

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет інформаційних технологій
Кафедра комп'ютерної інженерії та інформаційних систем



ЗАТВЕРДЖУЮ
Декан ФІТ
Говорущенко Т.О
1
5 вересня 2024р.

СИЛАБУС

Теорія і технології проєктування спеціалізованих операційних систем

Освітньо-наукова програма: Комп'ютерна інженерія та програмування
Рівень вищої освіти: другий (магістерський)

Загальна інформація

Позиція	Зміст інформації
Викладач(і)	Денисюк Дмитро Олександрович
Профайл викладача	http://kiis.khmnmu.edu.ua/personnel/denysyuk-dmytro-oleksandrovych/
E-mail викладача(ів)	denysiuk@khmnmu.edu.ua
Контактний телефон	+380967872767
Сторінка дисципліни в ІСУ	https://msn.khmnmu.edu.ua/course/view.php?id=3663
Навчальний рік	2024-2025
Консультації	Онлайн: за необхідністю та попередньою домовленістю

Характеристика дисципліни

Форма здобуття освіти	Курс	Семестр	Загальний обсяг		Кількість годин						Форма семестрового контролю			
			Кредити ЄКТС	Години	Аудиторні заняття				Індивідуальна робота студента	Самостійна робота, в т.ч. ІРС	Курсовий проєкт	Курсова робота	Залік	Іспит
					Разом	Лекції	Лабораторні роботи	Практичні заняття						
ОД	1	2	5,0	150	54	18	36	0		96				+
Разом ДНФ			5,0	150	54	18	36	0		96				1

Анотація дисципліни

Дисципліна "Теорія і технології проєктування спеціалізованих операційних систем" входить до переліку предметів, які вивчаються у магістрів спеціальності "Комп'ютерна інженерія". Ця дисципліна охоплює такі теми як архітектурні особливості сучасних спеціалізованих обчислювальних систем, їхню розподілену структуру, а також теорію та методи проєктування спеціалізованих операційних систем для цих систем. Вона також розглядає принципи роботи драйверної концепції, що лежить в основі взаємодії операційної системи з різними пристроями, будь то системні або периферійні, реальні чи віртуальні. Курс також включає в себе вивчення теорії та технологій розробки драйверів з використанням цієї концепції, а також розглядає питання проміжного програмного забезпечення для розподілених систем і його роль у загальній системі системного

При викладанні дисципліни використовуються активні і творчі форми проведення занять, зокрема, методи проблемного навчання.

Пререквізити: Іноземна мова за професійним спрямуванням; Теорія алгоритмів та обчислювальних процесів; Технології проектування програмних систем;

кореквізити: Організація та управління бізнес-проектами в галузі інформаційних технологій; Системна інженерія програмного забезпечення комп'ютерних систем; Безпека та захист комп'ютерних систем; Кваліфікаційна робота.

Мета і завдання дисципліни

Метою дисципліни є: 1) ознайомити студентів з поняттям про архітектурні особливості сучасних спеціалізованих обчислювальних систем, зокрема реального часу, Інтернету речей, кіберфізичних, мобільних, робототехнічних, розподілених; 2) сформувати вміння використання спеціалізованих операційних систем в сучасній промисловості при практичному вирішенні задач; 3) надати глибокі знання принципів функціонування спеціалізованих операційних систем і використання теорії проектування спеціалізованих операційних систем для сучасних спеціалізованих обчислювальних систем, зокрема реального часу, Інтернету речей, кіберфізичних, мобільних, робототехнічних, розподілених, а також шляхів вирішення проблемних наукових задач в галузі проектування спеціалізованих операційних систем.

Завдання дисципліни:

- розглянути архітектурні особливості сучасних спеціалізованих обчислювальних систем, зокрема реального часу, Інтернету речей, кіберфізичних, мобільних, робототехнічних, розподілених;
- навчити застосовувати знання основ функціонування сучасних спеціалізованих операційних систем та побудови драйверів при проектуванні сучасних спеціалізованих операційних систем, зокрема реального часу, Інтернету речей, кіберфізичних, мобільних, робототехнічних, розподілених;
- навчити застосовувати знання технічних характеристик та конструктивних особливостей спеціалізованих обчислювальних систем, зокрема реального часу, Інтернету речей, кіберфізичних, мобільних, робототехнічних, розподілених, при розробленні спеціалізованих операційних систем;
- навчити проектувати проміжне програмне забезпечення для розподілених систем;
- навчити проектувати та створювати спеціалізовані операційні системи, зокрема реального часу, Інтернету речей, кіберфізичних, мобільних, робототехнічних, розподілених систем з урахуванням усіх аспектів їх життєвого циклу.

Очікувані результати навчання

Студент, який успішно завершив вивчення дисципліни, повинен: застосовувати загальні підходи пізнання, методи математики, природничих та інженерних наук до розв'язання складних задач комп'ютерної інженерії; застосовувати спеціалізовані концептуальні знання, що включають сучасні наукові здобутки у сфері комп'ютерної інженерії, необхідні для професійної діяльності, оригінального мислення та проведення досліджень, критичного осмислення проблем інформаційних технологій та на межі галузей знань; аналізувати проблематику, ідентифікувати та формулювати конкретні проблеми, що потребують вирішення, обирати ефективні методи їх вирішення; вирішувати задачі аналізу та синтезу комп'ютерних систем та мереж; застосовувати знання технічних характеристик, конструктивних особливостей, призначення і правил експлуатації програмно-технічних засобів комп'ютерних систем та мереж для вирішення складних задач комп'ютерної інженерії та дотичних проблем; розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем; планувати і виконувати наукові дослідження в сфері комп'ютерної інженерії, формулювати і перевіряти гіпотези, обирати методики та інструменти, аналізувати результати, обґрунтовувати висновки; вміти виконувати експериментальні дослідження за професійною тематикою; вміти проектувати та розробляти системне та прикладне програмне забезпечення, включаючи інтерфейси користувача програмних систем, враховуючи його призначення, необхідний рівень якості та зручність використання (usability).

Компетентності, на формування яких спрямовано ОК:

Інтегральна – Здатність розв'язувати складні задачі і проблеми в галузі комп'ютерної інженерії або у процесі навчання, що передбачає проведення досліджень та/або здійснення інновацій та характеризується невизначеністю умов і вимог.

ЗК3. Здатність проводити дослідження на відповідному рівні.

ЗК6. Здатність виявляти, ставити та вирішувати проблеми.

ЗК10. Базові дослідницькі навички і вміння.

СК1. Здатність до визначення технічних характеристик, конструктивних особливостей, застосування і експлуатації програмних, програмно-технічних засобів, комп'ютерних систем та мереж різного призначення.

СК2. Здатність розробляти алгоритмічне та програмне забезпечення, компоненти комп'ютерних систем та мереж, Інтернет додатків, кіберфізичних систем з використанням сучасних методів і мов програмування, а також засобів і систем автоматизації проектування.

СК5. Здатність будувати архітектуру та створювати системне і прикладне програмне забезпечення комп'ютерних систем та мереж.

СК7. Здатність досліджувати, розробляти та обирати технології створення великих і надвеликих систем.

СК10. Здатність ідентифікувати, класифікувати та описувати роботу програмно-технічних засобів, комп'ютерних систем, мереж та їхніх компонентів.

СК11. Здатність обирати ефективні методи розв'язування складних задач комп'ютерної інженерії, критично оцінювати отримані результати та аргументувати прийняті рішення.

СК12. Здатність використовувати методи аналізу, ідентифікації й синтезу комп'ютерних систем та мереж, кіберфізичних систем, засобів Інтернету речей та ІТ-інфраструктур.

Спеціальні компетентності, визначені за освітньою програмою:

СК13. Здатність проектувати системи та їхні компоненти з урахуванням усіх аспектів їх життєвого циклу та поставленої задачі, включаючи створення, налаштування, експлуатацію, технічне обслуговування та утилізацію.

СК14. Здатність проектувати та розробляти інтерфейси користувача програмних систем, враховуючи їх призначення та зручність використання.

СК15. Здатність використовувати методи фундаментальних і прикладних дисциплін комп'ютерної інженерії при проектуванні та розробленні телекомунікаційних систем.

СК17. Здатність досліджувати проблему в галузі комп'ютерних та інформаційних технологій, визначати їх обмеження.

Програмні результати навчання, на забезпечення яких спрямовано ОК:

ПРН1. Застосовувати загальні підходи пізнання, методи математики, природничих та інженерних наук до розв'язання складних задач комп'ютерної інженерії.

ПРН4. Застосовувати спеціалізовані концептуальні знання, що включають сучасні наукові здобутки у сфері комп'ютерної інженерії, необхідні для професійної діяльності, оригінального мислення та проведення досліджень, критичного осмислення проблем інформаційних технологій та на межі галузей знань.

ПРН6. Аналізувати проблематику, ідентифікувати та формулювати конкретні проблеми, що потребують вирішення, обирати ефективні методи їх вирішення.

ПРН7. Вирішувати задачі аналізу та синтезу комп'ютерних систем та мереж.

ПРН8. Застосовувати знання технічних характеристик, конструктивних особливостей, призначення і правил експлуатації програмно-технічних засобів комп'ютерних систем та мереж для вирішення складних задач комп'ютерної інженерії та дотичних проблем.

ПРН9. Розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем.

ПРН14. Планувати і виконувати наукові дослідження в сфері комп'ютерної інженерії, формулювати і перевіряти гіпотези, обирати методики та інструменти, аналізувати результати, обґрунтовувати висновки.

ПРН15. Вміти виконувати експериментальні дослідження за професійною тематикою.

ПРН17. Вміти проектувати та розробляти системне та прикладне програмне забезпечення, включаючи інтерфейси користувача програмних систем, враховуючи його призначення, необхідний рівень якості та зручність використання (usability).

Тематичний і календарний план вивчення дисципліни

№ тижня	Тема лекції*	Тема лабораторної роботи*	Самостійна робота студентів		
			Зміст	Год.	Література
1-2	Вступ. Концептуальні моделі сучасних спеціалізованих операційних систем. Класифікація сучасних	Проектування спеціалізованої RTOS для Інтернету речей на платформі	Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи №1. Самостійне опрацювання теоретичного матеріалу.	11	[1-5]

	<p>спеціалізованих ОС. Місце спеціалізованих операційних систем в загальній класифікації ОС. Дослідження архітектурних особливостей сучасних спеціалізованих обчислювальних систем. Типові складові спеціалізованих ОС. Теоретичні основи операційних систем, зокрема спеціалізованих. Застосування спеціалізованих операційних систем. Апаратно-програмні пристрої в системі задач проектування спеціалізованих ОС. Дослідження проблематики та актуальності проектування спеціалізованих операційних систем. Задачі та проблеми підвищення надійності, безпеки та керування пам'яттю в сучасних спеціалізованих операційних системах. Приклади та шляхи розв'язку наукових задач з теорії проектування спеціалізованих операційних систем. Дослідження методик та інструментів для проектування спеціалізованих операційних систем.</p>	<p>Raspberry Pi. Створення макету проекту. Додавання макросів для виведення налагоджувальних повідомлень. Обгортання глобальних змінних у примітиви синхронізації. Тестування проекту в емуляторі QEMU.</p>	<p>Самостійна робота над розробкою програми до лабораторної роботи №1. Підготовка до захисту лабораторної роботи №1</p>		
3-4	<p>Теорія і технології проектування спеціалізованих операційних систем для керування і обробки даних в режимі реального часу. Моделі систем реального часу. Задачі і технології систем реального часу та методи їх реалізації. Апаратно-програмні пристрої. Інтерфейси апаратно-програмних пристроїв. Моделі і стандарти</p>	<p>Проектування спеціалізованої RTOS для Інтернету речей на платформі Raspberry Pi. Реалізація базових функцій ядра. Додавання до проекту драйверів GPIO та UART та перенесення його на цільову платформу. Реалізація базових функцій ядра RTOS.</p>	<p>Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи №2. Самостійне опрацювання теоретичного матеріалу. Самостійна робота над розробкою програми до лабораторної роботи №2. Підготовка до захисту лабораторної роботи №2</p>	11	[1, 6, 7]

	інтерфейсів.	Додавання функцій апаратного таймера для усунення циклічних затримок в драйвері GPIO. Тестування на реальній платформі.			
5-6	Концепції та технології проєктування ОС Інтернету речей, ОС кіберфізичних систем, мобільних ОС. Моделі систем Інтернету речей, кіберфізичних систем, мобільних систем та спеціалізованих ОС як їх складових в процесі синтезу систем. Методи проєктування спеціалізованих операційних систем для Інтернету речей. Дослідження проблематики гетерогенності та керування ресурсами в операційних системах Інтернету речей. Методи проєктування спеціалізованих операційних систем для мобільних систем. Методи проєктування спеціалізованих операційних систем для кіберфізичних систем.	Проєктування спеціалізованої RTOS для Інтернету речей на платформі Raspberry Pi. Реалізація рівнів привілеїв. Реалізація функції завантаження ядра по UART. Апаратне налагодження по JTAG. Реалізація рівнів привілеїв. Тестування на реальній платформі.	Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи №3. Самостійне опрацювання теоретичного матеріалу. Самостійна робота над розробкою програми до лабораторної роботи №3. Підготовка до захисту лабораторної роботи №3	11	[8-13]
7-8	Керування виконавчими пристроями в режимі реального часу по інтерфейсу USB. Моделі передачі даних.	Проєктування спеціалізованої RTOS для Інтернету речей на платформі Raspberry Pi. Реалізація функцій обробки виняткових ситуацій. Розмежування адресного простору ядра і користувача. Реалізація функцій обробки виняткових ситуацій. Комплексне тестування на реальній платформі.	Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи №4. Самостійне опрацювання теоретичного матеріалу. Самостійна робота над розробкою програми до лабораторної роботи №4. Підготовка до захисту лабораторної роботи №4	11	[1]
9-10	Робототехнічні системи та спеціалізовані ОС для них. Моделі робототехнічних систем. Технології штучного	Проєктування вбудованої RTOS для платформи Lego EV3 на базі ОС Linux.	Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи №5. Самостійне опрацювання теоретичного матеріалу.	11	[1, 14, 15]

	інтелекту в робототехнічних системах. Дослідження стратегій розвитку робототехнічних операційних систем на прикладі сучасних наукових задач.	Модифікація базового ядра Linux для забезпечення функцій RTOS. Дослідження роботи RTOS на платформі Lego EV3.	Самостійна робота над розробкою програми до лабораторної роботи №5. Підготовка до захисту лабораторної роботи №5		
11-12	Драйверна концепція. Теорія і технології проєктування драйверів для ОС Windows. Типи драйверів. Драйвери режиму ядра та режиму користувача. Віртуальна модель пристрою та його інтерфейсу. Human Interface Devices, HID. Network Driver Interface Specification, NDIS. Драйвери протоколів.	Розроблення драйверів пристроїв в ОС Windows. Kernel-Mode Driver Framework. User-Mode Driver Framework**. Створення, розгортання, тестування та налагодження драйверів периферійних та віртуальних пристроїв.	Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи №6. Самостійне опрацювання теоретичного матеріалу. Самостійна робота над розробкою програми до лабораторної роботи №6. Підготовка до захисту лабораторної роботи №6	11	[16]
13-14	Теорія і технології проєктування драйверів для ОС Linux. Технології проєктування драйверів для ОС Linux. Компіляція драйверів з ядром. Реалізація драйверів як модулів ядра.	Розроблення драйверів пристроїв як модулів ядра для ОС Linux**. Створення драйвера як модуля ядра. Збирання ядра разом з модулем драйвера.	Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи №7. Самостійне опрацювання теоретичного матеріалу. Самостійна робота над розробкою програми до лабораторної роботи №7. Підготовка до захисту лабораторної роботи №7	11	[17-20]
15-16	Проміжне програмне забезпечення для розподілених систем та його місце в загальній системі системного програмного забезпечення. Принципи створення розподілених систем. Теорія розподілених обчислень. Проблемні наукові задачі в системі здійснення розподілених обчислень. Проміжне програмне забезпечення для розподілених систем: принципи, засоби створення, зв'язок з операційними системами.	Реалізація блокчейн технологій. Побудова блокчейну для нефінансових операцій.	Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи №8. Самостійне опрацювання теоретичного матеріалу. Самостійна робота над розробкою програми до лабораторної роботи №8. Підготовка до захисту лабораторної роботи №8	11	[21-23]
17	Концепція блокчейн технології. Принципи побудови блокчейн-мереж. Життєвий цикл транзакції. Надійність та моделі розгортання	<i>Підсумкове заняття</i>	Підготовка до іспиту	8	[24-27]

блокчейну. Паралельні обчислення. Проблемні наукові задачі в системі реалізації паралельних обчислень. Балансування навантаження серверів Linux. HAProxy, Keepalived: ПЗ для забезпечення відмовостійкості мережних сервісів та балансування навантаження. Nginx.				
---	--	--	--	--

Примітки: *Лекції, лабораторні заняття проводяться по дві години; послідовність проведення занять визначається розкладом (може не відповідати нумерованим тижням)

** Лабораторна робота може бути зарахована за наявності сертифікатів з проходження курсів (Udemy, Linux Kernel Driver Development, <https://www.udemy.com/course/linux-kernel-driver-development/> та ін.)

Політика дисципліни

Організація освітнього процесу з дисципліни відповідає вимогам положень про організаційне і навчально-методичне забезпечення освітнього процесу, освітній програмі та навчальному плану. Студент зобов'язаний відвідувати лекції та лабораторні заняття згідно з розкладом, не запізнюватися на заняття, завдання виконувати відповідно до графіка. Пропущене лабораторне заняття студент зобов'язаний опрацювати самостійно у повному обсязі і відзвітувати перед викладачем не пізніше, ніж за тиждень до чергової атестації. До лабораторних занять студент має підготуватися за відповідною темою і проявляти активність. Набуті особою знання з дисципліни або її окремих розділів у неформальній освіті зараховуються відповідно до Положення про порядок перерахування результатів навчання у ХНУ.

Критерії оцінювання результатів навчання

Кожний вид роботи з дисципліни оцінюється за **чотирибальною** шкалою. Семестрова підсумкова оцінка визначається як середньозважена з усіх видів навчальної роботи, виконаних і зданих позитивно з врахуванням коефіцієнта вагомості. Вагові коефіцієнти змінюються залежно від структури дисципліни і важливості окремих її видів робіт. При оцінюванні знань студентів використовуються різні засоби контролю, зокрема: усне опитування перед допуском до виконання лабораторної роботи – здійснюється на її початку; якість виконання, набуття теоретичних знань і практичних навичок перевіряється шляхом захисту кожної лабораторної роботи згідно з робочою програмою дисципліни і робочим навчальним планом.

Структурування дисципліни за видами робіт і оцінювання результатів навчання студентів денної форми навчання у семестрі за ваговими коефіцієнтами

Аудиторна робота								Тестовий контроль	Форма підсумкового контролю
Лабораторні роботи №:									
1	2	3	4	5	6	7	8	1	
ВК: 0,5								0,1	0,4

Примітка: ВК – ваговий коефіцієнт.

Співвідношення інституційної шкали оцінювання і шкали оцінювання ЄКТС

Оцінка ЄКТС	Інституційна шкала балів	Інституційна оцінка	Зараховані	0	Критерії оцінювання	
A	4,75-5,00	5				Відмінно – глибоке і повне опанування навчального матеріалу і виявлення відповідних умінь та навичок.
B	4,25-4,74	4				Добре – повне знання навчального матеріалу з кількома незначними помилками.
C	3,75-4,24	4				Добре – в загальному правильна відповідь з двома-трьома

				суттєвими помилками.
D	3,25-3,74	3		Задовільно – неповне опанування програмного матеріалу, але достатнє для практичної діяльності за професією.
E	3.00-3,24	3		Задовільно – неповне опанування програмного матеріалу, що задовольняє мінімальні критерії оцінювання
FX	2,00-2,99	2	Незараховано	Незадовільно – безсистемність одержаних знань і неможливість продовжити навчання без додаткових знань з дисципліни
F	0,00-1,99	2		Незадовільно – необхідна серйозна подальша робота і повторне вивчення дисципліни.

Питання для підсумкового контролю з дисципліни

1. Класифікація сучасних спеціалізованих ОС.
2. Місце спеціалізованих операційних систем в загальній класифікації ОС.
3. Дослідження архітектурних особливостей сучасних спеціалізованих обчислювальних систем.
4. Типові складові спеціалізованих ОС.
5. Теоретичні основи операційних систем, зокрема спеціалізованих.
6. Застосування спеціалізованих операційних систем.
7. Апаратно-програмні пристрої в системі задач проектування спеціалізованих ОС.
8. Дослідження проблематики та актуальності проектування спеціалізованих операційних систем.
9. Задачі та проблеми підвищення надійності, безпеки та керування пам'яттю в сучасних спеціалізованих операційних системах.
10. Приклади та шляхи розв'язку наукових задач з теорії проектування спеціалізованих операційних систем.
11. Дослідження методик та інструментів для проектування спеціалізованих операційних систем.
12. Моделі систем реального часу.
13. Задачі і технології систем реального часу та методи їх реалізації.
14. Апаратно-програмні пристрої.
15. Інтерфейси апаратно-програмних пристроїв.
16. Моделі і стандарти інтерфейсів апаратно-програмних пристроїв.
17. Моделі систем Інтернету речей.
18. Моделі кіберфізичних систем.
19. Моделі мобільних систем.
20. Моделі спеціалізованих ОС як складових в процесі синтезу систем.
21. Методи проектування спеціалізованих операційних систем для Інтернету речей.
22. Дослідження проблематики гетерогенності та керування ресурсами в операційних системах Інтернету речей.
23. Методи проектування спеціалізованих операційних систем для мобільних систем.
24. Методи проектування спеціалізованих операційних систем для кіберфізичних систем.
25. Моделі передачі даних.
26. Моделі робототехнічних систем.
27. Технології штучного інтелекту в робототехнічних системах.
28. Стратегії розвитку робототехнічних операційних систем на прикладі сучасних наукових задач.
29. Типи драйверів.
30. Драйвери режиму ядра та режиму користувача.
31. Віртуальна модель пристрою та його інтерфейсу.
32. Human Interface Devices, HID.
33. Network Driver Interface Specification, NDIS.
34. Драйвери протоколів.
35. Потік даних в KMDF / UMDF.
36. Керування живленням і модель вводу / виводу в WDF.
37. Setup application programming interface (SetupAPI).
38. Особливості моделі введення / виведення UMDF (KMDF).
39. Об'єктна модель WDF.

40. Об'єктна модель KMDF.
41. Об'єкти KMDF. Життєвий цикл об'єкта KMDF.
42. Об'єкти UMDF. Життєвий цикл UMDF-драйвера.
43. Windows Driver Foundation (WDF).
44. Структура драйвера.
45. Основні функції для побудови драйвера.
46. Особливості вибору моделі драйвера для ОС Windows.
47. Технології проектування драйверів для ОС Linux.
48. Компіляція драйверів з ядром.
49. Реалізація драйверів як модулів ядра.
50. Принципи створення розподілених систем.
51. Теорія розподілених обчислень.
52. Проблемні наукові задачі в системі здійснення розподілених обчислень.
53. Проміжне програмне забезпечення для розподілених систем: принципи, засоби створення, зв'язок з операційними системами.
54. Принципи побудови блокчейн-мереж.
55. Життєвий цикл транзакції.
56. Надійність та моделі розгортання блокчейну.
57. Паралельні обчислення.
58. Проблемні наукові задачі в системі реалізації паралельних обчислень.
59. Балансування навантаження серверів Linux.
60. HAProxy.
61. Keepalived.
62. Nginx.

МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ

Бобровнікова К.Ю. Теорія і технології проектування спеціалізованих операційних систем. Методичні вказівки для виконання лабораторних робіт для здобувачів вищої освіти за освітньо-науковою програмою «Комп'ютерна інженерія та програмування» спеціальності 123 «Комп'ютерна інженерія» другого (магістерського) рівня вищої освіти. Хмельницький:ХНУ, 2021. – 99 с.

РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. Tanenbaum, Andrew S.; Woodhull, Albert S. Operating systems: design and implementation. Englewood Cliffs: Prentice Hall, 2017, 1120 p.
2. Marufuzzaman, M., Al Karim, S., Rahman, M. S., Zahid, N. M., & Sidek, L. M. A review on reliability, security and memory management of numerous operating systems. Indonesian Journal of Electrical Engineering and Informatics (IJEI), 7(3), 2019, pp. 577-585.
3. Szopa, A. Comparison Analysis of Accessible Features Built into Operating Systems. In International Conference on Applied Human Factors and Ergonomics – Springer, Cham, 2021.
4. Silberschatz, A., Galvin, P.B., Gagne, G. Operating system concepts essentials. John Wiley & Sons, Inc., 2019, 784 p.
5. Malallah, H., Zeebaree, S. R., Zebari, R. R., Sadeeq, M. A., Ageed, Z. S., Ibrahim, I. M., ... & Merceedi, K. J. A comprehensive study of kernel (issues and concepts) in different operating systems. Asian Journal of Research in Computer Science, 2021, pp. 16-31.
6. Erciyas, K. Distributed Real-Time Operating Systems and the Middleware. In Distributed Real-Time Systems. Springer, Cham, 2019, pp. 119-148.
7. Serino, A., & Cheng, L. Real-Time Operating Systems for Cyber-Physical Systems: Current Status and Future Research. In 2020 International Conferences on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData) and IEEE Congress on Cybermatics (Cybermatics), 2020, pp. 419-425. IEEE.
8. Agarwal, V. K., Kumar, N., Singh, R., Pathak, A., Bhushan, B. Operating Systems Support and Network Optimization View of Internet of Things. In Emerging Technologies in Data Mining and Information Security – Springer, Singapore, 2021.
9. Bansal, S., & Kumar, D. IoT ecosystem: A survey on devices, gateways, operating systems, middleware and communication. International Journal of Wireless Information Networks, Springer, Cham, 2020, pp. 1-25.
10. Javed, F., Afzal, M. K., Sharif, M., & Kim, B. S. Internet of Things (IoT) operating systems support, networking technologies, applications, and challenges: A comparative review. IEEE Communications


Tutorials, 20(3), 2018, pp. 2062-2100.

11. Musaddiq, A., Zikria, Y. B., Hahm, O., Yu, H., Bashir, A. K., & Kim, S. W. A survey on resource management in IoT operating systems. IEEE Access, 6, 2018, pp. 8459-8482.
12. Rivnak, M., Deligiannidis, L. Real-Time Operating Systems: Course Development. In Advances in Security, Networks, and Internet of Things - Springer, Cham, 2021.
13. Korchi, A., Khachouch, M. K., Lakhri, Y., & Moumen, A. Classification of existing mobile cross-platform approaches. In 2020 International Conference on Electrical, Communication, and Computer Engineering (ICECCE) IEEE, 2020, pp. 1-5.
14. Dawarka, V., & Bekaroo, G. Building and evaluating cloud robotic systems: A systematic review. Robotics and Computer-Integrated Manufacturing, 73, 2022, 102240.
15. Wang, K. C. Embedded real-time operating systems. In: Embedded and Real-Time Operating Systems. Springer, Cham, 2017. p. 401-475.
16. Windows Driver Kit documentation [Електронний ресурс] : [Інтернет-портал]. - Електронні дані. - [© Microsoft. 2018]. - Режим доступу: <https://docs.microsoft.com/en-us/windows-hardware/drivers/> (дата звернення 22.09.2018). Назва з скана.
17. Love, R. The Linux kernel: a description of the development process. Robert Love, 2019, 496 p.
18. Granneman, Scott. Linux phrasebook. Sams Publishing, 2019, 416 p.
19. Simmonds, Chris. Mastering Embedded Linux Programming. Packt Publishing Ltd, 2017, 360 p.
20. Cooling, Jim. Real-time Operating Systems Book I: The Foundations (The engineering of real-time embedded systems). Independently published, 2018, 296 p.
21. Balador, A., Ericsson, N., Bakhshi, Z. Communication middleware technologies for industrial distributed control systems: A literature review. In 2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), 2017, IEEE, pp. 1-6.
22. Gascon-Samson, J., Rafiuzzaman, M., Pattabiraman, K. Thingsjs: Towards a flexible and self-adaptable middleware for dynamic and heterogeneous iot environments. In Proceedings of the 4th Workshop on Middleware and Applications for the Internet of Things, 2017, pp. 11-16.
23. Mohamed, N., Al-Jaroodi, J., Lazarova-Molnar, S., Jawhar, I. Middleware challenges for cyber-physical systems. Scalable Computing: Practice and Experience, 18(4), 2017, pp. 331-346.
24. Shukur, H., Zeebaree, S. R., Ahmed, A. J., Zebari, R. R., Ahmed, O., Tahir, B. S. A., Sadeeq, M. A. A state of art survey for concurrent computation and clustering of parallel computing for distributed systems. Journal of Applied Science and Technology Trends, 1(4), 2020, pp. 148-154.
25. LU, Yang. The blockchain: State-of-the-art and research challenges. Journal of Industrial Information Integration, 2019 15, pp. 80-90.
26. Trobec, R., Slivnik, B., Bulic, P., Robic, B. Introduction to Parallel Computing: From Algorithms to Programming on State-of-Art Platforms, 2018, Springer.
27. Erciyes, K. (2019). Distributed real-time systems. In Distributed Real-Time Systems (pp. 41-62). Springer, Cham.
28. Модульне середовище Для навчання MOODLE. Доступ до ресурсу: <https://msn.khnu.km.ua/course/view.php?id=3663>.
29. Електронна бібліотека університету. Доступ до ресурсу: http://lib.khnu.km.ua/asp/php_f/plage_lib.php

Зав. каф. КПС:

Гарант ОПП «КІП»:

Розробник


Підпис

Денисюк Д.О.
Ініціали, прізвище

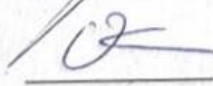
Погоджено

Зав. каф. КПС


Підпис

Засорнова І.О.
Ініціали, прізвище

Гарант ООП «КІП»


Підпис

Савенко О.С.