

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ  
ІМЕНІ СЕМЕНА КУЗНЕЦЯ

**ЗАТВЕРДЖЕНО**

на засіданні кафедри  
інформаційних систем.  
Протокол № 1 від 22.08.2023 р.

**ПОГОДЖЕНО**

Проректор з навчально-методичної роботи



Каріна НЕМАШКАЛО

**СИСТЕМНЕ ПРОГРАМУВАННЯ**

робоча програма навчальної дисципліни (РПНД)

Галузь знань  
Спеціальність  
Освітній рівень  
Освітня програма

12 "Інформаційні технології"  
121 "Інженерія програмного забезпечення"  
перший (бакалаврський)  
"Інженерія програмного забезпечення"

Статус дисципліни  
Мова викладання, навчання та оцінювання

вибіркова  
українська

Розробник:  
к.т.н., доцент

підписано КЕП

Дмитро ГОЛУБНИЧИЙ

Завідувач кафедри  
інформаційних систем

Дмитро БОНДАРЕНКО

Гарант програми

Олег ФРОЛОВ

Харків  
2024

## ВСТУП

Сьогоднішні умови господарювання вимагають від фахівців з економічного управління всебічного використання новітніх інформаційних технологій. Широкі можливості комп'ютеризованих засобів в питаннях збору, обробки та видачі необхідної інформації здатні значно підвищити якість економічних розрахунків, зробити більш ефективним процес обґрунтування економічних рішень. Але успішне використання потужного комп'ютеризованого засобу неможливо без чіткого уявлення особливостей функціонування всіх його складових частин, а це, в свою чергу, вимагає твердих знань процесів, які відбуваються в операційній системі на рівні управління ресурсами під час їхньої роботи.

Знання основ побудови операційних систем стає все далі актуальнішим, оскільки тенденції розвитку комп'ютерної техніки свідчать про те, що з одного боку складність та функціональні можливості комп'ютерної техніки постійно і швидко зростають, а з другого боку, спостерігається постійна тенденція до персоніфікації цієї складної техніки. Тобто задача підтримки персонального комп'ютера в роботоздатному стані, налагоджування роботи його програмного забезпечення та конфігурації, своєчасний upgrade (patch, crack), все далі стає проблемою не професіоналів-фахівців, а конкретного користувача цього персонального комп'ютера.

Необхідність вивчення системного програмування визначається появою нових механізмів взаємодії системного та користувачевого програмного забезпечення, які потрібні для складання програм з розповсюджених мов програмування.

Метою викладання навчальної дисципліни "Системне програмування" є надання здобувачам вищої освіти системи спеціальних знань з засвоєння теоретичних основ побудови, принципів проектування, конфігурування й застосування різних сучасних операційних систем, які забезпечують організацію обчислювальних процесів у корпоративних інформаційних системах економічного, управлінського, виробничого, наукового й іншого призначення, а також надання практичних навичок щодо автоматизації повсякденних завдань адміністрування.

Завданнями навчальної дисципліни є:

- засвоєння принципів побудови, призначення, структури, функції та порядку створення системних програм для різних операційних систем, їх підсистем, механізмів керування ресурсами;
- засвоєння основних методів діагностики, відновлення, моніторингу й оптимізації складових операційної системи за рахунок використання системних утиліт, спеціалізованих бібліотек тощо;
- оволодіння навичками взаємодії з об'єктами операційної системи шляхом дослідження їх характеристик та способів експлуатації за рахунок використання системних утиліт.

Предметом навчальної дисципліни є сучасні теоретичні концепції та методології, принципи функціонування, взаємодія системних компонент у складі операційної системи.

Об'єктом навчальної дисципліни є процеси створення спеціального програмного забезпечення комп'ютерних систем з урахуванням особливостей функціонування операційних систем.

Результати навчання та компетентності, які формує навчальна дисципліна визначено в табл. 1.

Таблиця 1

**Результати навчання та компетентності, які формує навчальна дисципліна**

<b>Результати навчання</b>	<b>Компетентності, якими повинен оволодіти здобувач вищої освіти</b>
PH 14	ЗК 2
PH 01	ЗК 6
PH 04	СК 04
PH 07	СК 06
PH 15	СК 10

де, PH01. Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.

PH04. Знати і застосовувати професійні стандарти і інші нормативно-правові документи в галузі інженерії програмного забезпечення.

PH07. Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.

PH14. Застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення.

PH15. Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення.

ЗК2. Здатність застосовувати знання у практичних ситуаціях.

ЗК6. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

СК04. Здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами.

СК6. Здатність аналізувати, вибирати і застосовувати методи і засоби для забезпечення інформаційної безпеки (в тому числі кібербезпеки).

СК10. Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя.

# ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

## Зміст навчальної дисципліни

### Змістовий модуль 1. Системне програмування в Windows

#### Тема 1. Архітектура Windows-додатку

**1.1.** Каркасний Windows-додаток. Клас вікон. Головна функція. Створення вікна. Створення вікон з використанням існуючих класів. Визначення структури WNDCLASS. Реєстрація класу вікон. Обробка повідомлень у віконній функції. Функції підтримки вікон. Приєднання даних класу вікна до вікна. Зміна зовнішнього вигляду вікна. Інтерфейс прикладного програмування

**1.2.** Головна функція додатку. Типи даних, які використовуються в Windows. Цикл обробки повідомлень. Джерела повідомлень. Функції перехвату повідомлень. Черги повідомлень.

**1.3.** Органи управління. Редактори тексту. Смуги прокрутки.

**1.4.** Користувачеві системні інтерфейси. Інтерфейс графічних пристроїв. Ввід об'єктів до контексту пристроїв. Отримання описувача контексту пристрою. Контекст дисплея. Контекст класу. Загальний контекст. Приватний контекст. Функції контексту дисплея. Графічний користувачевий інтерфейс. Багатовіконний інтерфейс.

#### Тема 2. Механізми перехвату повідомлень

**2.1.** Базові механізми перехвату повідомлень. Windows-хукі. Завантаження хуків. Завдання та організація процедур перехвату повідомлень. Функції перехвату. Інсталяція та деінсталяція хука.

**2.2.** Спеціалізовані хукі. Локальні хукі. Віддалені хукі. Системні хукі. Типі хуків. Повідомлення для роботи з хуками. Визначення аргументів для процедур обробки хуків.

#### Тема 3. Процеси, потоки та засоби міжпроцесної взаємодії в Windows.

**3.1.** Процеси. Функції процесів. Ідентифікатори додатків. Командний рядок процесу. Змінні оточення. Стан процесу. Обробка помилок. Робочі каталоги процесу. Створення і завершення процесів. Захист процесів від нерентабельного коду. Обробка помилок та виключень.

**3.2.** Потоки. Умови створення потоків. Стек потоку. Стан потоку. Періоди виконання потоку. Створення і завершення потоків. Розподіл процесорного часу між потоками. Зміна класу пріоритету потоку. Затримка та поновлення виконання потоку.

**3.3.** Планування та диспетчеризація потоків. Види планування. Стратегії планування. Витіснявальна і невитіснявальна багатозадачність. Алгоритми планування потоків. Квантування. Планування потоків в системах реального часу.

**3.4.** Базові механізми міжпроцесової взаємодії. Міжпроцесова взаємодія на базі спільної пам'яті. Технології передавання повідомлень.

**3.5.** Базові механізми синхронізації потоків. Семафори. М'ютекси.

Блокування читання-записування. Синхронізація за принципом бар'єра. Взаємодія потоків у Windows. Програмний інтерфейс взаємодії. Види міжпроцесової взаємодії. Методи розподілюваної пам'яті. Методи передавання повідомлень. Технологія відображуваної пам'яті.

#### **Тема 4. Основи безпеки операційної системи Windows**

**4.1.** Основні завдання забезпечення безпеки. Базові поняття криптографії. Поняття криптографічного алгоритму і протоколу. Криптосистеми з секретним ключем. Криптосистеми з відкритим ключем. Гібридні криптосистеми. Цифрові підписи. Сертифікати.

**4.2.** Принципи автентифікації і керування доступом. Типи об'єктів, які захищаються. Формування списків управління доступом. Реалізація захисту особистих об'єктів. Облікові записи користувачів. Аудит. Загальні принципи організації аудиту. Журнал подій Windows.

**4.3.** Принципи шифрування даних на файлових системах. Створення криптопровайдера. Шифрувальна файлова система Windows. Мережева безпека даних. Захист інформації на мережному рівні.

### **Змістовий модуль 2. Системне програмування в Linux**

#### **Тема 5. Архітектура операційної системи Linux**

**5.1.** Архітектура дистрибутивів операційних систем Linux. Багаторівневі системи. Мікроядерна архітектура. Базові механізми ядра. Менеджери ресурсів. Інтерфейс системних викликів. Ресурси операційної системи. Апаратна залежність та переносимість операційної системи. Графічні інтерфейси.

**5.2.** Ядро операційної системи та його функції. Допоміжні модулі операційної системи. Ядро в привілейованому режимі та в режимі користувача. Обмін між додатками при використанні ядра в привілейованому режимі.

**5.3.** Управління файловою системою. Реалізація файлових систем Linux. Стандартні каталоги файлової системи. Файли пристроїв. Монтування. Файлові менеджери в Linux.

**5.4.** Команди для управління файлами та папками. Команди для роботи з файлами. Команди для роботи з папками. Команда ln та її опції. Пошук файлів в консолі. Програма перегляду довідкового керівництва man. Архівування. Утиліти tar та gzip.

**5.5.** Командний рядок Linux. Огляд командних оболонок Linux. Командна оболонка bash. Особливості роботи. Доступ до командного рядку. Символи командного рядку. Перевірка умов. Операції. Цикли та розгалуження. Внутрішні та зовнішні команди. Команди системного адміністрування. Підстанова команд.

**5.6.** Основи роботи зі сценаріями. Основи складання сценаріїв. Використання структурованих команд. Обробка введення даних користувачем. Подання даних. Управління сценаріями. Створення функцій. Написання сценаріїв для графічних робочих столів. Загальні відомості про редактори sed та gawk. Робота з іншими командними інтерпретаторами.

## **Тема 6. Створення, компіляція та компоновка програм в Linux**

**6.1.** Створення програм в Linux. Вихідний код. Компіляція. Компоновка. Багатофайлові проекти.

**6.2.** Автозбирання. Огляд засобів автозбирання в Linux. Утиліта make. Базовий синтаксис MakeFile. Константи make. Рекурсивний виклик make. Отримання додаткової інформації.

**6.3.** Оточення. Поняття оточення. Читання та модифікація оточення. Очищення оточення.

**6.4.** Концепція введення-виведення в Linux. Бібліотечні механізми введення-виведення мови C. Концепція низкоуровневого введення-виведення. Консольний введення-виведення. Введення-виведення в C ++.

**6.5.** Базові операції введення-виведення. Створення файлу: create(). Відкриття файлу: open(). Закриття файлу: close(). Читання файлу: read(). Запис файлу: write(). Довільний доступ: lseek().

## **Тема 7. Основи багатозадачності в Linux**

**7.1.** Основи багатозадачності в Linux. Бібліотечний підхід: system(). Процеси в Linux. Дерево процесів. Отримання інформації про процес.

**7.2.** Базова багатозадачність. Концепція розвилки: fork(). Передача управління: execve(). Сімейство exec(). Очікування процесу: wait().

**7.3.** Концепція потоків в Linux. Створення потоку: pthread\_create(). Завершення потоку: pthread\_exit(). Очікування потоку: pthread\_join(). Отримання інформації про потік: pthread\_self(), pthread\_equal(). Скасування потоку: pthread\_cancel(). Отримання додаткової інформації.

**7.4.** Розширена багатозадачність. Поступливість процесу: nice(). Сімейство wait(). Зомбі.

## **Тема 8. Методи міжпроцесної взаємодії в Linux**

**8.1.** Огляд методів взаємодії між процесами в Linux. Загальні відомості про взаємодії між процесами в Linux. Локальні методи взаємодії між процесами. Віддалена міжпроцесова взаємодія.

**8.2.** Сигнали. Поняття сигналу в Linux. Відправлення сигналу: kill(). Обробка сигналу: sigaction(). Сигнали і багатозадачність.

**8.3.** Використання загальної пам'яті. Виділення пам'яті: shmget(). Активізація спільного доступу: shmat(). Відключення спільного доступу: shmdt(). Контроль використання пам'яті: shmctl(). Використання семафорів. Контроль за семафора.

**8.4.** Використання загальних файлів. Розміщення файлу в пам'яті: mmap(). Звільнення пам'яті: munmap(). Синхронізація: msync().

**8.5.** Канали. Створення каналу: pipe(). Перенаправлення введення-виведення: dup2(). Створення іменованого каналу. Читання, запис і закриття FIFO.

Перелік лабораторних занять за навчальною дисципліною наведено в табл. 2.

**Перелік лабораторних занять**

Назва теми	Зміст
Тема 1.	Дослідження структури Windows додатку
Тема 2.	Дослідження механізмів перехвату повідомлень в багатовіконних документах
Тема 3.	Дослідження процесів та потоків
Тема 3.	Дослідження засобів обміну даними між процесами
Тема 4.	Дослідження засобів безпеки операційної системи за допомогою CryptoAPI
Тема 5.	Дослідження можливостей shell-програмування
Тема 6.	Дослідження способів створення програм в Linux
Тема 7.	Дослідження способів багатозадачного програмування в Linux
Тема 7.	Управління процесами в Linux
Тема 8.	Дослідження механізмів взаємодії між процесами в Linux

Перелік самостійної роботи за навчальною дисципліною наведено в табл. 3.

**Перелік самостійної роботи**

Назва теми	Зміст
Тема 1 – 8	Вивчення лекційного матеріалу
Тема 1 – 8	Підготовка до лабораторних занять
Тема 1 – 8	Підготовка до екзамену

Кількість годин лекційних та лабораторних занять, а також годин самостійної роботи наведено в робочому плані (технологічній карті) з навчальної дисципліни.

**МЕТОДИ НАВЧАННЯ**

У процесі викладання навчальної дисципліни для набуття визначених результатів навчання, активізації освітнього процесу передбачено застосування таких методів навчання, як:

Словесні (лекція (Тема 1, 3, 5, 7, 8), проблемна лекція (Тема 2), лекція-візуалізація (Тема 4, 6)).

Наочні (демонстрація (Тема 1 – 8)).

Лабораторна робота (Тема 1 – 8), кейс-метод (Тема 1 – 3, 7).

**ФОРМИ ТА МЕТОДИ ОЦІНЮВАННЯ**

Університет використовує 100 бальну накопичувальну систему оцінювання результатів навчання здобувачів вищої освіти.

**Поточний контроль** здійснюється під час проведення лекційних та лабораторних занять і має на меті перевірку рівня підготовленості здобувача

вищої освіти до виконання конкретної роботи і оцінюється сумою набраних балів:

– для дисциплін з формою семестрового контролю екзамен (іспит): максимальна сума – 60 балів; мінімальна сума, що дозволяє здобувачу вищої освіти скласти екзамен (іспит) – 35 балів.

**Підсумковий контроль** включає семестровий контроль та атестацію здобувача вищої освіти.

**Семестровий контроль** проводиться у формі семестрового екзамену (іспиту). Складання семестрового екзамену (іспиту) здійснюється під час екзаменаційної сесії.

Максимальна сума балів, яку може отримати здобувач вищої освіти під час екзамену (іспиту) – 40 балів. Мінімальна сума, за якою екзамен (іспит) вважається складеним – 25 балів.

**Підсумкова оцінка за навчальною дисципліною** визначається сумуванням балів за поточний та підсумковий контроль.

Під час викладання навчальної дисципліни використовуються наступні контрольні заходи:

Поточний контроль: захист лабораторних робіт (40 балів), письмова контрольна робота (тестування) (20 балів).

Семестровий контроль: Екзамен (40 балів).

Більш детальну інформацію щодо системи оцінювання наведено в робочому плані (технологічній карті) з навчальної дисципліни.

Приклад екзаменаційного білета та критерії оцінювання для навчальної дисципліни.

### **Приклад екзаменаційного білета**

Харківський національний економічний університет імені Семена Кузнеця  
Перший (бакалаврський) рівень вищої освіти  
Спеціальність «Інженерія програмного забезпечення»  
Освітньо-професійна програма «Інженерія програмного забезпечення»  
Семестр V  
Навчальна дисципліна «Системне програмування»

#### **ЕКЗАМЕНАЦІЙНИЙ БІЛЕТ № 1**

**Завдання 1** (евристичне, 20 балів).

Провести розробку системного Windows – застосунку та навести лістинг програмного коду всіх необхідних модулів, які повинні виконувати наступні дії:

Створити застосунок, в якому встановити функцію-фільтр на хук WH\_CBT з кодом типу HCBT\_ACTIVATE. Продемонструвати та запротоколювати усі дії встановленого хуку. Супроводити роботу програми наданням виводу повідомлень про встановлений хук.

**Завдання 2** (евристичне, 20 балів).

Провести розробку системного Linux – застосунку та навести лістинг програмного коду всіх необхідних модулів, які повинні виконувати наступні дії:



Написати застосунок, що визначає класи, старші, молодші номери основних периферійних пристроїв, які реалізуються в системі: клавіатура, миша, USB-накопичувач, принтер, відеоадаптер, накопичувач CD/DVD (при наявності), cardreader тощо.

Затверджено на засіданні кафедри інформаційних систем  
протокол № \_\_\_\_ від «\_\_» \_\_\_\_\_ 20\_\_ р.

Екзаменатор к.т.н., доц. Голубничий Д.Ю.

Зав. кафедрою к.т.н., доц. Бондаренко Д.О.

### Узагальнені критерії оцінювання

Підсумкові бали за екзамен складаються із суми балів за виконання всіх завдань, що округлені до цілого числа за правилами математики.

Алгоритм вирішення кожного завдання включає окремі етапи, які відрізняються за складністю, трудомісткістю та значенням для розв'язання завдання. Тому окремі завдання та етапи їх розв'язання оцінюються відокремлено один від одного таким чином:

#### Завдання 1 (евристичне).

Особливістю першого завдання є його виконання під керуванням операційної системи Windows. Завдання присвячене розробці системного програмного забезпечення поставленої задачі та навести лістинг програмного коду всіх необхідних модулів. Складається програма на мові програмування C++. Здобувач повинен створити проект в середовище Visual Studio. Основна мета вирішення цієї задачі – перевірка практичних вмінь здобувача використовувати інтерфейс прикладного програмування для обміну даними між програмою та операційною системою. При цьому здобувачу дозволяється користуватися існуючою довідковою літературою. Після перевірки програми здобувач отримує  $K_1$  балів за наступними вимогами (табл. 4).

Таблиця 4

#### Узагальнені критерії оцінювання за евристичними завданнями

Бали $K_1$ або $K_2$	Вимоги
0	Здобувач не володіє навчальним матеріалом, відповідь на запитання відсутня
1 – 2	Здобувач не володіє навчальним матеріалом, програма відсутня або взагалі не запускається.
3 – 6	Здобувач володіє навчальним матеріалом на елементарному рівні пізнання і відтворення окремих фактів та фрагментів. Програма, допоміжні файли та схема алгоритму виконані задовільно, є значні орфографічні або синтаксичні помилки, відсутні коментарі. Програма частково відповідає завданню. В разі її виконання після незначної доробки створюється основне вікно, але більшість функцій по роботі з ним не реалізовано. Під час роботи програма або зависає або аварійно перериває роботу.
7 – 10	Здобувач на рівні запам'ятовування відтворює основні положення навчального матеріалу. Програма, допоміжні файли та схема алгоритму виконані добре, є незначні орфографічні або синтаксичні помилки, відсутні коментарі. Допускається наявність невикористаних змінних, особистих функцій та процедур. Програма частково або не в повному обсязі відповідає завданню. В разі її виконання створюється основне вікно, віконна процедура, відомості про автора, але решта функцій по роботі з ними не реалізовано.

Бали К <sub>1</sub> або К <sub>2</sub>	Вимоги
11 – 14	Здобувач розкриває суть основних положень навчального матеріалу. Програма, допоміжні файли та схема алгоритму виконані добре, без орфографічних та синтаксичних помилок, мають мінімальну кількість коментарів, присутній стиль оформлення. Допускається наявність невикористаних змінних, особистих функцій та процедур. Програма частково або не в повному обсязі відповідає завданню. В разі її виконання – модулі в основному функціонують правильно, але вона потребує додаткової доробки.
15 – 17	Здобувач розкриває суть основних положень навчального матеріалу. Програма, допоміжні файли та схема алгоритму виконані добре, без орфографічних та синтаксичних помилок, мають мінімальну кількість коментарів, присутній стиль оформлення. Допускається наявність невикористаних змінних. Програма відповідає завданню, але допускається відсутність незначних допоміжних функцій, які полегшують роботу з нею. В разі її виконання – основні модулі функціонують правильно, але мають місце деякі несуттєві неточності виконання завдання.
18 – 20	Здобувач володіє узагальненими знаннями з організації операційних систем та застосовує їх в практичній роботі. Програма та допоміжні файли виконані відмінно, без орфографічних та синтаксичних помилок, мають чітку структуру, певну кількість коментарів, оформлені на належному рівні та не містять зайвих змінних, незадіяних функцій, процедур. Вона повністю відповідає завданню та в разі виконання – усі модулі функціонують правильно.

### **Завдання 2 (евристичне).**

Особливістю другого завдання є його виконання під керуванням операційної системи Linux. Завдання присвячене розробці системного програмного забезпечення в операційній системі Linux. Навести лістинг програмного коду всіх необхідних модулів. Складається програма на мові програмування C++. Основна мета вирішення цієї задачі – перевірка практичних вмінь здобувача використовувати інтерфейс прикладного програмування для обміну даними між програмою та операційною системою. При цьому здобувачу дозволяється користуватися існуючою довідковою літературою. Після перевірки програми здобувач отримує К<sub>2</sub> балів за наступними вимогами (табл. 4).

## **РЕКОМЕНДОВАНА ЛІТЕРАТУРА**

### **Основна**

1. Галісеєв Г.В. Системне програмування: навч. посіб. / Г.В. Галісеєв. – Київ: Університет "Україна", 2019. – 113 с.
2. Підручник з предмету: Системне програмування. - Дніпро: ДФК, 2024. - 255 с. Режим доступу: [https://library.kre.dp.ua/Books/2-4\\_kurs/Sистемне\\_програмування/Sистемне\\_програмування\\_C++.pdf](https://library.kre.dp.ua/Books/2-4_kurs/Sистемне_програмування/Sистемне_програмування_C++.pdf)
3. Черевик В.М. Операційна система Linux: принципи роботи з файловою системою. Навчальний посібник / В.М. Черевик, Л.І. Танцюра, С.С. Коротков, В.О. Сосновий. – Київ: ДУТ, 2021. – 147 с.
4. Мосіюк О. О. Операційні системи та системне програмування: навчально-методичний посібник // О. О. Мосіюк, А. Л. Федорчук. – Житомир: Вид-во ЖДУ ім. Івана Франка, 2022. – 76 с.

5. Tanenbaum E. Modern operating systems / E. Tanenbaum, H. Boss. – New Jersey: Pearson Prentice-Hall, 2020. – 1120 p.
6. Silberschatz A. Operating System Concepts / A. Silberschatz, G. Gagne, P.B. Galvin. – New Jersey: Wiley, 2021. – 1040 p.
7. Love R. Linux. System programming: Talking Directly to the Kernel and C Library / R. Love. – Newton: O'Reilly, 2023. – 448 p.
8. Schotts W. The Linux Command Line: A Complete Introduction / W. Schotts. – San Francisco: No Starch Press, 2022. – 480 p.
9. Suehring S. CompTIA Linux+ Practice Tests: Exam XK0-005 / S. Suehring. – New Jersey: Sybex-Wiley, 2022. – 1352 p.

#### **Додаткова**

10. Граннеман С. Linux. Кишеньковий довідник / С. Граннеман. – Київ: Діалектика, 2019. – 464 с.
11. Dave T. Shell Scripting. Linux, OS X and Unix / T. Dave, P. Brandon. – San Francisco: No Starch Press, 2019. – 392 p.
12. Cooper M. Advanced Bash Scripting Guide - Volume 1: An in-depth exploration of the art of shell scripting / M. Cooper. – Independently published, 2019. – 582 p.
13. Uzayr S-b. Linux: The Ultimate Guide / Sufyan bin Uzayr. – Boca Raton: CRC Press, 2022. – 305 p.
14. Stollings V. Operating system / V. Stollings. – Washington: Pearson, 2020. – 1264 p.
15. Hud O. The Project Management Information System in Linux / O. Hud, O. Veres //COLINS'2020, Volume II: Workshop. – Lviv: Ukraine, 2020. – Pp. 270 – 273.

#### **Інформаційні ресурси**

16. Офіційний сайт розробників ОС Linux Ubuntu [Електронний ресурс]. – Режим доступу: <https://ubuntu.com/>.
17. Linux [Електронний ресурс]. – Режим доступу: <https://www.linux.org/>.
18. Мізюк О. Путівник по Linux [Електронний ресурс]. – Режим доступу: [https://linuxguide.rozh2sch.org.ua/#\\_вступ](https://linuxguide.rozh2sch.org.ua/#_вступ)
19. Windows [Електронний ресурс]. – Режим доступу: <http://windows.microsoft.com/ru-ru/windows/home>.
20. Персональна навчальна система "Системне програмування" [Електронний ресурс]. – Режим доступу: <https://pns.hneu.edu.ua/course/view.php?id=4909>