceedings. URL: [https://link.springer.com/book/10.1007/978-3-](https://link.springer.com/book/10.1007/978-3-031-15922-0)

031-15922-0

21. Новини і статті по темі CUDA. URL: <https://itc.ua/ua/tag/cuda/>

22. Kafka. BOOKS AND PAPERS URL: <https://kafka.apache.org/books-and-papers>

23. B. Yuen, A. Matyushentsev, T. Ekenstam, and J. Suen, "GitOps and Kubernetes: Continuous Deployment with

24. Argo CD, Jenkins X, and Flux" O'Reilly Media, 2021, ISBN: 978-1492094317

25. Bohdan Fedoryshyn, Olena Krasko. MIGRATION OF SERVICES IN A KUBERNETES CLUSTER BASED ON WORKLOAD FORECASTING. ICTEE. 2024; Volume 4, Number 2 : 82-92 <https://doi.org/10.23939/ictee2024.02.082>

26. Mazurova, O., Naboka, A. і Shirokopetleva, M. (2021) «ДОСЛІДЖЕННЯ МЕТОДІВ РЕАЛІЗАЦІЇ РОЗПОДІЛЕНИХ ACID ТРАНЗАКЦІЙ ЗА ТЕХНОЛОГІЄЮ РЕПЛІКАЦІЇ», *СУЧАСНИЙ СТАН НАУКОВИХ ДОСЛІДЖЕНЬ ТА ТЕХНОЛОГІЙ В ПРОМИСЛОВОСТІ*, (2 (16), с. 19–31. doi: 10.30837/ITSSI.2021.16.019.

27. Белоус Р.В., Крилов Є.В. ОПТИМІЗАЦІЯ ВИКОРИСТАННЯ RAFT В

РОЗПОДІЛЕНИХ СИСТЕМАХ З БАЗАМИ ДАНИХ. Інформатика, обчислювальна техніка та автоматизація. URL: <https://doi.org/10.32782/2663-5941/2024.3.1/07>

28. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. URL: <https://www.tensorflow.org/about/bib>

29. Introduction to TensorFlow. URL: <https://www.tensorflow.org/learn>

30. Scikit-learn URL: <https://scikit-learn.org/stable/index.html>

31. Shantyr, A., & Zinchenko, O. (2024). Forecasting software system quality metrics using modifications of the DeepLIFT interpretation method. *INNOVATIVE TECHNOLOGIES AND SCIENTIFIC SOLUTIONS FOR INDUSTRIES*, (4(30), 110–124. <https://doi.org/10.30837/2522-9818.2024.4.110>

32. Козирев, А. і Шубін, І. (2023) «Метод планування завдань оброблення даних у розподілених системах з обмеженою інформацією про доступні ресурси», *СУЧАСНИЙ СТАН НАУКОВИХ ДОСЛІДЖЕНЬ ТА ТЕХНОЛОГІЙ В ПРОМИСЛОВОСТІ*, (3 (25), с. 27–39. doi: 10.30837/ITSSI.2023.25.027.

33. Stanislav Zhuravel, S. Dumych, Olha Shpur. Дослідження методів збору та обробки даних в розподілених інформаційних системах. ICTEE. 2021; Volume 1, Number 1 : 20-38 <https://doi.org/10.23939/ictee2021.01.020/>

34. С. Журавель, С. Думич, О. Шпур. ДОСЛІДЖЕННЯ МЕТОДІВ ЗБОРУ ТА ОБРОБКИ ДАНИХ В РОЗПОДІЛЕНИХ ІНФОРМАЦІЙНИХ СИСТЕМАХ. Information and communication technologies, electronic engineering. Vol. 1, No. 1. pp. 20– 38 (2021)

35. Nagel Chrisian, Professional C# and .NET. Wrox; 2021. - 1008 p.

36. Oaks S. Java Performance: In-depth Advice for Tuning and Programming Java 8, 11, and Beyond. O'Reilly Media, Inc.; 2end Edition, 2020. – 452 p.

Додаткова:

1. Моделі, методи та засоби аналізу надійності програмних систем: монографія / В.С. Яковина, Д.В. Федасюк, М.М. Сенів, О.О. Нитребич. – Львів: Видавництво Львівської політехніки, 2015.-220 с.

2. Ilia Vonta, Mangey Ram. Reliability Engineering: theory and applications (Advanced research in reliability and system assurance engineering). CRC Press, 1st edition, 2018, 228 p.

3. Aman Ullah. Software Reliability in Safety Critical Systems. OmniScriptum Publishing KS. 2016. – 60 p.

4. Nikolay Pavlov, Anton Iliev, Asen Rahnev and Nikolay Kyurkchiev. Some Software Reliability Models. OmniScriptum Publishing KS. 2018. – 124 p.

5. B.S. Dhillon. Enginnering Systems Reliability, Safety and Maintence^ An Integrated Approach. CRC Press, 1st edition. 2019. 298 p.

6. Поджаренко В.О., Васілевський О.М., Кучерук В.Ю. Опрацювання результатів вимірювань на основі концепції невизначеності: навч. посібник. Вінниця: ВНТУ, 2008. 128 с.

7. Czarnul P.Parallel Programming for Modern High Performance Computing Systems// CRC Press, 2018. - 304 p.

Навчальне видання

Геннадій ШВАЧИЧ

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

«Моделі та методи створення програмних систем паралельної та розподіленої
обробки даних»

для докторів філософії спеціальності 121 «Інженерія програмного
забезпечення»

Підготовлено до друку
у Національному технічному університеті
«Дніпровська політехніка».

Свідоцтво про внесення до Державного реєстру ДК № 1842 від 11.06.2004

49005, м. Дніпро, просп. Д. Яворницького, 19