

# НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ВОДНОГО ГОСПОДАРСТВА ТА ПРИРОДОКОРИСТУВАННЯ

Навчально-науковий інститут кібернетики, інформаційних технологій та інженерії

04-01-157S

## СИЛАБУС

*навчальної дисципліни*

## SYLLABUS

Архітектура обчислювальних систем та системне програмування		Computer system architecture and system programming
Шифр за ОП	<b>ОК 26</b>	Code in Degree Programme
Освітній рівень: Бакалаврський (перший)		Level of Education: Bachelor's (first)
Галузь знань <b>Інформаційні технології</b>	<b>12</b>	Field of Knowledge <b>Information Technology</b>
Спеціальність <b>Комп'ютерні науки</b>	<b>122</b>	Field of Study <b>Computer science</b>
Освітня програма: <b>Комп'ютерні науки</b>		Degree Programme: <b>Computer science</b>

РІВНЕ – 2024

Силабус навчальної дисципліни «Архітектура обчислювальних систем та системне програмування» для здобувачів вищої освіти ступеня «бакалавр», які навчаються за освітньо-професійною програмою «Комп'ютерні науки» спеціальності 122 «Комп'ютерні науки». Рівне. НУВГП. 2024. 12 стор.

ОП на сайті університету: <https://ep3.nuwm.edu.ua/23461/>.

Розробник силабусу: *Герус Володимир Андрійович, старший викладач кафедри Комп'ютерних наук та прикладної математики*

Силабус схвалений на засіданні кафедри комп'ютерних наук та прикладної математики  
Протокол №9 засідання науково-методичної ради з якості ННІ КІТІ  
НУВГП від 30 серпня 2024 року

Завідувач кафедри: Турбал Юрій Васильович, доктор технічних наук, професор

Керівник (гарант) освітньої програми: Каштан Сергій Степанович, кандидат технічних наук, доцент, доцент кафедри комп'ютерних наук та прикладної математики

Схвалено науково-методичною радою з якості ННІ кібернетики, інформаційних технологій та інженерії  
Протокол №9 від 30 серпня 2024 року

Голова науково-методичної ради з якості ННІ: Мартинюк Петро Миколайович, доктор технічних наук, професор, директор ННІ кібернетики, інформаційних технологій та інженерії.

© НУВГП, 2024

<b>ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ</b>	
<b>Архітектура обчислювальних систем та системне програмування</b>	
<b>ЗАГАЛЬНА ІНФОРМАЦІЯ</b>	
Ступінь вищої освіти	<i>Бакалавр</i>
Освітня програма	<i>Комп'ютерні науки</i>
Спеціальність	<i>122 Комп'ютерні науки</i>
Рік навчання, семестр	<i>2-й рік навчання, 3-й семестр</i>
Кількість кредитів	<i>3,5</i>
Лекції:	<i>24 годин</i>
Лабораторні заняття:	<i>24 годин</i>
Самостійна робота:	<i>57 годин</i>
Курсова робота:	<i>Ні</i>
Форма навчання	<i>денна, заочна</i>
Форма підсумкового контролю	<i>Екзамен</i>
Мова викладання	<i>державна</i>
<b>ІНФОРМАЦІЯ ПРО РОЗРОБНИКА</b>	

Лектор



Герус Володимир Андрійович, старший викладач кафедри «комп'ютерних наук та прикладної математики», сумісник кафедри «Обчислювальної техніки»

Вікіситет

[http://wiki.nuwm.edu.ua/index.php/Герус Володимир Андрійович](http://wiki.nuwm.edu.ua/index.php/Герус_Володимир_Андрійович)

ORCID

<http://orcid.org/0000-0003-4388-7425>

Канали комунікації

[v.a.gerus@nuwm.edu.ua](mailto:v.a.gerus@nuwm.edu.ua)

## ІНФОРМАЦІЯ ПРО НАВЧАЛЬНУ ДИСЦИПЛІНУ

### Мета та завдання

Метою дисципліни "Архітектура обчислювальних систем та системне програмування" є формування у здобувачів освіти теоретичних знань і практичних навичок з архітектури сучасних обчислювальних систем, принципів їх функціонування, організації та взаємодії апаратного й програмного забезпечення, а також оволодіння методами розробки, налагодження та оптимізації системного програмного забезпечення.

Завдання дисципліни "Архітектура обчислювальних систем та системне програмування":

1. Ознайомлення з принципами побудови та функціонування сучасних обчислювальних систем.

2. Вивчення архітектурних рішень центральних процесорів, пам'яті, шинних інтерфейсів та інших компонентів комп'ютерних систем.

3. Дослідження операційних систем, їх структур та взаємодії з апаратним забезпеченням.

4. Вивчення основ системного програмування, роботи з низькорівневими мовами та засобами розробки системного ПЗ.

5. Формування навичок розробки, тестування та налагодження програм, що працюють на рівні операційної системи, застосування технологій для розробки ефективного системного програмного забезпечення.

**Посилання на розміщення освітнього компонента на навчальній платформі Moodle, на платформі освітніх програм та їхніх освітніх компонентів**

### Передумови вивчення

(місце освітнього компоненту в структурно-логічній схемі)

Для опанування даного ОК здобувачам необхідні знання із таких ОК: ОК14. Математична логіка та логічне програмування, ОК22. Програмування, ОК28 Об'єктно-орієнтоване програмування, ОК12 Комп'ютерна дискретна математика, ОК21 Практикум з програмування.

### Компетентності

Дисципліна підсилює набуття наступних компетентностей.

ЗК2. Здатність застосовувати знання у практичних ситуаціях, ЗК3. Знання та розуміння предметної області та розуміння професійної діяльності, ЗК4. Здатність спілкуватися державною мовою як усно, так і письмово, ЗК11. Здатність приймати обґрунтовані рішення. ФК8. Здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління, ФК12. Здатність забезпечити організацію обчислювальних процесів в інформаційних системах різного призначення з урахуванням архітектури, конфігурування, показників результативності функціонування операційних систем і системного програмного забезпечення.

### Програмні результати навчання (ПРН). Результати навчання (РН)\*

ПРН13. Володіти мовами системного програмування та методами розробки програм, що взаємодіють з компонентами комп'ютерних систем, знати мережні технології, архітектури комп'ютерних мереж, мати практичні навички технології адміністрування комп'ютерних мереж та їх програмного забезпечення

### Структура та зміст навчальної дисципліни

№	Теми (лекції)	Опис лекції	№	Теми лабораторних занять
---	---------------	-------------	---	--------------------------

#### МОДУЛЬ 1. Сучасна архітектура та компоненти обчислювальних систем

1	<b>Вступ до архітектури обчислювальних систем (2 год.)</b>	Основні поняття та класифікація обчислювальних систем. Історія розвитку та перспективи архітектури комп'ютерів.	1	Аналіз архітектури обчислювальних систем, визначення основних характеристик процесора та пам'яті. Дослідження конвеєрної обробки команд (2 год.)
2	<b>Обчислювальні системи (2 год.)</b>	Функціональна організація апаратного забезпечення.	2	Дослідження конвеєрної обробки команд (2 год.)
3	<b>Процесори у сучасних обчислювальних системах (2 год.)</b>	Архітектура процесорів: будова, основні блоки, принципи роботи..	3	Дослідження організації пам'яті (2 год.)
4	<b>Пам'ять у сучасних обчислювальних системах (2 год.)</b>	Організація пам'яті: оперативна пам'ять, кеш-пам'ять, віртуальна пам'ять.	4	Взаємодія процесів в операційній системі (4 год.)
5	<b>Сучасні обчислювальні системи (2 год.)</b>	Паралельна та конвеєрна обробка даних.	5	Вивчення механізмів введення-виведення (2 год.)
<b>МОДУЛЬ 2. Системне програмне забезпечення та основи системного програмування</b>				
6	<b>Системне програмне забезпечення (2 год.)</b>	Ядро та файлові системи.	6	Оптимізація коду та використання апаратних ресурсів (4 год.)
7	<b>Операційні системи (2 год.)</b>	Основи роботи операційних систем та планування процесів.	7	Використання API ОС для системного програмування. (4 год.)
8	<b>Аналізу процесів у операційній системі (2 год.)</b>	Створення скриптів для адміністрування.	8	Основи програмування на Assembler (4 год.)
9	<b>Основи системного програмування (6 год.)</b>	Мови системного програмування: Assembler, C, C++. Робота з API операційної системи. Управління пам'яттю, міжпроцесна взаємодія, робота з файлами на рівні ОС.		

10	Тенденції розвитку обчислювальних систем та системного програмування (2 год.)	Сучасні тенденції: багатоядерні процесори, ARM-архітектура, квантові обчислення		
----	---	---	--	--

### Форми та методи навчання

*Методи навчання: демонстрація, навчальна дискусія, дебати; технології викладання: тренінги, аналіз конкретних ситуацій, обговорення, мультимедійні презентації, міні-лекції, ситуаційні дослідження, навчання на основі досвіду та інші.*

### Інструменти, обладнання, програмне забезпечення

*Персональні комп'ютери, одноплатні комп'ютери (Raspberry Pi, Arduino для практичного вивчення системної архітектури), стенди для дослідження мікропроцесорної архітектури та взаємодії з периферійними пристроями, програмовані мікроконтролери для вивчення системного програмування на вбудованих платформах, жорсткі диски, SSD, флеш-накопичувачі для аналізу роботи файлових систем, роутери та комутатори для вивчення взаємодії між обчислювальними системами.*

*Операційні системи: Windows 10/11, Linux (Ubuntu, Debian, CentOS) – для тестування різних архітектур ОС. VirtualBox, VMware Workstation, – для емуляції різних апаратних середовищ.*

### Порядок оцінювання програмних результатів навчання/ результатів навчання

- Сума балів = 100:
- 60 – поточна робота;
- 40 – модульний контроль;
- Розподіл балів:
- Модульні контрольні роботи: 40 балів
- 1-й модульний контроль 20 балів, 8 тиждень, (Рівень1 -18 питань 0.5 бали за питання, Рівень 2 -10 питань 0.6 бали за питання, Рівень 3 5 питань 1 бал за питання)
- 2-й модульний контроль 20 балів, 15 тиждень; Рівень1 -18 питань 0.5 бали за питання, Рівень 2 -10 питань 0.6 бали за питання, Рівень 3 5 питань 1 бал за питання)
- Лабораторні роботи: 60 балів, 6 балів за лабораторну роботу: 3 бали – виконання лабораторної роботи; 2 бала – усний захист лабораторної роботи; 1 бал – робота на занятті та вчасно зданий звіт.
- Додаткові бали (участь олімпіадах, конференціях, написання статей, активна студентська діяльність тощо) 5-10 балів.

Основні критерії, що характеризують рівень компетентності здобувача вищої освіти при оцінюванні результатів поточного та підсумкового контролів з навчальної дисципліни:

- виконання всіх видів навчальної роботи, що передбачені силабусом
- навчальної дисципліни;
- глибина і характер знань навчального матеріалу за змістом навчальної
- дисципліни, що міститься в основних та додаткових рекомендованих
- літературних джерелах;
- вміння аналізувати явища, що вивчаються, у їх взаємозв'язку і розвитку;
- характер відповідей на поставлені питання (чіткість, лаконічність, логічність, послідовність тощо);
- вміння застосовувати теоретичні положення під час розв'язання практичних задач;
- вміння аналізувати достовірність одержаних результатів;
- своєчасність виконання;
- дотримання вимог до оформлення (конструкторської та технологічної
- документації, ДСТУ тощо).

Критерії оцінювання практичних завдань, задач, лабораторних робіт (у % від кількості балів, виділених на завдання із заокругленням до цілого числа):

0% – завдання не виконано;

40% – завдання виконано частково та містить суттєві помилки методичного або розрахункового характеру, порушені терміни виконання та вимоги до оформлення;

60% – завдання виконано повністю, але містить суттєві помилки у розрахунках або в методиці, порушені терміни виконання та вимоги до оформлення;

80% – завдання виконано повністю і вчасно, проте містить окремі несуттєві недоліки (розмірності, висновки, оформлення тощо);

100% – завдання виконано правильно, вчасно і без зауважень.

**Рекомендована література (основна, допоміжна)**

### Основна література:

1. Паттерсон Д., Хеннесі Дж. Архітектура комп'ютера: кількісний підхід. – Київ: Діалектика, 2005. – 920 с.
2. Таненбаум Е. Сучасна архітектура комп'ютерів. – Київ: Вільямс, 2006. – 816 с.
3. Брайант Р., О'Халларон Д. Комп'ютерні системи: структура та програмування. – Київ: Вільямс, 2004. – 992 с.
4. Сосин В.В. Архітектура комп'ютерів. – Київ: Каравела, 2010. – 368 с..
5. Булгакова О.С., Зосімов В.В., Поздєєв В.О. Методи та системи штучного інтелекту: теорія та практика: навч. посіб. – Херсон: Олді-плюс, 2020. – 356 с..
6. Закладний О.М., Матвієнко М.П., Розен В.П. Архітектура комп'ютера. – Київ: Ліра К, 2016. – 264 с.
7. Крупельницький Л.В. Архітектура комп'ютерів. Частина 1. – Вінниця: ВНТУ, 2020. – 104 с.

### Допоміжна література:

1. Бушма А. Архітектура обчислювальних систем: Робоча програма навчальної дисципліни. – Київ: Київський університет імені Бориса Грінченка, 2022. – 45 с.
2. Рикалюк Р.Є. Архітектура комп'ютерів: Текст лекцій. – Львів: ЛНУ ім. І. Франка, 2002. – 120 с.
3. Злобін Г.Г., Рикалюк Р.Є. Архітектура та апаратне забезпечення ПЕОМ: Навчальний посібник. – Київ: Каравела, 2006. – 304 с.
4. Савченко В.В., Поляков О.М. Архітектура і технології мобільних мереж LTE і 5G. – Дніпро: Ліра, 2022. – 298 с.
5. Артюхов В.Г. та ін. Архітектура обчислювальних систем: Методичні рекомендації до виконання лабораторних робіт. – Харків: ХНУРЕ, 2015. – 64 с.

**Інформаційні ресурси в Інтернеті**



. IEEE Xplore Digital Library URL: <https://ieeexplore.ieee.org>  
Ресурс містить наукові статті та матеріали з різних галузей комп'ютерних наук, включаючи архітектуру обчислювальних систем, мікропроцесори та системне програмування.

. ACM Digital Library URL: <https://dl.acm.org>  
Ресурс для досліджень у галузі комп'ютерних наук. Він надає доступ до технічних статей, журналів та конференцій з архітектури комп'ютерних систем, програмування та обчислювальних мереж.

. ArXiv URL: <https://arxiv.org>  
Ресурс з відкритим доступом, який пропонує безкоштовний доступ до наукових статей з різних областей науки, в тому числі архітектури обчислювальних систем та програмування.

. Stack Overflow URL: <https://stackoverflow.com>  
Платформа для запитань та відповідей, де можна знайти багато обговорень з архітектури комп'ютерних систем та системного програмування, а також отримати допомогу від досвідчених програмістів.

. GitHub URL: <https://github.com>  
Ресурс для хостингу програмного забезпечення та спільної розробки. Можна знайти проекти з системного програмування та архітектури комп'ютерів, а також брати участь у відкритих розробках.

. Coursera URL: <https://www.coursera.org>  
Платформа онлайн-курсів, де доступні курси з архітектури комп'ютерних систем та програмування, у тому числі від провідних університетів і компаній.

. edX URL: <https://www.edx.org>  
Онлайн-платформа для вивчення дисциплін в різних галузях, у тому числі комп'ютерних наук. Можна знайти курси на тему архітектури обчислювальних систем та операційних систем.

. Linux Documentation Project URL: <https://www.tldp.org>  
Ресурс з документацією для операційної системи Linux, корисний для розуміння принципів роботи системного програмування на відкритих платформах.

. W3Schools URL: <https://www.w3schools.com>  
Ресурс для вивчення основ програмування, в тому числі мов, таких як C/C++ та Python, які використовуються у системному програмуванні.

## **ПОЛІТИКИ ВИКЛАДАННЯ ТА НАВЧАННЯ**

### **Перелік соціальних, «м'яких» навичок (soft skills)**

*Критичне мислення – здатність аналізувати вимоги та визначати оптимальні шляхи реалізації проєктів, знаходити причини проблем та підходи до їх вирішення.*

*Креативність та інноваційне мислення – допомагає знаходити нестандартні рішення та створювати зручний і цікавий користувацький досвід.*

*Комунікаційні навички – вміння ефективно спілкуватися з членами команди, клієнтами та користувачами, пояснювати технічні деталі доступною мовою.*

*Управління часом – здатність планувати роботу, правильно розподіляти час між задачами і дотримуватись дедлайнів.*

*Адаптивність – вміння швидко підлаштовуватися під зміни в технологіях, вимогах проєкту чи методах розробки.*

*Робота в команді – здатність взаємодіяти з іншими спеціалістами (дизайнерами, тестувальниками, менеджерами), ділитися знаннями та спільно працювати над досягненням мети.*

*Увага до деталей – допомагає уникнути помилок у кодї, які можуть вплинути на функціональність додатку або користувацький досвід.*

*Стресостійкість – здатність справлятися з високими навантаженнями, залишатися продуктивним під тиском та в умовах стиснених термінів.*

*Емоційний інтелект (EQ) – розуміння та управління власними емоціями та розуміння емоцій інших, що допомагає налагоджувати ефективну співпрацю та вирішувати конфлікти.*

*Вирішення проблем – вміння швидко знаходити рішення для різних технічних та організаційних проблем, зокрема під час розробки, тестування або розгортання додатків.*

*Самоорганізація та ініціативність – здатність працювати самостійно, знаходити шляхи покращення проєкту та пропонувати нові ідеї без зовнішнього контролю.*

*Постійне самовдосконалення – готовність навчатися новим технологіям, покращувати свої знання та навички, щоб залишатися конкурентоспроможним у швидко мінливій галузі.*

## **Дедлайни та перескладання**

Оформлення та захист звіту виконується протягом двох неділь: Захист звіту на поточному занятті на 2-й тиждень може бути оціненим максимальною кількістю балів, на 3-й тиждень на один бал менше, На здачу кожного з модулів студенту надається одна спроба. У разі якщо здобувач вчасно не встиг здати та захистити звіт, він має право здати та захистити його на консультації. Прийом звітів та їх поточне оцінювання завершується на останньому підсумковому занятті. Якщо студент набрав достатню кількість балів він може претендувати на зарахування поточних балів як підсумкового контролю. Якщо балів не достатньо то студент має можливість звернутися до викладача для перездачі модульних контролів у вигляді підсумкового заліку, який проводить центр незалежного оцінювання.

### **Неформальна та інформальна освіта (за потреби)**

Студенти мають право на часткове або повне перезарахування предмету за умови написання ними відповідної заяви та надання документів, які підтверджують ті результати навчання, які здобувач отримав (див. положення <https://ep3.nuwm.edu.ua/18660/>). Зокрема студенти можуть самостійно проходити онлайн-курси на таких навчальних платформах, як Prometheus, Coursera, edEx, edEra, FutureLearn та інших, для наступного перезарахування результатів навчання.

### **Правила академічної доброчесності**

За списування під час проведення модульного контролю чи підсумкового контролю, студент позбавляється подальшого права здавати матеріал і у нього виникає академічна заборгованість. Здобувачі не допускаються до списування та обману. За порушення принципів академічної доброчесності викладач може накладати такі санкції:

- усне зауваження;
- попередження про можливість притягнення до академічної відповідальності;
- зниження чи анулювання результатів оцінювання навчального завдання здобувача вищої освіти;
- повторне виконання навчального завдання;
- призначення додаткового навчання з питань академічної доброчесності;
- призначення додаткових контрольних заходів (додаткові індивідуальні навчальні завдання, тести тощо).

Документи стосовно академічної доброчесності (про плагіат, порядок здачі курсових робіт, кодекс честі студентів, документи Національного агентства стосовно доброчесності) наведені на сторінці **ЯКІСТЬ ОСВІТИ** сайту **НУВГП** - <http://nuwm.edu.ua/sp/akademichna-dobrochesnistj>

### **Вимоги до відвідування**

Пропущені заняття відпрацьовуються здобувачами самостійно та передбачають оформлення звіту виконання. Електронні варіанти лекцій доступні на платформі moodle та/або в групі Telegram відповідного курсу, лабораторні виконуються аудиторно та захищаються на поточному або наступному занятті чи на консультації, індивідуальні завдання надаються для студентів, що знаходяться на індивідуальному плані навчання.

Автор  
Старший викладач

Володимир ГЕРУС

Затверджено

Проректор з науково-педагогічної та  
навчальної роботи

Валерій СОРОКА



документ підписаний КЕП  
Номер документа СИЛ №639  
Підписувач Сорока Валерій Степанович  
Підписувач (дані КЕП):  
Сертифікат 3FAA9288358EC003040000009B6C3700C8C2C100