

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ І.І. МЕЧНИКОВА
Кафедра математичного забезпечення комп'ютерних систем



ЗАТВЕРДЖУЮ

Проректор з науково-педагогічної роботи

_____ 20__ р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

ВЛ06 «Мережеве програмування»

(назва навчальної дисципліни)

Рівень вищої освіти _____ Перший (бакалаврський) _____

Галузь знань: _____ 12 – Інформаційні технології _____

Спеціальність _____ 123 – Комп'ютерна інженерія _____
код і назва спеціальності (тей)

Освітньо-професійна/наукова програма: _____ Комп'ютерна інженерія _____

дисципліни

(підпис)

(підпис)

(підпис)

(підпис)

(підпис)
(підпис)

2

1. Опис навчальної дисципліни

Найменування показників	Галузь знань, напрям підготовки, освітньо-кваліфікаційний рівень	Характеристика навчальної дисципліни	
		денна форма навчання	заочна форма навчання
Загальна кількість: кредитів – 3 годин – 90 змістових модулів – 3 (вид завдання)	Галузь знань <u>12 – Інформаційні технології</u> (шифр і назва) Спеціальність <u>123 – Комп’ютерна інженерія</u> (шифр і назва) Рівень вищої освіти: <u>перший</u> <u>(бакалаврський)</u>	Вибіркова	
		Рік підготовки:	
		4-й	5-й
		Семестр	
		8-й	10-й
		Лекції	
		24 год.	6 год.
		Практичні, семінарські	
		год.	год.
		Лабораторні	
		24 год.	6 год.
		Самостійна робота	
		42 год.	78 год.
у т.ч. ІНДЗ*: -			
Форма підсумкового контролю: іспит			

2. Мета дисципліни

Дисципліна “Мережеве програмування” присвячений вивченню методик створення мережеских додатків у середовищі Unix, інтерфейсів операційної системи, що застосовуються у мережевому програмуванні, а також оптимізації мережевої взаємодії. Пильна увага приділяється параметрам функціонування мережеских протоколів та різним способам організації введення-виведення

Метою викладання дисципліни є отримання знань, вмінь та навичок, необхідних фахівцю, який спеціалізується в області створення та використання програмного забезпечення для розробки та експлуатації комп’ютерних систем та мереж.

Завданнями вивчення навчальної дисципліни є:

- ознайомлення з сучасними технологіями програмування комп’ютерних мереж;
- отримання практичних навичок роботи з конкретними технологіями мережевого програмування.

Процес вивчення дисципліни спрямований на формування елементів наступних **компетентностей** (згідно ОПП «Інформаційні системи та технології від 2019 р.):

а) загальних: —

б) фахових:

КС2. Здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення.

КС3. Здатність створювати системне та прикладне програмне забезпечення комп’ютерних систем та мереж.

КС9. Здатність системно адмініструвати, використовувати, адаптувати та експлуатувати наявні інформаційні технології та системи.

КС15. Здатність аргументувати вибір методів розв’язування спеціалізованих задач, критично оцінювати отримані результати, обґрунтовувати та захищати прийняті рішення.

Програмні результати навчання:

ПР1. Знати і розуміти наукові положення, що лежать в основі функціонування комп’ютерних засобів, систем та мереж.

ПР2. Мати навички проведення експериментів, збирання даних та моделювання в комп’ютерних системах. ПР 4. Проводити системний аналіз об’єктів проектування та обґрунтовувати вибір структури, алгоритмів та способів передачі інформації в інформаційних системах та технологіях.

ПР6. Вміти застосовувати знання для ідентифікації, формулювання і розв’язування технічних задач спеціальності, використовуючи методи, що є найбільш придатними для досягнення поставлених цілей.

ПР10. Вміти розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем, розраховувати, експлуатувати, типове для спеціальності обладнання

Очікувані результати навчання. У результаті вивчення навчальної дисципліни студент повинен

ЗНАТИ:

- основи мережевого програмування та інтерфейсу для різних сімейств протоколів; організацію взаємодії між процесами;
- призначення та склад прикладного програмного інтерфейсу сокетів, основні відмінності реалізації цього інтерфейсу для ОС сімейства UNIX і Windows;
- способи опису адресів сокетів, функції створення, встановлення режимів, використання та видалення сокетів;
- моделі мережевого введення-виведення;
- відмінність між багатоадресною та широкомовною розсилкою, принципи використання багатоадресних груп.

ВМІТИ:

- проектувати та реалізовувати застосунки в середовищі UNIX;
- застосовувати методологію розробки мережеских програм для UNIX;
- обирати програмну платформу для реалізації мережевого додатка, модель мережевого входу-виводу, тип мережевої взаємодії,
- застосовувати структури, файли, мережевий інтерфейс та функції інших користувачів для різних сімейств протоколів;
- використовувати параметрів сокетів для оптимізації мережевої взаємодії
- дотримуватись основних підходів щодо економії пам'яті та ресурсів при створенні мережеских програм.

Курс складається з наступних частин: теоретичної (лекції), практичної (завдання, лабораторні роботи, тести для самоперевірки) та самостійної роботи.

Лабораторні роботи виконуються в середовищі операційної системи UNIX, яка застосовується сьогодні практично в усіх сферах інформаційних систем.

3. Зміст навчальної дисципліни

Змістовний модуль 1. Введення до мережевого програмування

Тема 1. Сімейство TCP/IP протоколів в UNIX та його співставлення з моделлю OSI стандарту ISO. Рівні TCP/IP протоколів в UNIX, адресація та інкапсуляція.

Література: [1].

Тема 2. Концепції програмування на базі TCP/IP. Сервіси, орієнтовані на підключення, і TCP. Сервіси без підключення та UDP.

Література: [1].

Змістовний модуль 2. Мережеве програмування за допомогою сокетів

Тема 3. Введення у сокети. Структури адрес сокету IPv4 та IPv6. Управління порядком байтів. Читання з сокету та запис у сокет.

Література: [1].

Тема 4. Управління сокетами. Створення сокету. З'єднання клієнта із сервером. Підготовка сервера до з'єднання. Прийом з'єднання. Закриття з'єднання.

Література: [1].

Тема 5. Розробка клієнта та сервера TCP на прикладі ехо-серверу. Мультиплексування введення-виведення. Моделі введення-виведення.

Література: [1].

Тема 6. Розробка клієнта та сервера UDP. Модель взаємодії клієнта та сервера UDP. Втрата дейтаграм. Валідація дейтаграм. Мультиплексування для UDP сервера.

Література: [1].

Тема 7. Претворення імен та адрес. Система доменних імен DNS. Запис ресурсів. Розпізнавачі та сервери імен. Альтернативи DNS. Розробка TCP та UDP-клієнтів дати та часу.

Література: [1,3,5,12].

Змістовний модуль 3. Розширені можливості сокетів

Тема 8. Мережеві служби. Принципи функціонування служби (демона) у Unix. Перетворення процесу на службу. Розробка служби дати та часу.

Література: [1,3,5,12].

Тема 9. Функції введення-виведення. Тайм-аут для введення-виведення. Прийом-передача допоміжних даних. Використання для сокетів функцій стандартного вводу-виведення.

Література: [1,2,4,9].

Тема 10. Введення-виведення, що не блокується. Функції, що викликають блокування введення-виведення. Неблоковані читання та запис. Переведення сокетів у стан, що не блокується. Цикл обробки читання чи запису. Усунення блокування за допомогою паралелізації процесів. Варіанти функцій connect, assert, що не блокуються..

Література: [1,2,6,7].

Тема 11. Широкомовна передача. Широкомовні адреси. Спрямована та широкомовна передача. Фрагментація IP-пакетів при широкомовній передачі.

Література: [1,2,3,5].

Тема 12. Багатоадресна передача. Робоча дисципліна Адреса багатоадресної передачі IPv4 та IPv6. Область дії адрес багатоадресної передачі. Багатоадресна та широкомовна передачі. Багатоадресна передача у глобальній мережі. Багатоадресна передача від відправника (SSM). Параметри сокетів багатоадресної передачі. Отримання анонсів сигналів багатоадресної передачі. Відправлення та отримання даних.

Література: [1,3,4].

4. Структура навчальної дисципліни

Назви змістових модулів і тем	Кількість годин											
	Денна форма						Заочна форма					
	Форма контролю	Усього	у тому числі				Форма контролю	Усього	у тому числі			
			л	п	лаб	ср			л	п	лаб	ср
1	2	3	4	5	6	7	2	3	4	5	6	7
Змістовний модуль 1. Введення до мережевого програмування												
Тема 1.	КО	6	2		2	2	КО	6	0,5		0,5	5
Тема 2.	КО	6	2		2	2	КО	6	0,5		0,5	5
Змістовний модуль 2. Мережеве програмування за допомогою сокетів												
Тема 3.	КО	8	2		2	4	КО	8	0,5		0,5	7
Тема 4.	КО	8	2		2	4	КО	8	0,5		0,5	7
Тема 5.	КО	8	2		2	4	КО	8	0,5		0,5	7
Тема 6.	КО	6	2		2	2	КО	6	0,5		0,5	5
Тема 7.	КО	8	2		2	4	КО	8	0,5		0,5	7
Змістовний модуль 3. Розширені можливості сокетів.												
Тема 8.	КО	8	2		2	4	КО	8	0,5		0,5	7
Тема 9.	КО	8	2		2	4	КО	8	0,5		0,5	7
Тема 10.	КО	8	2		2	4	КО	8	0,5		0,5	7
Тема 11.	КО	8	2		2	4	КО	8	0,5		0,5	7
Тема 12.	КР	8	2		2	4	КР	8	0,5		0,5	7
Всього годин		90	24	0	24	42		90	6	0	6	78

Форма контролю:

КО – контрольне опитування (поточне)

ІЗ – індивідуальне завдання (домашнє)

КР – контрольна робота

КМ – контроль модуля за тестовою системою

5. Теми семінарських занять

Семінарські заняття не передбачені

6. Теми практичних занять

Практичні заняття не передбачені

7. Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
1	Організація клієнт-серверної взаємодії на базі потокових сокетів	8
2	Організація клієнт-серверної взаємодії на базі дейтаграмних сокетів	8
3	Багатоадресне і широкомовленне разсилка	8
	Разом	24

8. Самостійна робота

№ з/п	Назва теми	Кількість годин	
		Денна	Заочна
1	2	3.	4.
1	Сімейство TCP/IP протоколів в UNIX та його співставлення з моделлю OSI стандарту ISO. Рівні TCP/IP протоколів в UNIX, адресація та інкапсуляція. / підготовка до лекцій	2	5
2	Концепції програмування на базі TCP/IP. Сервіси, орієнтовані на підключення, і TCP. Сервіси без підключення та UDP / підготовка до лекцій та лабораторних занять с	2	5
3	Введення у сокети. Структури адрес сокету IPv4 та IPv6. Управління порядком байтів. Читання з сокету та запис у сокет / підготовка до лекцій та лабораторних занять с	4	7
4	Управління сокетом. Створення сокету. З'єднання клієнта із сервером. Підготовка сервера до з'єднання. Прийом з'єднання. Закриття з'єднання / підготовка до лекцій та лабораторних занять	4	7
5	Розробка клієнта та сервера TCP на прикладі ехо-серверу. Мультиплексування введення-виведення. Моделі введення-виведення / підготовка до лекцій та лабораторних занять с	4	7
6	Розробка клієнта та сервера UDP. Модель взаємодії клієнта та сервера UDP. Втрата дейтаграм. Валідація дейтаграм. Мультиплексування для UDP сервера / підготовка до лекцій та лабораторних занять	2	5
7	Претворення імен та адрес. Система доменних імен DNS. Запис ресурсів. Розпізнавачі та сервери імен. Альтернативи DNS. Розробка TCP та UDP-клієнтів дати та часу / підготовка до лекцій та лабораторних занять	4	7
1	2	3.	4.

8	Мережеві служби. Принципи функціонування служби (демона) у Unix. Перетворення процесу на службу. Розробка служби дати та часу / підготовка до лекцій та лабораторних занять.	4	7
9	Функції введення-виведення. Тайм-аут для введення-виведення. Прийом-передача допоміжних даних. Використання для сокетів функцій стандартного вводу-виведення / підготовка до лекцій та лабораторних занять с	4	7
10	Введення-виведення, що не блокується. Функції, що викликають блокування введення-виведення. Неблоковані читання та запис. Переведення сокетів у стан, що не блокується. Цикл обробки читання чи запису. Усунення блокування за допомогою паралелізації процесів. Варіанти функцій connect, assert, що не блокуються. / підготовка до лекцій та лабораторних занять	4	7
11	Широкомовна передача. Широкомовні адреси. Спрямована та широкомовна передача. Фрагментація IP-пакетів при широкомовній передачі. / підготовка до лекцій та лабораторних занять	4	7
12	Багатоадресна передача. Робоча дисципліна Адреса багатоадресної передачі IPv4 та IPv6. Область дії адрес багатоадресної передачі. Багатоадресна та широкомовна передачі. Багатоадресна передача у глобальній мережі. Багатоадресна передача від відправника (SSM). Параметри сокетів багатоадресної передачі. Отримання анонсів сигналів багатоадресної передачі. Відправлення та отримання даних / підготовка до лекцій лабораторних занять с	4	7
	Разом	42	78

9. Методи навчання

Основна підготовка студентів здійснюється на лекційних та лабораторних заняттях. Під час викладання курсу використовуються такі методи навчання:

- словесні (лекція, пояснення);
- наочні (презентація);
- практичні (лабораторні роботи);
- робота з літературними джерелами (самостійна робота студентів).

10. Форми контролю і методи оцінювання

Протягом семестру студенти виконують три лабораторних роботи. В кожній роботі оцінюються якість виконання та якість відповідей на контрольні запитання. В разі невиконання лабораторної роботи в строк оцінка знижується на 2% за кожен тиждень прострочки.

Крім того, кожен студент готує доповідь з презентацією на узгоджену з викладачем тему. Наприкінці семестру студенти виконують по одній теоретичній контрольній роботі. Перелік питань міститься у п. 11.1

10.1. Критерії оцінювання на підсумковому контролі:

1. Відповідь повинна бути повною і короткою. Вона не повинна мати в собі матеріал, що не відноситься до сутті питання.
2. Чітко формулювати твердження, вправно застосовувати необхідні формули і знання основних питань програми.
3. Відповіді, що мають помилкові твердження оцінюються виходячи з близькості відповіді до правильної.
4. Пропуски в обґрунтуванні тверджень враховуються і це призводить до зменшення кількості балів.
5. Малі недоліки, неточності при викладенні матеріалу, зменшують кількість балів.
6. Незнання і нерозуміння основної ідеї теоретичного питання або задачі призводить до зняття до 90 % балів.
7. Якщо відповідь на питання відсутня то виставляється нуль балів.

10.2. Шкала оцінювання: національна та ECTS

Загальна сума балів	Оцінка ECTS	Національна шкала	
90 — 100	A – «відмінно»	5 «відмінно»	«залік»
85 — 89	B – «дуже добре»	4 «добре»	
75 — 84	C – «добре»		
70 — 74	D – «задовільно»	3 «задовільно»	
60 — 69	E – «допустимо»		
35 — 59	F – «незадовільно з можливістю повторного складання»	2 «незадовільно»	«незалік»
0 — 34	FX – «незадовільно з обов’язковим повторним курсом»		

11. Перелік контрольних, екзаменаційних (залікових) питань

11.1. Питання до модульної контрольної роботи №1

1. Дайте порівняльну характеристику протоколів TCP UDP.

2. Перелічить команди інтерфейсу транспортного рівня, пояснить, що вони означають.
3. Які команди інтерфейсу транспортного рівня відсутні у простішого UDP-клієнта і присутні в TCP-клієнті?
4. Що таке сокет?
5. Якими параметрами характеризується сокет.
6. Скільки сокетів необхідно для взаємодії клієнта та сервера?
7. Що таке адреса сокета?
8. Назвіть типи сокетів. У якому разі краще використовувати той чи інший тип сокетів?
9. Яку адресу можна використовувати для прослуховування всіх мережних інтерфейсів локального вузла?
10. Що таке порт? Як поділяються номери портів?
11. Яка функція використовується при створенні сокета? Охарактеризуйте параметри котрі передаються у функцію?
12. Опишіть поля структури `sockaddr_in`.
13. Який порядок байтів використовується в Intel-сумісних процесорах?
14. Який порядок байтів застосовується до роботи з сокетами?
15. Назвіть функції, які задають порядок байтів та опишіть їх параметри.
16. Назвіть функції, що служать для перетворення адрес та опишіть їх параметри.
17. Які номери портів можуть використовувати прикладні програми при роботі з сокетами?
18. Яка функція дозволяє дізнатися номери портів, які використовуються стандартними службами? Який файл містить цю інформацію?
19. Чим відрізняється процес отримання та відправлення даних на сокеті, що не потребує з'єднання?
20. Як здійснюється коректне завершення сеансу роботи із сокетом?
21. У чому відмінність завершення роботи з потоковим та дейтаграмним сокетом?
22. Що робить функція прослуховування, що викликається сервером TCP?
23. Правда, чи функція `listen`, що викликається сервером TCP, очікує підключення першого клієнта? Якщо ні, то що тоді робить функція `listen`?
24. Опишіть принцип роботи дейтаграмних сокетів?
25. Що в API сокетів називають сімейством адрес? Наведіть приклади сімейств і адрес, що належать до них.
26. Які сокети називають дейтаграмними (перелічить їхні особливості)?
27. Які сокети називають блокувальними? Наведіть приклади, як це проявляється.
28. Назвіть послідовність операцій із сокетом, які необхідно здійснити для приймання дейтаграми, і відповідні функції API.
29. Назвіть послідовність операцій із сокетом, які необхідно здійснити для широкомовної розсилки дейтаграм, і відповідні функції API.

30. Вкажіть призначення спеціальних адрес IPv4 і діапазонів: 127.0.0.1, 0.0.0.0, 255.255.255.255, 172.16.0.0/12.
31. Які порти називають привілейованими? Наведіть приклади відомих привілейованих портів не з опису ЛР.
32. Що станеться, якщо спробувати прийняти дейтаграму, надавши recvfrom() буфер недостатньо великого розміру?
33. Що станеться, якщо спробувати надіслати sendto() дейтаграму більшого розміру, ніж допустимо в цій мережі?
34. Сформулюйте алгоритм роботи простішого клієнта (UDP).
35. Сформулюйте алгоритм роботи простішого однопоточного TCP сервера.
36. Сформулюйте алгоритм роботи простішого однопоточного UDP сервера.
37. Як сервер TCP, що спілкується одночасно з кількома клієнтами, перевіряє від того, який клієнт отримав дані функції recv?
38. Якою є найбільша довжина, яку ми можемо передати функції sendto для сокету UDP/IPv4, тобто якою є найбільша кількість даних, які можуть поміститися в дейтаграму UDP/IPv4? Що змінюється у разі UDP/IPv6?
39. У якій ситуації розробник програми швидше за все скористається функцією sctp_peeloff?
40. Чому передача даних користувача в третьому пакеті рукостискання можлива тільки для сокетів типу «один-багатьом»?
41. У якій ситуації дані можуть бути передані в третьому і четвертому пакетах чотириетапного рукостискання?
42. Чому для служби DNS використовується протокол UDP?
43. Яке поле заголовка UDP забезпечує мультиплексування пакетів між додатками?
44. Як програми визначають, де знаходиться найближчий сервер імен?
45. В чому полягає різниця між Winsock та сокетами, котрі використовуються в UNIX подібних системах?

11.2. Питання до підсумкового контролю

У підсумковий контроль включено контрольні питання до модульних контрольних робіт.

12. Розподіл балів, які отримують студенти

Поточне тестування та самостійна робота												Підсумковий контроль (іспит)	Сума балів
Змістовий модуль №1		Змістовий модуль №2					Змістовий модуль №3						
T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12		
5	5	5	5	5	5	5	5	5	5	5	5	40	100

T1, T2 ... – теми змістових модулів

Шкала оцінювання: національна та ECTS

Загальна сума балів	Оцінка ECTS	Національна шкала	
90 — 100	A – «відмінно»	5 «відмінно»	«залік»
85 — 89	B – «дуже добре»	4 «добре»	
75 — 84	C – «добре»		
70 — 74	D – «задовільно»	3 «задовільно»	
60 — 69	E – «допустимо»		
35 — 59	F – «незадовільно з можливістю повторного складання»	2 «незадовільно»	«нез

13. Навчально-методичне забезпечення

Конспект лекцій; комплекс навчально-методичного забезпечення дисципліни; нормативні документи; презентаційні матеріали.

14. Рекомендована література

14.1. Основна література

1. Мельник В.М., Ройко О.Ю. Мережеве програмування в середовищах операційних систем UNIX та Linux: частина перша. Навчальний посібник, □ Луцьк. – «Вежа друк», 2017. - 192 с.,
2. Князева Н.О., Жуковецька С.Л., Трубіна Н.Ф. Системне програмування – Одеса: ВМВ, 2013 – 272с.
3. Рочкинд М. Программирование для UNIX. 2-е изд. Пере раб. И доп. – М. Издательско-торговый дом «Русская редакция»; СПб. БХВ – Санкт-Петербург, 2005.
4. Hands-On Network Programming with C: Learn socket programming in C and write secure and optimized network code - Taschenbuch – 13. Mai 2019

14.2. Допоміжна література

1. Трегубенко, І. Б. Сучасні технології програмування в мережах [Електронний ресурс] : навч. посіб. / І. Б. Трегубенко, Г. Т. Олійник, О. М. Панаско ; М-во освіти і науки України, Черкас. держ. технол. ун-т. – Черкаси : ЧДТУ, 2010. – 2-е вид., доповн. – 175 с.
2. Mastering Python Networking: Your one-stop solution to using Python for network automation, programmability, and DevOps, 3rd Edition Taschenbuch – 30. Januar 2020

3. Лав Р. Linux. Системное программирование. Пер. с англ., 2-е изд. — СПб.: Питер, 2014. — 448 с.:
4. М. Митчелл, Д.Оулдем, А.Саммюэл. Программирование для LINUX. Профессиональный подход. Пер. с англ., — М.: Издательский дом «Вильямс», 2002
5. Michael Kerrisk. The Linux Programming Interface. A Linux and UNIX System Programming Handbook [. – No Starch Press, October 2010, Pages: 1552
6. Sri Manikanta Palakollu. Pragmatic Example Applications in Linux and Unix-Based Operating/ Pragmatic Example Applications in Linux and Unix-Based Operating Systems – Systems Apress,-Berkeley, CA 27,October 2020, Pages: 1552

15. Інформаційні ресурси

1. The Internet Engineering Task Force [Электронный ресурс]. - Режим доступа : <https://tools.ietf.org/html/>
2. The Single UNIX Specification, Version 4 - Contents . [Электронный ресурс]. - Режим доступа : http://www.unix.org/version4/sus_contents.html
3. GCC online documentation – Режим доступа: <https://gcc.gnu.org/onlinedocs/>