



Сучасні операційні системи

Робоча програма навчальної дисципліни (Силабус)

• Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	<i>12 Інформаційні технології</i>
Спеціальність	<i>121 Інженерія програмного забезпечення, 123 Комп'ютерна інженерія, 126 Інформаційні системи та технології</i>
Освітня програма	<i>Інженерія програмного забезпечення інформаційних систем, Інженерія програмного забезпечення комп'ютерних систем, Інтегровані інформаційні системи, Інформаційні управляючі системи та технології, Інформаційне забезпечення робототехнічних систем, Комп'ютерні системи та мережі</i>
Статус дисципліни	<i>Вибіркова</i>
Форма навчання	<i>Очна (денна)</i>
Рік підготовки, семестр	<i>3 курс, весняний семестр</i>
Обсяг дисципліни	<i>4 кредити / 120 год (36 годин лекцій, 18 годин комп'ютерних практикумів (лабораторних робіт), 66 годин самостійна робота)</i>
Семестровий контроль / контрольні заходи	<i>Залік / Модульні контрольні роботи</i>
Розклад занять	<i>https://my.kpi.ua https://schedule.kpi.ua/</i>
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	<i>Лектор: старший викладач Олесь Володимирович Ковтунець, kovtunets.oles@iit.kpi.ua Комп'ютерні практикуми: старший викладач Олесь Володимирович Ковтунець, kovtunets.oles@iit.kpi.ua, асистент Богдан Борисович Бабич, babych.bohdan@iit.kpi.ua</i>
Розміщення курсу	<i>https://classroom.google.com/c/ODQyNDIzMjEyMTY5?cjc=4zq5pm5k</i>

• Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Вивчення дисципліни спрямовано на вивчення особливостей, процесів організації роботи та підходів до розробки архітектури операційних систем, що дає змогу виявляти проблеми швидкодії та оптимізувати роботу існуючого програмного забезпечення, а також власноруч створювати ефективне програмне забезпечення. Дисципліна розвиває та удосконалює професійні навички студентів, набуті в попередні роки навчання з дисциплін.

Предмет навчальної дисципліни – методи і засоби організації передачі та опрацювання інформації в операційних системах.

Метою дисципліни є отримання студентами фундаментальних знань про принципи побудови та функціонування сучасних операційних систем, формування розуміння суті та взаємодії процесів у операційних системах у розрізі роботи прикладного програміста.

За результатами вивчення дисципліни студент повинен набути **досвід** з оптимізації функціонування існуючого програмного забезпечення та проектування нового програмного забезпечення з урахуванням можливостей операційної системи.

Програмні результати навчання студента. Згідно з вимогами освітньо-професійної програми студент після засвоєння навчальної дисципліни повинен **знати**:

- Особливості операційних систем.
- Принципи побудови та функціонування операційних систем.
- Практики використання можливостей операційних систем.
- Процедури пошуку проблем швидкодії програмного забезпечення.
- Практики оптимізації програмного забезпечення.

2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

Вивчення дисципліни спирається на базові знання з комп'ютерної дискретної математика, алгоритмів та структур даних, основ комп'ютерних систем та мереж, та підходів до програмування і засобів візуалізації інтерфейсу користувача. Знання та навички, набуті студентом при вивченні дисципліни, використовуються в системному програмуванні, в розробці дипломних проєктів здобувачів першого (бакалаврського) рівня вищої освіти за освітніми програмами спеціальностей: 121 Інженерія програмного забезпечення 123 Комп'ютерна інженерія, 126 Інформаційні системи та технології..

3. Зміст навчальної дисципліни

Тема 1. Вступ до операційних систем.

Тема 2. Взаємодія ОС з апаратною частиною.

Тема 3. Бінарний програмний інтерфейс (ABI).

Тема 4. Пам'ять в операційній системі.

Тема 5. Виконувані файли.

Тема 6. Процеси в операційній системі.

Тема 7. Синхронізація процесів.

Тема 8. Файлові системи в ОС.

Тема 9. Мережева взаємодія.

Тема 10. Питання безпеки в операційній системі.

Тема 11. ОС Unix.

Тема 12. ОС Windows.

4. Навчальні матеріали та ресурси

Базова література

1. A. Tanenbaum, H. Bos. Modern Operating Systems. 4th edition. – 2015. – 1120 с.
2. W. Stallings. Operating Systems: Internals and Design Principles, 8th edition. – 2014. – 800 с.

Додаткова література

3. Шеховцов В. А. Операційні системи. — К.: Видавнича група BHV, 2005. — 576с.
4. An Introduction to x86_64 Assembly Language [Електронний ресурс] / О. Kovtunets // Google Drive. – 2021. 7 с. – Режим доступу до ресурсу: https://drive.google.com/file/d/1JDN80dq-Kqq9WpeK6gliMZTRbEaSPgok/view?usp=share_link
5. Introduction to x64 Assembly [Електронний ресурс] / О. Kovtunets // Google Drive. – 2021. 12 с. – Режим доступу до ресурсу: https://drive.google.com/file/d/1MLhnlkdOUYvurDmoSy_8fqXljl0t4HV7/view?usp=share_link
6. Linux assemblers: A comparison of GAS and NASM [Електронний ресурс] / R. Narayan // Google Drive. – 2021. 13 с. – Режим доступу до ресурсу: https://drive.google.com/file/d/1Ucu0mDeFjY3XOH41_PgOc3eQXzNnVVPm/view?usp=share_link

• Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

Матеріали для вивчення дисципліни розміщені викладачем на платформі Google Classroom <https://classroom.google.com/c/ODQyNDIzMjEуMTY5?cjc=4zq5pm5k>, до якого є доступ студентів. Контент платформи доступний із будь-якого місця в мережі Інтернет. Навчальний курс вміщає теоретичні матеріали, матеріали для виконання завдань комп'ютерних практикумів, та матеріали для самостійної роботи.

Лекції з дисципліни проводяться викладачем із використанням сучасних мультимедійних презентаційних технологій. На лекціях проводяться експрес-опитування, які дають можливість лектору отримати інформацію про якість засвоєння матеріалу та, за необхідності, розглянути більш детально складний матеріал.

Завдання комп'ютерних практикумів виконуються з використанням обраного редактора коду, системи контролю версій Git та при потребі віртуальної машини із відповідною операційною системою.

Під час проведення комп'ютерних практикумів використовуються методичні вказівки до виконання завдань комп'ютерних практикумів з дисципліни. Студент на початку семестру отримує календарний план початку та завершення захисту виконаних робіт.

Модульні контрольні роботи містять завдання з перевірки як теоретичних знань, так і практичних навичок, і проводяться на онлайн-заняттях за допомогою Zoom.

5.1. Тематика лекцій

№ з/п	Назва теми лекції та перелік основних питань (дидактичні матеріали: презентації Power Point або PDF, відеолекції)
1	Тема 1. Вступ до операційних систем. Предмет і задачі. Ідеї оптимізації. Пакутий принцип. Одно- та багатозадачність. Розподіл часу. Основні функції. Ядро ОС. Драйвери. Системні виклики, функції API. Архітектури комп'ютера: принципи та особливості. Література: 1, 2, 3.
2	Тема 2. Взаємодія ОС з апаратною частиною. Периферія: класифікація та керування. Апаратна частина та ОС. Апаратний таймер. Цикл роботи процесора. Переривання: пріоритети та генератори. Розмежування доступу. Схема вводу/виводу. Завантаження ОС. Література: 1, 2, 3.
3	Тема 3. Бінарний програмний інтерфейс (ABI). Суть, місце серед інтерфейсів. Мова програмування Асемблер: поняття, синтаксис, команди. Адресація пам'яті. Регістри процесора, їх описання. Стек. Угода про виклики. Системні виклики, способи задання викликів. Схема обробки системних викликів. Література: 1, 2, 3.
4	Тема 4. Пам'ять в операційній системі. Задачі керування пам'яттю. Віртуальні та фізичні адреси, налаштування. Фіксовані та динамічні розділи. Віртуальна пам'ять: схема адресації та завдання. Сегментна та сторінкова організації, схеми. Фрейми. Менеджер пам'яті. Література: 1, 2, 3.
5	Тема 5. Виконувані файли. Типи мов програмування. Збирання програм. Виконуваний файл, формати. Бібліотеки, способи підключення. Література: 1, 2, 3.
6	Тема 6. Процеси в операційній системі. Процес, його властивості. Паралельність процесів. Мультипрограмування. Квазіпаралельність. Потoki процесів, їх типи, види і стани. Переходи між станами. Життєвий цикл процесу. Породження і завершення процесу. Перемикання процесів. Література: 1, 2, 3.
7	Тема 7. Синхронізація процесів. Реентерабельність. Алгоритми диспетчера процесів. Міжпроцесна взаємодія, проблеми взаємодії. Література: 1, 2, 3.
8	Тема 8. Файлові системи в ОС. Термінологія, види, типові задачі файлових систем. Файли та каталоги. Операції з файлами. Файли і потоки процесів. Прив'язка файла до каталога. Операції з каталогами. Схеми розміщення файлів. Оптимізація роботи файлової системи. Література: 1, 2, 3.
9	Тема 9. Мережева взаємодія. Мережа – це комп'ютер. Основні принципи мережі. Проблематика мереж. Багаторівневий підхід. Задача зв'язування вузлів. Протоколи, інтерфейси, стеки. Модель OSI. Інтерфейс BSD сокетів. Віддалений виклик процедур. Сериалізація. Література: 1, 2, 3.

10	Тема 10. Питання безпеки. Інформаційна безпека. Створення безпечної системи. Реалізації контролю доступу. Мандатна система: переваги та недоліки. Апаратні примітиви безпеки. Система безпеки в ОС. Математична модель СМО, імовірності. Література: 1, 2, 3.
11	Тема 11. ОС Unix. Сімейство нащадків Unix і Unix-подібних ОС, основні варіанти. Підтримка GUI. Управління залежностями. Принципи розробки під Unix. Критика Unix. Література: 1, 2, 3.
12	Тема 12. ОС Windows. Історія Windows. Windows NT. Ключові рішення Windows. Принципи розробки під Windows. Критика Windows. Література: 1, 2, 3.

5.2. Тематика комп'ютерних практикумів/лабораторних робіт

№ з/п	Назва теми лекції та перелік основних питань (дидактичні матеріали: презентації Power Point або PDF, відеолекції)
1	Комп'ютерний практикум 1. Базові команди в Linux. Мета: оволодіти практичними навичками роботи в системі Linux, ознайомитися із структурою файлової системи, основними командами роботи з файлами. Література: 1, 2, 3.
2	Комп'ютерний практикум 2. Написання shell-скриптів. Мета: ознайомитися з особливостями інтерфейсу взаємодії з операційною системою та отримати практичні навички написання скриптів командної оболонки, дослідити процес оптимізації shell-скрипта на великих обсягах даних. Література: 1, 2, 3.
3	Комп'ютерний практикум 3. Програмування взаємодії з операційною системою Мета: ознайомитися з довідками розробки в Linux/Unix-подібних операційних системах і системними викликами в таких ОС, отримати навички написання програм для Linux/Unix-подібних ОС, вивчити метод взаємодії з ОС за допомогою системних викликів. Література: 1, 2, 3.
4	Комп'ютерний практикум 4. Системні виклики та асемблер. Мета: ознайомитися з мовою програмування Асемблер, навчитися використовувати її для вирішення задач управління ОС, засвоїти навички написання програм з використанням мови програмування Асемблер. Література: 1, 2, 3.
5	Комп'ютерний практикум 5. Реалізація файлової системи. Мета: ознайомитися з механізмом взаємодії ОС із файловою системою, освоїти бібліотеку FUSE, засвоїти концепцію побудови файлової системи, реалізувати файлову систему у довідці користувача за допомогою бібліотеки FUSE. Література: 1, 2, 3

6	<p>Комп'ютерний практикум 6. Взаємодія через мережу.</p> <p>Мета: ознайомитися з інтерфейсом сокетів ОС для взаємодії програм через мережу, зрозуміти принцип мережевої взаємодії на прикладному рівні, отримати навички використання сокетів для створення клієнтських мережевих додатків.</p> <p>Література: 1, 2, 3.</p>
---	--

6. Самостійна робота студента

До самостійної роботи студента відноситься виконання завдань комп'ютерних практикумів, робота з документацією, опрацювання лекційного та додаткового теоретичного матеріалу за наданими презентаціями лекцій, навчальним посібником та додатковою літературою. Матеріали для самостійного вивчення дисципліни розміщені викладачем в електронному вигляді на платформі Google Classroom. Контент платформи доступний із будь-якого місця в мережі Інтернет. На самостійну роботу студент має витратити 66 годин.

Самостійна робота

№ з/п	Назва розділу, теми (окремого питання), що виноситься на самостійне опрацювання
1	Архітектура фон Неймана, вузькі місця, проблематика розпаралелювання.
2	Прошивки, завантаження з BIOS.
3	Підбір компілятора для комп'ютерного практикуму (лабораторної роботи).
4	Виділення пам'яті процесу, трансляція адрес у x86.
5	Написання підключуваних бібліотек.
6	Алгоритмами планування.
7	Схеми розміщення файлів.
8	Алгоритми оптимізації.
9	Сокетні з'днання.
10	Математична модель СМО, оцінка безпеки.
11	Принципи розробки під Unix.
12	Принципи розробки під Windows.

Задіяні методи і засоби навчання

Задіяні різні **методи навчання**, а саме за характером логіки пізнання, проблемно-орієнтований, на основі цілісного підходу до процесу навчання (лекційний матеріал та завдання до комп'ютерних практикумів (лабораторних робіт) тісно пов'язані між собою), самостійна навчально-пізнавальна діяльність, в процесі яких проявляється та оцінюється індивідуальна діяльність студентів (на самостійне опрацювання студентів виноситься ряд завдань, які студенти готують самостійно та презентують перед аудиторією, окремі завдання передбачають виконання поза межами відведених занять за розкладом, а на занятті вони лише захищають роботу), міждисциплінарний підхід до навчання (при вивченні дисципліни активно використовуються знання, які студенти здобули на інших дисциплінах. Перевага надається продуктивним методам, що спрямовані на активізацію і стимулювання навчально-пізнавальної діяльності студента.

Основними **засобами навчання** є:

- Інформаційні засоби (конспекти лекцій, презентації до лекцій, теоретичні відомості до виконання лабораторних робіт, відомості до виконання самостійної роботи студента, додаткова література до вивчення навчальної дисципліни);
- Дидактичні засоби (таблиці, презентації, демонстраційні приклади реалізації практичних завдань, програмні засоби навчального призначення у вигляді інструментарію для виконання лабораторних робіт);
- Технічні засоби:
 - Мультимедійні системи (мультимедійний проектор або дошка);
 - Апаратне та програмне забезпечення, зокрема засоби роботи в командному рядку та IDE;
 - Засоби комунікації зі студентами:
 - розміщення в інтернеті теоретичних матеріалів, завдань до лабораторних робіт, самостійної роботи студентів, питань до заліку та інше;
 - обмін повідомленнями (телеграм-чат).

● Політика та контроль

7. Політика навчальної дисципліни (освітнього компонента)

Студент повинен вивчати дисципліну протягом семестру, дотримуючись рекомендованого календарного плану виконання комп'ютерних практикумів, вивчення тем лекційного матеріалу та виконання модульних контрольних робіт. Усі завдання студент має виконувати самостійно і вчасно. Комп'ютерний практикум вважається виконаним, якщо студент захистив завдання практикума у викладача та розмістив звіт з виконання у відповідному розділі дисципліни на платформі Google Classroom.

При захистах комп'ютерних практикумів викладач дотримується такої політики справедливості:

- 1) студент, що захищає комп'ютерний практикум у відповідності з рекомендованим календарним планом, має вищий пріоритет у черзі на захист;
- 2) чим більше студент відстає від календарного плану захисту виконаних завдань, тим відповідно нижчим є його пріоритет у черзі на захист;
- 3) після захисту одного комп'ютерного практикуму на парі студент може захистити ще один лише після того, як усі присутні на занятті студенти захистили свій один комп'ютерний практикум.

Студент не має права претендувати на захист комп'ютерного практикуму поза межами пар, передбачених розкладом в семестрі. Можливість захистити комп'ютерний практикум надається в межах першої та другої перездач.

Дві модульні контрольні роботи проводяться в семестрі з використанням Zoom. Дату та час проведення робіт оголошує лектор на початку семестру, відмічає її у журналі та нагадує про проведення роботи на лекційних заняттях. Повторне виконання модульної контрольної роботи поза призначеним часом її виконання не передбачено.

Якщо студент не задоволений сумарною оцінкою за модульні контрольні роботи, він може переписати їх на семестровій контрольній роботі, що містить завдання з усього курсу.

Заохоченням до своєчасного засвоєння теоретичного матеріалу є бали за правильні відповіді на запитання під час експрес-опитування на лекціях.

Оцінювання студентів здійснюється згідно рейтингової оцінки рівня підготовки студентів з дисципліни. Поточний стан успішності студенти можуть бачити в електронному журналі.

Рейтингова система оцінювання з кредитного модуля описана у наступному розділі робочої програми.

8. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

Поточний контроль: завдання комп'ютерних практикумів та МКР.

Календарний контроль: двічі на семестр як моніторинг поточного стану виконання вимог робочої програми.

Семестровий контроль: залік.

8.1 Поточний контроль

Поточний контроль успішності засвоєння знань студентами виконується шляхом виконання ними завдань комп'ютерних практикумів та МКР. Таким чином, семестровий рейтинг студента з дисципліни складається з балів, котрі він отримує за:

- виконання та захист завдань комп'ютерних практикумів;
- виконання модульних контрольних робіт;
- відповіді на заліку (залікову роботу).

8.2 Система рейтингових (вагових) балів та критерії оцінювання

Комп'ютерні практикуми

Вагові бали завдання кожного комп'ютерного практикуму наведено у таблиці 2.

Сумарний ваговий бал за даний контрольний захід складає 36 балів. Критерії оцінювання завдань включають якість їх виконання і якість захисту із оформленням звіту (таблиця 2).

Таблиця 2 – Вагові бали та критерії оцінювання лабораторних робіт (комп'ютерних практикумів)

Завдання	Критерії та бали	
	Виконання та захист зі звітом	Макс. сума балів
1	- завдання виконане та захищене без зауважень – 6 балів;	6
2	- завдання виконане достатньо повно з деякими похибками та захищене із зауваженнями – 4-5 балів;	6
3	- не повністю виконане завдання, при захисті частина відповідей відсутня або надано часткові відповіді – 2-3 бали;	6
4	- є суттєві зауваження без відповідей – 1 бал;	6
5		6
6		6
Разом		36

Виконаний та зарахований цикл всіх комп'ютерних практикумів є умовою допуску до семестрового контролю. Студенти, що на момент консультації перед семестровим контролем не захистили завдання комп'ютерних практикумів, не допускаються до основної задачі та готуються до перескладання.

Для допуску до перескладання семестрового контролю треба у визначений викладачем термін здати всі заборгованості з комп'ютерних практикумів.

Модульна контрольна робота

Модульна контрольна робота складається з 14 тестових питань за темами, котрі були розглянуті при вивченні освітнього компоненту. Максимальний ваговий бал за даний контрольний захід для = 14 балів.

Оцінювання кожного питання в МКР здійснюється наступним чином:

- правильна відповідь на запитання: 1 бал;
- неправильна відповідь на запитання: 0 балів.

Виконана на позитивну оцінку МКР є умовою допуску до семестрового контролю.

Студенти, що на момент консультації перед семестровим контролем не виконали модульну контрольну роботу, не допускаються до основної здачі та готуються до перескладання.

Для допуску до перескладання семестрового контролю треба у визначений викладачем термін виконати модульну контрольну роботу.

8.3 Календарний контроль

Календарний контроль провадиться двічі на семестр на 7 та 13 тижнях як моніторинг поточного стану виконання вимог силабусу - студент отримує «атестовано» під час першого та другого календарного контролю, якщо його поточний рейтинг складає не менше за 50% від максимальної кількості балів, можливої на момент контролю.

За результатами навчальної роботи за перші 7 тижнів «ідеальний студент» має набрати 18 балів. На першій атестації (7-й тиждень) студент отримує «зараховано», якщо його поточний рейтинг не менший за 9 балів.

За результатами 13 тижнів навчання «ідеальний студент» має набрати 36 балів. На другій атестації (13-й тиждень) студент отримує «зараховано», якщо його поточний рейтинг не менший за 18 балів.

8.4 Розрахунок шкали рейтингу

Семестровий контроль: залік.

У випадку централізованого рішення в університеті та з дозволу кафедри може здійснюватися перерахунок рейтингу, отриманого протягом семестру, за формулою, наведеною в Регламенті проведення семестрового контролю в дистанційному режимі.

Верхня межа рейтингової шкали з дисципліни становить 100 балів.

Максимальна сума вагових балів $R1$ за виконання оцінювальних заходів протягом семестру становить 50 балів. Сума вагових балів, які отримує студент за роботу протягом семестру, обчислюється за формулою:

$$R1 = W + M$$

де

W – сума балів, які студент отримав протягом семестру за оцінені завдання, перелічені в табл. 2;

M – сума балів, отриманих за модульну контрольну роботу.

Залікова складова шкали $R2 = 50$ балів.

Сумарний ваговий бал (максимальний) за семестр R , визначається як:

$$R = R1 + R2 = 36 + 14 + 50 = 100$$

Для отримання студентом відповідних оцінок його рейтингова оцінка R переводиться згідно з таблицею відповідності рейтингових балів оцінкам за університетською шкалою.

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

Кількість балів	Оцінка
100-95	Відмінно
94-85	Дуже добре

84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно
Не виконані умови допуску	Не допущено

1. Додаткова інформація з дисципліни (освітнього компонента)

Якщо студент пред'являє сертифікат проходження курсів з системного програмування або низькорівневого програмування, у нього є можливість здати курс екстерном. Для цього студент проходить тестування та співбесіду з викладачем.

Робочу програму навчальної дисципліни (силабус):

складено старшим викладачем кафедри ІПІ Ковтунцем О.В.

ухвалено кафедрою ІПІ (протокол № 2/1 від 10.10.2025 р.)

погоджено Методичною комісією факультету (протокол № 3 від 17.10.2025