

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Львівський національний університет імені Івана Франка
Факультет прикладної математики та інформатики
Кафедра програмування

Затверджено

На засіданні кафедри програмування
факультету прикладної математики
Львівського національного університету
імені Івана Франка
(протокол № 1 від 29 серпня 2025 р.)



Зав. кафедри к. ф.-м. н., доц. Ярошко С. А.

Силабус навчальної дисципліни
«Програмування під UNIX-подібними системами»,
що викладається в межах ОПП «Середня освіта (Інформатика)»
першого (бакалаврського) рівня вищої освіти
для здобувачів з спеціальності А4.09 Середня освіта (Інформатика)

Назва дисципліни	Програмування під UNIX-подібними системами
Адреса викладання дисципліни	Львівський національний університет імені Івана Франка, вул. Університетська 1, м. Львів, Україна, 79000
Факультет та кафедра, за якою закріплена дисципліна	Факультет прикладної математики та інформатики, кафедра програмування
Галузь знань, шифр та назва спеціальності	Галузь знань: А Освіта / Педагогіка, Спеціальність: А4 Середня освіта Спеціалізація: А4.09 Середня освіта (Інформатика)
Викладачі дисципліни	Літинський Святослав Володимирович, к. ф.-м. н., доцент, доцент кафедри програмування
Контактна інформація викладачів	Електронна пошта: svyatoslav.litynskyi@lnu.edu.ua веб-сторінка: https://ami.lnu.edu.ua/employee/litynskyi
Консультації з питань навчання по дисципліні відбуваються	Консультації проводять раз на тиждень згідно з оприлюдненим розкладом консультацій викладача. Можливі он-лайн консультації через Microsoft Teams. Для погодження часу он-лайн консультацій слід писати у Microsoft Teams викладачу.
Сторінка курсу	https://ami.lnu.edu.ua/course/prohramuvannia-pid-unix-podibnymy-systemamy-co
Інформація про дисципліну	Курс “Програмування під UNIX-подібними системами” є вибірковою дисципліною зі спеціальності Середня освіта (Інформатика) для освітньої програми «Середня освіта (Інформатика)», яка викладається в 5-му семестрі в обсязі 4 кредитів (за Європейською Кредитно-Трансферною Системою ECTS).
Коротка анотація дисципліни	У курсі розглядаються основи Unix-подібних операційних систем та виконуються завдання різними мовами програмування під ці операційні системи. Теми: файлова система, користувацькі оболонки, базові команди, доступ до файлів, канали, регулярні вирази.
Мета та цілі дисципліни	Метою нормативної дисципліни «Програмування під UNIX-подібними системами» є навчити студента: <ul style="list-style-type: none"> • основам Unix-подібних операційних систем; • застосовувати основні мови програмування для створення програм у прикладних задачах під ОС UNIX.
Література для вивчення дисципліни	<i>Основна література</i> 1. Sumitabha Das. Your UNIX: The Ultimate Guide. McGraw-Hill Science/Engineering/Math. 2005. <i>Додаткова література</i> 2. Arnold Robbins. UNIX in a nutshell. O'Reilly Media. 2005.
Обсяг курсу	4 кредити ЄКТС – 120 годин. З них 32 годинь лекцій, 32 години лабораторних занять та 56 годин самостійної роботи
Очікувані результати навчання	Після завершення цього курсу студент буде: <p><i>знати</i></p> <ul style="list-style-type: none"> • Структуру, організацію доступу до даних та взаємодію між процесами у UNIX-подібних операційних системах. <p><i>вміти</i></p> <ul style="list-style-type: none"> • Самостійно налаштувати та писати скрипти та програми під UNIX-подібні операційні системи.
Компетентності	<i>Загальні:</i> Здатність застосовувати знання у практичних ситуаціях. Знання та розуміння предметної області та розуміння професійної діяльності. <i>Фахові компетентності спеціальності:</i> 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.
Програмні результати навчання	Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук.
Ключові слова	Команди Linux, сценарії оболонки, файлова система Linux, взаємодія процесів, управління пакетами, безпека файлової системи Linux, архітектура Linux.
Формат курсу	Очний: проведення лекцій, лабораторних робіт та консультацій в приміщеннях університету, онлайнвий на платформі Microsoft Teams
Теми	<p>Теоретичні теми</p> <ul style="list-style-type: none"> ● Файлова система, inodes, папки ● Оболонки ● Основні команди UNIX (маніпуляції файлами каталогу) ● Права доступу до файлів ● Посилання на файли ● C Shell або bash ● Конвеєр виконання ● Перенаправлення ● Псевдоніми ● Керуючі структури в bash або tssh ● Змінні в bash або tssh ● Вбудовані змінні оболонки ● Qt Signal-Slot ● Події Qt ● База даних Qt ● Python ● Perl ● Java ● .net Core під Linux, збірки ● Регулярні вирази <p>Практичні</p> <p>Реалізація програм у UNIX-подібних ОС на задану тему використовуючи</p> <ul style="list-style-type: none"> ● Bash або tssh ● gcc ● QT + visual ● python ● perl ● java console ● java visual ● .net Core
Підсумковий контроль, форма	залік
Пререквізити	Для вивчення курсу студенти потребують базових знань з дисциплін “Математичний аналіз”, «Алгоритми та структури даних», “Інформатика” за перший семестр курсу “Програмування”. Дисципліна “Програмування під UNIX-подібними системами” також тісно пов'язана з курсом “Програмування”, який вивчають у першому та другому семестрах.
Навчальні методи та техніки, які використовують під час викладання курсу	Лекції з мультимедійними презентаціями та з демонстрацією прийомів практичного використання середовища програмування; лабораторні заняття - виконання практичних завдань; самостійне опрацювання навчальних матеріалів: підручників. Обговорення теоретичного та практичного матеріалу в онлайн сервісах, формулювання творчих завдань для студентів, виконання яких готує до вивчення нового теоретичного матеріалу.
Необхідне обладнання	Для проведення лекцій: комп'ютер, проектор, доступ до мережі інтернет. Для проведення лабораторних та виконання завдань: комп'ютер, ОС Linux, доступ до інтернету. Уся література, яку студенти не можуть знайти самостійно, буде надана викладачем виключно в освітніх цілях без права її передачі третім особам.

	Студенти заохочуються до використання також й іншої літератури та джерел, яких немає серед рекомендованих.
Критерії оцінювання (окремо для кожного виду навчальної діяльності)	<p>Оцінювання проводиться за 100-бальною шкалою. 100 балів нараховують за виконання лабораторних завдань (70 балів) та відповіді на запитання у формі тесту (10) та у формі розповіді/пояснення (20). Лабораторні завдання є індивідуальними. Упродовж семестру студент виконує 7 лабораторних робіт, кожен з яких оцінюють в 10 балів.</p> <p>Для кожного завдання визначено термін виконання: зазвичай до закінчення навчального тижня. Вчасно виконані завдання оцінюють так (у відсотках від максимальної оцінки):</p> <ul style="list-style-type: none"> • 100% – умови завдання виконано повністю, алгоритми складено правильно, програма містить належні коментарі, роботу програми перевірено на достатньому наборі тестових даних, автор відповідає на всі запитання щодо використаних підходів, чітко інтерпретує отримані результати, немає ознак недоброчесності; • 80% – наведено логічно правильну послідовність розв'язування, алгоритми складено правильно, бракує окремих коментарів чи тестів, автор не досить повно пояснює використані підходи, немає ознак недоброчесності; • 60% – у правильній послідовності розв'язування допущено окремі помилки, які автор уміє виправити після зауваження викладача, бракує коментарів чи тестів, на запитання щодо використаних підходів автор відповідає з помилками, немає ознак недоброчесності; • 40% – у правильній послідовності розв'язування пропущено окремі етапи, завдання виконано частково, автор не розуміє недоліків поданої роботи, не вміє їх виправити, немає ознак недоброчесності; • 20% – завдання виконано частково, немає тестів, програма працює правильно для окремих наборів вхідних даних, автор не може самостійно інтерпретувати отримані результати, виправити помилки, немає ознак недоброчесності; • 0% – завдання не виконано, написана програма не відповідає умові, або ж виявлено ознаки недоброчесності: запозичення, фрагменти коду, дію яких автор пояснити не може, автор не володіє відповідним теоретичним матеріалом тощо; <p>Відвідування занять є важливою складовою навчання. Очікується, що всі студенти відвідують усі лекції і лабораторні заняття курсу. Активність під час проведення лекцій і лабораторних заохочується балами. У будь-якому випадку студенти зобов'язані дотримуватися усіх строків визначених для виконання всіх видів робіт, передбачених курсом. Виконані роботи складаються викладачу. Альтернативою відвідування лабораторних занять в університеті може бути дистанційна онлайн робота за розкладом проведення занять. Активність на лекціях і лабораторних враховують при оцінюванні відповідного лабораторного завдання.</p> <p>Академічна доброчесність: очікується, що роботи студентів будуть їхнім оригінальними дослідженнями чи міркуваннями. Відсутність посилань на використані джерела, фабрикування джерел, списування, втручання в роботу інших студентів, здавання чужих комп'ютерних програм як своїх становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в письмовій роботі студента є підставою для її не зарахування викладачем, незалежно від масштабів плагіату чи обману.</p>
Опитування	Анкету-оцінку з метою оцінювання якості курсу буде надано після завершення курсу.

Схема курсу.

Теми лекцій та лабораторних занять

Тиждень	Тема, план, короткі тези	Форма заняття	Тривалість, год
1	Файлова система, inodes, папки	Лекція	1
	Реалізація програм у UNIX-подібних ОС на задану тему використовуючи Bash	Лабораторна робота	2

2	Оболонки Основні команди UNIX (маніпуляції файлами каталогу)	Лекція	1
	Реалізація програм у UNIX-подібних ОС на задану тему використовуючи Bash	Лабораторна робота	2
3	Права доступу до файлів Посилання на файли	Лекція	1
	Реалізація програм у UNIX-подібних ОС на задану тему використовуючи gcc	Лабораторна робота	2
4	C Shell, bash	Лекція	1
	Реалізація програм у UNIX-подібних ОС на задану тему використовуючи gcc	Лабораторна робота	2
5	Конвеєр виконання Перенаправлення	Лекція	1
	Реалізація програм у UNIX-подібних ОС на задану тему використовуючи QT	Лабораторна робота	2
6	Псевдоніми Керуючі структури в bash, tsch	Лекція	1
	Реалізація програм у UNIX-подібних ОС на задану тему використовуючи QT	Лабораторна робота	2
7	Змінні в bash Вбудовані змінні оболонки	Лекція	1
	Реалізація програм у UNIX-подібних ОС на задану тему використовуючи python	Лабораторна робота	2
8	Qt Signal-Slot	Лекція	1
	Реалізація програм у UNIX-подібних ОС на задану тему використовуючи python	Лабораторна робота	2
9	Qt Signal-Slot	Лекція	1
	Реалізація програм у UNIX-подібних ОС на задану тему використовуючи perl	Лабораторна робота	2
10	Події Qt	Лекція	1
	Реалізація програм у UNIX-подібних ОС на задану тему використовуючи perl	Лабораторна робота	2
11	База даних Qt	Лекція	1
	Реалізація програм у UNIX-подібних ОС на задану тему використовуючи	Лабораторна робота	2
12	Python	Лекція	1
	Реалізація програм у UNIX-подібних ОС на задану тему використовуючи java console	Лабораторна робота	2
13	Perl	Лекція	1
	Реалізація програм у UNIX-подібних ОС на задану тему використовуючи java console	Лабораторна робота	2
14	Java	Лекція	1
	Реалізація програм у UNIX-подібних ОС на задану тему використовуючи java visual	Лабораторна робота	2
15	.net Core під Linux, збірки	Лекція	1

	Реалізація програм у UNIX-подібних ОС на задану тему використовуючи java visual.	Лабораторна робота	2
16	Регулярні вирази	Лекція	1
	Реалізація програм у UNIX-подібних ОС на задану тему використовуючи.net Core	Лабораторна робота	2

Індивідуальні завдання.

№	Індивідуальне завдання	Форма заняття	Тривалість, год	Термін виконання, тиждень
1	Підготовка доповіді-презентації на тему згідно переліку тем лабораторних занять. Максимальна оцінка – 5 балів.	Самостійна робота	2	2
2	Підготовка доповіді-презентації на додаткову тему у UNIX. Максимальна оцінка – 5 балів.	Самостійна робота	18	тиждень, на який запланована обрана тема
3	Створення «доброї» статті в україномовному розділі Вікіпедії довільної тематики. Максимальна оцінка – 5 балів (додатковий бал за статтю з галузі математики чи інформаційних технологій)	Самостійна робота	18	16
4	Написання індивідуальних програм	Самостійна робота	34	тиждень, на який запланована обрана тема