

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



Програмування для Linux
робоча програма навчальної дисципліни

Галузь знань *12 "Інформаційні технології"*
Спеціальність *122 "Комп'ютерні науки"*
Освітній рівень *перший (бакалаврський)*
Освітня програма *Комп'ютерні науки*

Статус дисципліни *вибіркова*
Мова викладання, навчання та оцінювання *українська*

Завідувач кафедри
інформаційних систем

Ірина УШАКОВА

Харків
2021

ЗАТВЕРДЖЕНО
на засіданні кафедри *Інформаційних систем*
Протокол № 6 від 05.01.2021 р.

Розробник:
Голубничий Дмитро Юрійович, кандидат технічних наук, доцент.

**Лист оновлення та перезатвердження
робочої програми навчальної дисципліни**

Навчальний рік	Дата засідання кафедри – розробника РПНД	Номер протоколу	Підпис завідувача кафедри

1. Вступ

Анотація навчальної дисципліни

Найкращі технології – це рухома мішень. Коли технологія стагнує, суспільство зазвичай обганяє і переростає її. Linux, широко використовувана операційна система з відкритим кодом – це фундаментальна технологія, яка є основою для однієї з найбільш прогресивних сучасних ідей комп'ютерних систем. Тому, хоча протягом трьох десятиріч розвитку вона залишається вражаюче незмінною, ця операційна система забезпечує адаптованість. Linux домінує на всіх типах комп'ютерів, крім настільних (десктопи і ноути) і, напевно, мейнфреймів. Вбудовані системи, мобільні, сервери-хмари-суперкомп'ютери – всюди переважають системи на ядрі Linux. Ця операційна система провалилася на десктопах і мейнфреймах тільки тому, що в цих сегментах домінують монополії – Microsoft та IBM.

В результаті цього Linux знаходиться в унікальному становищі – з одного боку, це надійна інвестиція в розвиток своїх навичок, тому що вона не змінюється, з іншого – це удавана нескінченної рушійна сила для отримання нових навичок.

Дисципліна спрямована на формування у студентів загальних основ взаємодії системного та користувачевого програмного забезпечення, які потрібні для складання програм з розповсюджених мов програмування в операційній системі Linux на дистрибутиві Ubuntu.

Мета навчальної дисципліни: формування системи професійних знань з засвоєння основних концепцій файлів та процесів в операційній системі Linux, які дозволять студентам самостійно удосконалювати отримані навички як при роботі з терміналом, так й при створенні програмних додатків з урахуванням особливостей операційної системи Linux.

Характеристика навчальної дисципліни

Курс	3
Семестр	2
Кількість кредитів ECTS	5
Форма підсумкового контролю	залік

Структурно-логічна схема вивчення дисципліни

Пререквізити	Постреквізити
Програмування	Розподілені та паралельні обчислення
Основи об'єктно-орієнтованого програмування	Дипломний проект
Операційні системи	

Компетентності та результати навчання за дисципліною

Компетентності	Результати навчання
ЗК1. Здатність до абстрактного мислення, аналізу та синтезу	ПР1. Застосовувати знання основних форм і законів абстрактно-логічного мислення, основ методології наукового пізнання, форм і методів вилучення, аналізу, обробки та синтезу інформації в предметній області комп'ютерних наук.
ЗК7. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.	
ЗК9. Здатність працювати в команді.	
СК3. Здатність до логічного мислення, побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проектування, розроблення й аналізу алгоритмів, оцінювання їх ефективності та складності, розв'язності та нерозв'язності	ПР5. Проектувати, розробляти та аналізувати алгоритми розв'язання обчислювальних та логічних задач, оцінювати ефективність та складність алгоритмів на основі застосування формальних моделей алгоритмів та обчислюваних функцій.

Компетентності	Результати навчання
алгоритмічних проблем для адекватного моделювання предметних областей і створення програмних та інформаційних систем.	
СК8. 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.	ПР10. Використовувати інструментальні засоби розробки клієнт-серверних застосувань, проектувати концептуальні, логічні та фізичні моделі баз даних, розробляти та оптимізувати запити до них із застосуванням мов веб-програмування.
СК12. Здатність забезпечити організацію обчислювальних процесів в інформаційних системах різного призначення з урахуванням архітектури, конфігурування, показників результативності функціонування операційних систем і системного програмного забезпечення.	ПР13. 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

2. Програма навчальної дисципліни

Змістовний модуль 1. Програмування в командній оболонці

Тема 1. Архітектура операційної системи Linux

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

1.2. *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.

Тема 2. Файлові системи Linux

2.1. *Поняття файлу і файлової системи.* Організація інформації у файловій системі. Зв'язки, імені та атрибутів файлів. Операції над файлами і каталогами.

2.2. *Фізична організація файлової системи.* Базові відомості про дискові пристрої. Розміщення інформації у файлових системах. Надійність та продуктивність файлових систем.

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

2.4. *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.

Тема 3. Командний рядок Linux

3.1. *Поняття командної оболонки.* Огляд командних оболонок Linux. Командна оболонка bash. Особливості роботи. Доступ до командного рядку.

3.2. *Команди командного рядку.* Символи командного рядку. Перевірка умов. Операції. Цикли та розгалуження. Внутрішні та зовнішні команди. Команди системного адміністрування. Підстановка команд.

3.3. *Багатозадачність в консолі.* Завдання. Управління завданнями. Змінні середовища. Midnight commander.

Тема 4. Сценарії командного рядку

4.1. *Основи роботи зі сценаріями.* Основи складання сценаріїв. Використання структурованих команд. Обробка введення даних користувачем. Подання даних. Управління сценаріями.

4.2. *Удосконалені сценарії командного інтерпретатора.* Створення функцій. Написання сценаріїв для графічних робочих столів. Загальні відомості про редактори sed та gawk. Робота з іншими командними інтерпретаторами.

Змістовний модуль 2. Основи програмування в Linux

Тема 5. Створення, компіляція та компоновка програм в Linux

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

5.2. *Автоматизація.* Огляд засобів автоматизації в Linux. Утиліта make. Базовий синтаксис Makefile. Константи make. Рекурсивний виклик make. Отримання додаткової інформації.

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

Тема 6. Використання бібліотек та опцій при виклику програм

6.1. *Бібліотеки.* Бібліотеки та заголовки. Підключення бібліотек. Створення статичних бібліотек. Створення спільно використовуваних бібліотек. Взаємодія бібліотек.

6.2. *Аргументи і опції програми.* Аргументи програми. Використання опцій. Використання довгих опцій. Отримання додаткової інформації.

Тема 7. Основи введення-виведення в Linux

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

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

7.3. *Розширені можливості введення-виведення в Linux.* Взаємодія з бібліотечними механізмами. Векторне читання: readv(). Векторна запис: writev(). Концепція "чорних дірок".

Змістовний модуль 3. Багатозадачність та міжпроцесна взаємодія

Тема 8. Основи багатозадачності в Linux

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

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

Тема 9. Потoki в Linux

9.1. *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.

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

Тема 10. Методи міжпроцесної взаємодії в Linux

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

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

10.3. *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

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

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

10.6. *Сокети*. Типи сокетів. Створення та видалення сокетів. Призначення адреси: bind(). З'єднання сокетів: connect(). Прослуховування сокета: listen(). Прийняття запрошення на підключення: accept(). Прийом і передача даних через сокети.

Перелік лабораторних занять, а також питань та завдань до самостійної роботи наведено у таблиці "Рейтинг-план навчальної дисципліни".

3. Методи навчання і викладання

Методи навчання, спрямовані на активізації та стимулювання навчально-пізнавальної діяльності здобувачів вищої освіти. При викладанні навчальної дисципліни для активізації навчального процесу передбачено застосування сучасних навчальних технологій, таких, як: проблемні лекції; міні-лекції; робота в малих групах; презентації; ділові та рольові ігри; кейс-метод.

Проблемні лекції спрямовані на розвиток логічного мислення студентів. Коло питань теми лекції обмежується двома-трьома ключовими моментами, увага студентів концентрується на матеріалі, що не знайшов широкого відображення в підручниках, використовується досвід закордонних навчальних закладів з виділенням головних висновків з питань, що розглядаються. Вони передбачають поряд із розглядом основного лекційного матеріалу встановлення та розгляд кола проблемних питань дискусійного характеру, які недостатньо розроблені в науці й мають актуальне значення для теорії та практики. Лекції проблемного характеру відрізняються поглибленою аргументацією матеріалу, що викладається. При викладанні лекційного матеріалу студентам пропонуються питання для самостійного розмірковування. При цьому лектор задає питання, які спонукають студента шукати розв'язання проблемної ситуації. Така система примушує студентів сконцентруватися і почати активно мислити в пошуках правильної відповіді. Проблемні лекції сприяють формуванню у студентів самостійного творчого мислення, прищеплюють їм пізнавальні навички. Студенти стають учасниками наукового пошуку та вирішення проблемних ситуацій.

На початку проведення проблемної лекції потрібно чітко сформулювати проблему, яку необхідно вирішити студентам. При викладанні лекційного матеріалу слід уникати прямої відповіді на поставлені запитання, а висвітлювати лекційний матеріал таким чином, щоб отриману інформацію студент міг використовувати при розв'язанні проблеми.

Міні-лекції передбачають викладення навчального матеріалу за короткий проміжок часу й характеризуються значною ємністю, складністю логічних побудов, образів, доказів та узагальнень. Вони проводяться, як правило, як частина заняття-дослідження. На початку проведення міні-лекції за вказаними темами лектор акцентує увагу студентів на необхідності представити викладений лекційний матеріал у так званому структурно-логічному вигляді. На розгляд лекції виносяться питання, які зафіксовані в плані лекції, але викладаються стисло.

Лекційне заняття проведене у такий спосіб, пробуджує у студента активність та увагу при сприйнятті матеріалу, а також спрямовує його на використання системного підходу при відтворенні інформації, яку він отримав від викладача.

Проблемні лекції та міні-лекції доцільно поєднувати з такою формою активізації навчального процесу, як робота в малих групах.

Робота в малих групах дає змогу структурувати лекційні або лабораторні заняття за формою і змістом, створює можливості для участі кожного студента в роботі за темою заняття, забезпечує формування особистісних якостей та досвіду соціального спілкування. Після висвітлення проблеми (при використанні проблемних лекцій) або стислого викладання матеріалу (при використанні міні-лекцій) студентам пропонується об'єднуватися у групи по 5-6 осіб та презентувати наприкінці заняття своє бачення та сприйняття матеріалу.

Презентації – виступи перед аудиторією, що використовуються для представлення певних досягнень, результатів роботи групи, звіту про виконання індивідуальних завдань. Однією з позитивних рис презентації та її переваг при використанні в навчальному процесі є обмін досвідом, який здобули студенти при роботі у певній малій групі.

Лабораторні заняття (з елементами семінарської дискусії) дозволяють формувати у студентів навички практичної реалізації практичних задач шляхом розробки алгоритмічного забезпечення та побудови на основі останнього відповідного програмного забезпечення з використанням структурного та базових принципів процедурно-орієнтовного програмування, узагальнювати отримані результати, формулювати висновки та думки, вести подальший обмін думками та поглядами з іншими учасниками щодо отриманих результатів досліджень із заданої проблематики, а також розвивають творче мислення, допомагають формувати погляди і переконання, вчать об'єктивно оцінювати результати і пропозиції опонентів, критично підходити до власних результатів та поглядів.

Ділові та рольові ігри – форма активізації студентів, за якої вони задіяні в процесі інсценізації певної виробничої ситуації у ролі безпосередніх учасників подій. Наприклад, при проведенні лабораторного заняття студенти можуть бути розділені на групи, кожна з яких отримує від викладача певне завдання, реалізація якого повинна бути виконана шляхом використання різних підходів.

Кейс-метод — метод аналізу конкретних ситуацій, який дає змогу наблизити процес навчання до реальної практичної діяльності спеціалістів і передбачає розгляд виробничих, управлінських та інших ситуацій, складних конфліктних випадків, проблемних ситуацій, інцидентів у процесі вивчення навчального матеріалу.

4. Порядок оцінювання результатів навчання

Система оцінювання сформованих компетентностей у студентів враховує види занять, які згідно з програмою навчальної дисципліни передбачають лекційні та лабораторні заняття, а також виконання самостійної роботи. Оцінювання результатів вивчення навчальної дисципліни здійснюється за накопичувальною (100-бальною) системою оцінювання.

Оцінювання здійснюється за такими видами контролю:

поточний контроль, що здійснюється протягом семестру під час проведення лекційних, лабораторних занять і оцінюється сумою набраних балів (максимальна сума – 100 балів; мінімальна сума, що дозволяє студенту отримати залік – 60 балів);

підсумковий/семестровий контроль, що проводиться у формі семестрового заліку, відповідно до графіку навчального процесу.

Залік виставляється як загальна сума балів, набраних за результатами поточного контролю.

Порядок проведення поточного оцінювання знань студентів. Поточний контроль включає оцінювання студентів під час:

Лекцій – активна робота на парі (2 бали за кожне заняття) за умови надання студентом при проведенні опитування правильних відповідей на контрольні запитання, участі в

обговоренні результатів вирішення поставленої на початку заняття проблемної ситуації. Загальна кількість балів складає 20 балів.

Контрольних робіт – передбачає виявлення опанування студентом матеріалу лекційного модуля та вміння застосовувати його для вирішення практичних ситуацій. Проводиться під час лекційних аудиторних занять тестовим методом. На протязі семестру передбачено 2 контрольні роботи (10 балів за кожне заняття). Перша контрольна робота включає теми 1 – 4, а друга – теми 5 – 10. Загальна кількість балів складає 20 балів.

Лабораторних робіт – має на меті перевірку рівня підготовленості студента до виконання конкретної роботи. Оцінювання передбачає захист звітів з лабораторної роботи (5 балів за кожний звіт), за умови відповідності рівня знань студента критеріям, що висуваються. Загальна кількість балів складає 50 балів.

Оцінювання знань студента під час лабораторних занять проводиться за такими критеріями:

розуміння, ступінь засвоєння теорії та методології проблем, що розглядаються;

ступінь засвоєння фактичного матеріалу навчальної дисципліни;

ознайомлення з рекомендованою літературою, а також із сучасною літературою з питань, що розглядаються;

вміння поєднувати теорію з практикою при розгляді виробничих ситуацій, розв'язанні задач, проведенні розрахунків у процесі виконання індивідуальних завдань та завдань, винесених для самостійного опрацювання, а також завдань, винесених на розгляд аудиторії;

логіка, структура, стиль викладу матеріалу в письмових роботах і при виступах в аудиторії, вміння обґрунтовувати свою позицію, здійснювати узагальнення інформації та робити висновки;

Оцінка на максимальну кількість балів ставиться за умови відповідності виконаного завдання студента або його усної відповіді до всіх п'яти зазначених критеріїв. Відсутність тієї чи іншої складової знижує оцінку на відповідну кількість балів.

При оцінюванні практичних завдань увага приділяється також їх якості та самостійності, своєчасності здачі виконаних завдань викладачу (згідно з графіком навчального процесу). Якщо будь-яка із вимог не буде виконана, то оцінка буде знижена.

Самостійна робота здобувача може включати:

опрацювання теоретичних основ прослуханого лекційного матеріалу;

вивчення окремих тем або питань, що передбачені для самостійного опрацювання;

виконання домашніх завдань;

підготовка до практичних (лабораторних) занять;

підготовка до контрольних робіт та інших форм поточного контролю;

підготовка до захисту індивідуальних робіт;

аналіз конкретної виробничої ситуації;

пошук (підбір) джерел для підготовки презентацій за заданою тематикою;

виконання індивідуальних завдань з використанням програмного забезпечення тощо.

Загальними критеріями, за якими здійснюється оцінювання самостійної роботи студентів, є: глибина і міцність знань, рівень мислення, вміння систематизувати знання за окремими темами, вміння робити обґрунтовані висновки, володіння категорійним апаратом, навички і прийоми виконання практичних завдань, вміння знаходити необхідну інформацію, здійснювати її систематизацію та обробку, самореалізація на лекційних та лабораторних заняттях.

За умови правильного виконання домашнього завдання, наявності відповіді, оформлення висновків та звіту. Кожне домашнє завдання оцінюється в 2 бали при відповідності вказаним критеріям. За відсутності відповіді або за наявності помилок оцінка знижується на 1 бал. Кількість домашніх завдань – 5. Максимальна кількість за всі домашні завдання становить – 10 балів.

Підсумковий контроль знань та компетентностей студентів з навчальної дисципліни здійснюється на підставі отриманих балів за результатами поточного контролю за змістовними модулями дисципліни. Підсумкова оцінка з навчальної дисципліни розраховується з

урахуванням балів, отриманих під час поточного контролю за накопичувальною системою. Сумарний результат у балах за семестр складає "60 і більше балів – зараховано", "59 і менше балів – не зараховано" та заноситься у залікову "Відомість обліку успішності" навчальної дисципліни.

Студента слід вважати атестованим, як під час поточного контролю, так і під час підсумкового, якщо сума балів, одержаних за результатами підсумкової перевірки успішності, дорівнює або перевищує 60 балів.

Виставлення підсумкової оцінки здійснюється за шкалою, наведено в таблиці "Шкала оцінювання: національна та ЄКТС".

Форми оцінювання та розподіл балів наведено у таблиці "Рейтинг-план навчальної дисципліни".

Шкала оцінювання: національна та ЄКТС

Сума балів за всі види навчальної діяльності	Оцінка ЄКТС	Оцінка за національною шкалою	
		для екзамену, курсового проекту (роботи), практики	для заліку
90 – 100	A	відмінно	зараховано
82 – 89	B	добре	
74 – 81	C		
64 – 73	D		
60 – 63	E	задовільно	не зараховано
35 – 59	FX	незадовільно	
1 – 34	F		

5. Рейтинг-план навчальної дисципліни

Тема	Форми та види навчання		Форми оцінювання	Мак бал
Тема 1. Архітектура операційної системи Linux	<i>Аудиторна робота</i>			
	Лекція	Лекція за питаннями: 1.1. Загальні відомості про Linux. 1.2. Архітектура операційної системи Linux	Активна робота на парі	2
	Лабораторне заняття	Лабораторна робота № 1. Дослідження базових команд для CLI (робота в малих групах)	Захист звіту з лабораторної роботи	5
	<i>Самостійна робота</i>			
	Питання та завдання до самостійного опрацювання	Опрацювання лекційного матеріалу. Виконання домашнього завдання по встановлення операційної системи Linux Ubuntu на віртуальну машину. Підготовка до лабораторного заняття	Перевірка домашнього завдання	2
Тема 2. Файлові системи Linux	<i>Аудиторна робота</i>			
	Лекція	Проблемна лекція на тему 2.1 Файлові системи Linux	Активна робота на парі	2
	Лабораторне заняття	Лабораторна робота № 2. Дослідження властивостей файлової структури Linux (робота в малих групах)	Захист звіту з лабораторної роботи	5

Тема	Форми та види навчання		Форми оцінювання	Мак бал
	Самостійна робота			
	Питання та завдання до самостійного опрацювання	Опрацювання лекційного матеріалу. Підготовка до лабораторного заняття	Експрес-опитування за темою лабораторної роботи	
Тема 3. Командний рядок Linux	Аудиторна робота			
	Лекція	Лекція за питаннями: 3.1. Поняття командної оболонки. 3.2. Команди командного рядку. 3.3. Багатозадачність в консолі.	Активна робота на парі	2
	Лабораторне заняття	Лабораторна робота № 3. Дослідження скриптової мови CLI	Захист звіту з лабораторної роботи	5
	Самостійна робота			
	Питання та завдання до самостійного опрацювання	Опрацювання лекційного матеріалу. Виконання домашнього завдання по вивченню форматів команд CLI. Презентація студентом результатів виконання домашнього завдання Підготовка до лабораторного заняття	Перевірка домашнього завдання	2
Тема 4. Сценарії командного рядку	Аудиторна робота			
	Лекція	Проблемна лекція на тему 4.1. Сценарії командного рядку	Активна робота на парі	2
			Письмова контрольна робота	10
	Лабораторне заняття	Лабораторна робота № 4. Дослідження можливостей shell-програмування (заняття проводиться методом ділової гри)	Захист звіту з лабораторної роботи	5
Самостійна робота				
	Питання та завдання до самостійного опрацювання	Опрацювання лекційного матеріалу. Підготовка до лабораторного заняття Підготовка до контрольної роботи.	Експрес-опитування за темою лабораторної роботи	
Тема 5. Створення, компіляція та компоновка програм в Linux	Аудиторна робота			
	Лекція	Лекція за питаннями: 5.1. Створення програм в Linux. 5.2. Автозбирання. 5.3. Оточення	Активна робота на парі	2
	Лабораторне заняття	Лабораторна робота № 5. Дослідження способів створення програм в Linux (заняття проводиться кейс-методом)	Захист звіту з лабораторної роботи	5
	Самостійна робота			
	Питання та завдання до самостійного опрацювання	Опрацювання лекційного матеріалу. Підготовка до лабораторного заняття	Експрес-опитування за темою лабораторної роботи	

Тема	Форми та види навчання		Форми оцінювання	Мак бал
Тема 6. Використанні бібліотек та опцій при	<i>Аудиторна робота</i>			
	Лекція	Проблемна лекція на тему 6.1. Використанні бібліотек та опцій при виклику програм	Активна робота на парі	2
	Лабораторне заняття	Лабораторна робота № 6. Дослідження операцій з бібліотеками в Linux	Захист звіту з лабораторної роботи	5
	<i>Самостійна робота</i>			
Питання та завдання до самостійного опрацювання	Опрацювання лекційного матеріалу. Виконання домашнього завдання по створенню бібліотек вирішення заданої функції. Підготовка до лабораторного заняття	Перевірка домашнього завдання	2	
Тема 7. Основи введення-виведення в Linux	<i>Аудиторна робота</i>			
	Лекція	Лекція за питаннями: 7.1. Концепція введення-виведення в Linux. 7.2. Базові операції введення-виведення. 7.3. Розширені можливості введення-виведення в Linux.	Активна робота на парі	2
	Лабораторне заняття	Лабораторна робота № 7 Дослідження можливостей введення-виведення в Linux	Захист звіту з лабораторної роботи	5
	<i>Самостійна робота</i>			
Питання та завдання до самостійного опрацювання	Опрацювання лекційного матеріалу. Підготовка до лабораторного заняття	Експрес-опитування за темою лабораторної роботи		
Тема 8. Основи багатозадачності в Linux	<i>Аудиторна робота</i>			
	Лекція	Лекція за питаннями: 8.1. Основи багатозадачності в Linux. 8.2. Базова багатозадачність.	Активна робота на парі	2
	Лабораторне заняття	Лабораторна робота № 8. Дослідження способів багатозадачного програмування в Linux	Захист звіту з лабораторної роботи	5
	<i>Самостійна робота</i>			
Питання та завдання до самостійного опрацювання	Опрацювання лекційного матеріалу. Виконання домашнього завдання по керуванню обліковими записами та правами доступу. Підготовка до лабораторного заняття	Перевірка домашнього завдання	2	
Тема 9. Потоки в Linux	<i>Аудиторна робота</i>			
	Лекція	Проблемна лекція на тему 9.1. Потоки в Linux	Активна робота на парі	2
	Лабораторне заняття	Лабораторна робота № 9. Управління процесами в Linux	Захист звіту з лабораторної роботи	5
	<i>Самостійна робота</i>			
Питання та завдання до самостійного опрацювання	Опрацювання лекційного матеріалу. Виконання домашнього завдання по створенню процесів у Linux із застосуванням системних викликів fork() і exec(). Підготовка до лабораторного заняття	Перевірка домашнього завдання	2	

Тема	Форми та види навчання		Форми оцінювання	Мак бал
Тема 10. Методи міжпроцесної взаємодії в Linux	<i>Аудиторна робота</i>			
	Лекція	Міні-лекція за питаннями: 10.1. Сигнали. 10.2. Використання загальної пам'яті. 10.3. Використання загальних файлів. 10.4. Канали. 10.5. Сокети.	Активна робота на парі	2
			Письмова контрольна робота	10
	Лабораторне заняття	Лабораторна робота № 10. Дослідження механізмів взаємодії між процесами в Linux (робота в малих групах)	Захист звіту з лабораторної роботи	5
	<i>Самостійна робота</i>			
Питання та завдання до самостійного опрацювання	Опрацювання лекційного матеріалу. Підготовка до лабораторного заняття. Підготовка до контрольної роботи.	Експрес-опитування		

6. Рекомендована література

6.1. Основна

1. Иванов Н.Н. Программирование в Linux. Самоучитель / Н.Н. Иванов. – СПб.: БХВ-Петербург, 2012. – 400 с.
2. Донцов В.П. Linux на примерах / В.П. Донцов, I.B. Сафин. – СПб.: Наука та техніка, 2017. – 352 с.
3. Блум Р. Командная строка Linux и сценарии оболочки. Библия пользователя / Р. Блум, К. Бреснахэн: пер. с англ. – М.: ООО “И.Д. Вильямс”, 2012. – 784 с.
4. Лав Р. Linux. Системное программирование / Р. Лав. – СПб.: Питер, 2014. – 448 с.
5. Костромин В. А. Самоучитель Linux для пользователя / В.А. Костромин. – СПб.: БХВ-Петербург, 2003. – 672 с.
6. Дейв Т. Сценарии командной оболочки. Linux, OS X і Unix. / Т.Дейв, П. Брейдон, - СПб.: Питер, 2017. – 448 с.
7. Волох С.М. Ubuntu Linux с нуля /С.М. Волох. – СПб.: БХВ-Петербург, 2018. – 400 с.
8. Шоттс У. Командная строка Linux. Полное руководство / У.Шоттс. – СПб.: Питер, 2017. – 480 с.

6.2. Додаткова

9. Прата С. Язык программирования С. Лекции и упражнения / С. Прата: пер. с англ. – М.: Вильямс, 2015. – 928 с.
10. Буленок В.Г. Основы работы в среде Linux: Учебное пособие / В.Г. Буленок, Е.Г. Пьяных. – Томск: ТГПУ. 2008. – 40 с.
11. Навчальний курс Ubuntu Desktop Training. – NC SA USA, 2019. – 410 с.
12. Купер М. Искусство программирования на языке сценариев командной оболочки / М. Купер. – М.: Вильямс, 2011. – 746 с.
13. Керриск М. Linux API. Исчерпывающее руководство / М. Керриск. – СПб.: Питер, 2018. – 1248 с.
14. Hud O. The Project Management Information System in Linux / O. Hud, O.Veres //COLINS'2020, Volume II: Workshop. – Lviv: Ukraine, 2020. – Pp. 270 – 273.
15. Пташкін Р.Л. Використання операційних систем Linux в експертній практиці / Р.Л. Пташкін // Теорія і практика судової експертизи і криміналістики: матеріали III-ої Всеукраїнської науково-практичної конференції (Київ, 27 лютого 2020 року). – Київ-Маріуполь, 2020. – С. 243 – 246.

16. Таненбаум Э., Бос Х. Современные операционные системы. – СПб.: Питер, 2015. – 1120 с.

6.3. Інформаційні ресурси

17. Linux. Виртуальная энциклопедия [Электронный ресурс]. – Режим доступа: <http://rus-linux.net/>.
18. Linux для программиста. Тонкие вопросы программирования и администрирования Linux. [Электронный ресурс]. – Режим доступа: <https://mylinuxprog.blogspot.com/>.
19. Программирование в Linux [Электронный ресурс]. – Режим доступа: <http://www.linuxlib.ru/prog/>.
20. Персональна навчальна система "Програмування для Linux" [Електронний ресурс]. – Режим доступу: <https://pns.hneu.edu.ua/course/view.php?id=6814>