НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені ІГОРЯ СІКОРСЬКОГО»
ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ
КАФЕДРА ПРИКЛАДНОЇ МАТЕМАТИКИ

ЗАТВЕРДЖЕНО

Методичною радою
КПІ ім. Ігоря Сікорського
(протокол № 5 від «29» лютого 2024 р.)

Фаховий КАТАЛОГ

вибіркових навчальних дисциплін
першого (бакалаврського) рівня вищої освіти
для освітньої програми
«Наука про дані та математичне моделювання»
спеціальності 113 Прикладна математика
(на 2024-2025 навчальний рік)

УХВАЛЕНО:

Вченою радою ФПМ
КПІ ім. Ігоря Сікорського
(протокол № 7 від «29» січня 2024 р.)

Київ 2024

Зміст

Інструкція користувачам каталогу **4**

Ф-Каталог – 2024 р. **5**

Дисципліни для 2 курсу (вибір першокурсників) **5**

Дисципліни для 3 курсу (вибір другокурсників) **6**

Дисципліни для 4 курсу (вибір третьокурсників) **7**

Анотації вибіркових дисциплін для 2 курсу **8**

1. Об’єктно-орієнтоване програмування 8
2. Програмування мовою Java 9
3. Програмування з використанням Spring Framework 10
4. Функційне програмування 11
5. Структура та інтерпретація комп’ютерних програм 12
6. Програмування мовою Scala 13
7. Основи тестування програмного забезпечення 14
8. Забезпечення якості програмного забезпечення 15
9. Автоматизоване тестування програмного забезпечення 16

Анотації вибіркових дисциплін для 3 курсу **17**

10. Дослідження операцій 17
11. Задачі дискретної оптимізації 18
12. Лінійне програмування та теорія ігор 19
13. Математичне програмування 20
14. Практичні аспекти чисельної оптимізації 21
15. Нелінійне програмування 22
16. Випадкові процеси 23
17. Статистичне моделювання 24
18. Марківські моделі 25
19. Рівняння математичної фізики 26
20. Інтегральні перетворення і диференціальні рівняння 27
21. Класична та сучасна теорії параболічних задач 28
22. Front-end розробка 29
23. Побудова REST-сервісів 30
24. Веб-технології візуалізації даних 31

Анотації вибіркових дисциплін для 4 курсу **32**

25. Системи глибинного навчання 32
26. Обчислювальний інтелект 33
27. Програмування систем штучного інтелекту 34
28. Інформаційні системи 35

29. Забезпечення стійкості інформаційних систем 36
30. Інтеграційні шаблони в інформаційних системах 37
31. Розподілені і хмарні обчислення 38
32. Програмне забезпечення розподілених систем 39
33. Операційні системи 40
34. Навчання з підкріпленням 41
35. Навчання з підкріпленням в глибинному навчанні 42
36. Методи пошуку аномалій 43
37. Системний аналіз 44
38. Основи теорії прийняття рішень 45
39. Основи теорії інформаційних систем 46
40. Теорія оптимального керування 47
41. Теорія автоматичного керування 48
42. Теорія динамічної оптимізації 49

Інструкція користувачам каталогу

1. Кількість і обсяг (у кредитах ЄКТС) навчальних дисциплін, які може обрати студент (вибіркових дисциплін), визначається навчальним планом, а саме: для II курсу – 12 кредитів, III курсу – 20 кредитів, IV курсу – 24 кредити. У навчальному плані зазначається також семестр, у якому викладають вибіркову дисципліну, форма семестрового контролю, види та обсяги навчальних занять.

2. Безпосередній вибір студентами дисциплін здійснюється через інформаційну систему https://my.kpi.ua/.

3. У разі неможливості формування навчальної групи/потоку для вивчення певної дисципліни Фахового каталогу, студентам надається можливість або здійснити повторний вибір – приєднавшись до вже сформованих навчальних груп/потоків, або опановувати обрану дисципліну індивідуально з використанням змішаної форми навчання та індивідуальних консультацій (можливість надається за обґрунтованою заявою студента та рішенням кафедри, яка забезпечує викладання цієї дисципліни).

4. Здобувача, який знехтував своїм правом вибору, може бути записано на вивчення навчальних дисциплін, які обрав завідувач випускової кафедри, для оптимізації навчальних груп/потоків.

5. Якщо здобувач із поважної причини не зміг обрати дисципліни вчасно, або виявив помилку щодо свого волевиявлення, він звертається до деканату з заявою для запису на вивчення обраних ним дисциплін, надавши документі, які засвідчують поважність причин. Заява на зміну вибіркової дисципліни у сформованому індивідуальному навчальному плані має подаватися не пізніше, ніж за місяць до початку семестру, в якому викладають цю дисципліну.

6. Не допускається зміна обраних дисциплін після початку навчального семестру, в якому їх викладають.

7. Навчальні дисципліни, які обрав студент, зазначаються у його індивідуальному навчальному плані.

8. Більше інформації можна знайти в Положенні про порядок реалізації студентами права на вільний вибір навчальних дисциплін.

Ф-Каталог – 2024 р.

Дисципліни для 2 курсу (вибір першокурсників)

(потрібно обрати 12 кредитів)

3-й семестр: одна дисципліна обсягом у 4 кредити

4-й семестр: дві дисципліни обсягом у 4 кредити кожна

№	Назва навчальної дисципліни	Семестр	Кіл-ть кредитів	Семестрова атестація
1	Об’єктно-орієнтоване програмування	3	4	залік
2	Програмування мовою Java	3	4	залік
3	Програмування з використанням Spring Framework	3	4	залік
4	Функційне програмування	4	4	залік
5	Структура та інтерпретація комп’ютерних програм	4	4	залік
6	Програмування мовою Scala	4	4	залік
7	Основи тестування програмного забезпечення	4	4	залік
8	Забезпечення якості програмного забезпечення	4	4	залік
9	Автоматизоване тестування програмного забезпечення	4	4	залік

Дисципліни для 3 курсу (вибір другокурсників)

(потрібно обрати 24 кредити)

5- й семестр: три дисципліни обсягом у 4 кредити кожна

6- й семестр: дві дисципліни обсягом у 4 кредити кожна

№	Назва навчальної дисципліни	Семестр	Кіл-ть кредитів	Семестрова атестація
1	Дослідження операцій	5	4	залік
2	Задачі дискретної оптимізації	5	4	залік
3	Лінійне програмування та теорія ігор	5	4	залік
4	Математичне програмування	5	4	залік
5	Практичні аспекти чисельної оптимізації	5	4	залік
6	Нелінійне програмування	5	4	залік
7	Випадкові процеси	5	4	залік
8	Статистичне моделювання	5	4	залік
9	Марківські моделі	5	4	залік
10	Рівняння математичної фізики	6	4	залік
11	Інтегральні перетворення і диференціальні рівняння	6	4	залік
12	Класична та сучасна теорії параболічних задач	6	4	залік
13	Front-end розробка	6	4	залік
14	Побудова REST-сервісів	6	4	залік
15	Веб-технології візуалізації даних	6	4	залік

Дисципліни для 4 курсу (вибір третьокурсників)

(потрібно обрати 24 кредити)

7-й семестр: чотири дисципліни обсягом у 4 кредити кожна

8-й семестр: дві дисципліни обсягом у 4 кредити кожна

№	Назва навчальної дисципліни	Семестр	Кіл-ть кредитів	Семестрова атестація
1	Системи глибинного навчання	7	4	залік
2	Обчислювальний інтелект	7	4	залік
3	Програмування систем штучного інтелекту	7	4	залік
4	Інформаційні системи	7	4	залік
5	Забезпечення стійкості інформаційних систем	7	4	залік
6	Інтеграційні шаблони в інформаційних системах	7	4	залік
7	Розподілені і хмарні обчислення	7	4	залік
8	Програмне забезпечення розподілених систем	7	4	залік
9	Операційні системи	7	4	залік
10	Навчання з підкріпленням	7	4	залік
11	Навчання з підкріпленням в глибинному навчанні	7	4	залік
12	Алгоритми оптимізації через підкріплення	7	4	залік
13	Системний аналіз	8	4	залік
14	Основи теорії прийняття рішень	8	4	залік
15	Основи теорії інформаційних систем	8	4	залік
16	Теорія оптимального керування	8	4	залік
17	Теорія автоматичного керування	8	4	залік
18	Теорія динамічної оптимізації	8	4	залік

Анотації вибіркових дисциплін для 2 курсу

1. Об’єктно-орієнтоване програмування	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	2, 3
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Розуміння принципів процедурного програмування мовою С. Розуміння класичних алгоритмів та структур даних.
Що буде вивчатися	Головні принципи об’єктно-орієнтованої розробки, SOLID, реалізація багатошарової архітектури.
Чому це цікаво/треба вивчати	ООП — найбільш поширена парадигма програмування у сучасному світі, дана парадигма застосовується при реалізації широкого спектру програм, таких як користувацькі інтерфейси, серверна логіка web-додатків чи математичні моделі в Data Science.
Чому можна навчитися	Ідентифікувати та реалізувати програмно об’єктне представлення предметної області. Ідентифікувати недоліки у об’єктному представленні предметної області з використанням принципів SOLID. Ідентифікувати та реалізовувати найбільш поширені шаблони (патерни) проектування. Розуміти внутрішню структуру представлення об’єктів у мові Java. Реалізовувати програмні продукти мовою Java. Працювати над розробкою програми у команді.
Як можна користуватися набутими знаннями і уміннями	По завершенню курсу, студенти будуть в змозі використовувати свої знання для реалізації проектів мовою Java довільної складності. Фундаментальні знання щодо принципів об’єктно-орієнтованої розробки можуть бути використані для розуміння інших мов програмування.
Інформаційне забезпечення дисципліни	Монографії та онлайн-підручники. Відеолекції. Репозиторій лекційних прикладів і тестових завдань. Онлайн-курси за методикою змішаного навчання.
Вид семестрового контролю	Залік

2. Програмування мовою Java	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	2, 3
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Розуміння принципів процедурного програмування мовою С. Розуміння класичних алгоритмів та структур даних.
Що буде вивчатися	Синтаксис мови Java та принципи роботи у середовищі з автоматичним управлінням пам'яттю. Принципи роботи багатопотокових та web додатків.
Чому це цікаво/треба вивчати	Java — одна з найбільш поширених сучасних мов програмування, що використовується для вирішення широкого кола задач: від програмування вбудованої техніки до створення додатків для мобільних телефонів, до високонавантажених розподілених обчислювальних систем для роботи з BigData, до комерційних проектів для бізнесу.
Чому можна навчитися	Реалізовувати програмні продукти на платформі JVM, використовуючи мову програмування Java. Проводити аналіз предметної області для розробки архітектури майбутньої програми. Ідентифікувати та використовувати шаблони (патерни) проектування. Розуміти принципи роботи середовищ з автоматичним управлінням пам'яттю.
Як можна користуватися набутими знаннями і уміннями	По завершенню курсу, студенти будуть в змозі реалізовувати програмні продукти мовою Java для будь-яких систем, що підтримують середовище JVM, з акцентом на використання у Data Science та web-розробці.
Інформаційне забезпечення дисципліни	Монографії та онлайн-підручники. Відеолекції. Репозиторій лекційних прикладів і тестових завдань. Онлайн-курси за методикою змішаного навчання.
Вид семестрового контролю	Залік

3. Програмування з використанням Spring Framework	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	2, 3
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Розуміння синтаксису та принципів процедурного програмування мовою С.
Що буде вивчатися	Принципи роботи Spring Framework. Розробка корпоративних систем в екосистемі Spring.
Чому це цікаво/треба вивчати	Екосистеми Spring є де-факто стандартом для розробки корпоративних систем на Java. Тому знання цього фреймворку суттєво спростить пошук першої роботи там хто більше зацікавлений розвиватись як софтвер інженер.
Чому можна навчитися	Основні компоненти Spring Framework та принципи його роботи. Кращі практики по розробці та розгортанню корпоративних систем на основі Spring Boot.
Як можна користуватися набутими знаннями і уміннями	По закінченню курсу, студенти будуть у змозі розробляти системи мовою Java з використанням принципів об’єктно-орієнтованого програмування, на основі Spring Framework.
Інформаційне забезпечення дисципліни	Монографії та онлайн-підручники. Відеолекції. Репозиторій лекційних прикладів і тестових завдань. Онлайн-курси за методикою змішаного навчання.
Вид семестрового контролю	Залік

4. Функційне програмування	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	2, 4
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Навички впевненого програмування на Python чи будь-якій іншій імперативній чи об'єктно-орієнтованій мові. Розуміння принципів імперативного програмування.
Що буде вивчатися	Функційний підхід до програмування мовою Haskell; алгебраїчні типи даних та рекурсивні типи; монади та railway-oriented programming; принципи автоматичного доведення теорем.
Чому це цікаво/треба вивчати	Функційне програмування — парадигма програмування, що максимально близька до математики та стрімко набирає популярності у світі. Завдяки гнучкій та потужній системі типів, що дозволяє максимально точно представляти предметну область та ефективним методам маніпуляції даними, що побудовані задля уникнення виникненню проблем у програмі, функційне програмування дозволяє доводити правильність роботи програм під час їх компіляції, дає можливість програмного опису математичних тверджень та автоматичного доведення теорем. Haskell є найбільш характерним представником чистих функційних мов.
Чому можна навчитися	Розробляти програми мовою Haskell, використовуючи усі переваги її системи типів та статичного аналізатору коду. Ідентифікувати проблеми в імперативному коді та застосовувати функційні підходи до їх вирішення. Розуміти принципи роботи компіляторів та систем перевірки типів. Розуміти принципи автоматичного доведення теорем.
Як можна користуватися набутими знаннями і уміннями	По закінченню курсу, студенти будуть у змозі використовувати функційну парадигму програмування та розробляти програмні засоби використовуючи мову Haskell, з акцентом на застосування багатої системи типів, відсутньої в інших популярних мовах програмування.
Інформаційне забезпечення дисципліни	Онлайн-підручники. Монографії з функційного програмування. Репозиторій лекційних прикладів. Онлайн-курси за методикою змішаного навчання. Силабус
Вид семестрового контролю	Залік

5. Структура та інтерпретація комп'ютерних програм	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	2, 4
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Навички програмування на Python чи C. Розуміння принципів і конструкцій, що зустрічаються у комп'ютерних програмах.
Що буде вивчатися	Підходи до структурування та компіляції комп'ютерних програм, принципи мета-циклічного обчислення та уніфікованого представлення даних як коду на прикладі діалекту мови Lisp – Scheme.
Чому це цікаво/треба вивчати	У традиційних мовах програмування ми звикли відділяти дані від коду, який їх обробляє. У цьому курсі ми розглянемо протилежну ідею — представлення даних у вигляді максимально подібному коду програми та принцип само-обчислюваних програм. Ми поглянемо на принципи, що стоять за побудовою інтерпретаторів і компіляторів, та побачимо як традиційні мови запозичують елементи функціональних мов, а також як ідея само-обчислюваних програм впливає на розвиток популярних мов, таких як JavaScript та Python.
Чому можна навчитися	Розуміти принципи побудови компіляторів та інтерпретаторів різних мов програмування з точки зору користувача. Розробляти програми використовуючи принципи мета-циклічного обчислення мовою Lisp та її діалектами. Аналізувати предметну область та розробляти рекурсивні структури даних.
Як можна користуватися набутими знаннями і уміннями	По закінченню курсу, студенти будуть у змозі розробляти програми мовою Lisp та її діалектами, використовуючи принципи мета-циклічного обчислення, та представляти задачу використовуючи структури даних близькі до коду.
Інформаційне забезпечення дисципліни	Онлайн підручник. Слайди лекцій та репозиторій лекційних прикладів. Онлайн-курси за методикою змішаного навчання. Силабус.
Вид семестрового контролю	Залік

6. Програмування мовою Scala	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	2, 4
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Навички роботи з середовищем JVM та мовою програмування Java.
Що буде вивчатися	Синтаксис та принципи розробки програмної архітектури мовою Scala.
Чому це цікаво/треба вивчати	Scala — функціональна мова програмування, що використовує платформу JVM, але на відміну від Java має переваги характерні для функціональних мов: іммутабельність даних, багату систему типів з нативною підтримкою типів-сум і типів-добутків, функції вищих порядків тощо. Вона активно застосовується у задачах, пов'язаних із машинним навчанням та процесингом даних, як заміна для Java.
Чому можна навчитися	Розуміти принципи функціонального програмування. Розробляти застосунки для платформи JVM мовою Scala. Розуміти принципи представлення даних у вигляді типів у системі типів Scala та використання системи перевірки типів для забезпечення валідності програми.
Як можна користуватися набутими знаннями і уміннями	По закінченню курсу, студенти будуть у змозі розробляти програмні засоби для платформи JVM мовою Scala, використовуючи принципи функціонального програмування.
Інформаційне забезпечення дисципліни	Слайди презентацій та онлайн-репозиторій лекційних прикладів. Сайти документації розглянутого програмного забезпечення. Монографії та онлайн-підручники. Онлайн-курси за методикою змішаного навчання. Силабус.
Вид семестрового контролю	Залік

7. Основи тестування програмного забезпечення	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	2, 4
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Тестування програмного забезпечення не потребує спеціальних передумов, але бажаними є базові знання з дисциплін «Програмування», «Алгоритми і структури даних», «Математична логіка»
Що буде вивчатися	Фундаментальні аспекти тестування програмного забезпечення; тестування в контексті життєвих циклів ПЗ (Waterfall, Agile, ітеративні цикли), аналіз вимог до ПЗ, необхідність раннього тестування та ціна виправлення помилок на кожному етапі життєвого циклу; методи статичного тестування: аналіз програмного коду (reviews), аналіз потоку даних; методи «чорного ящика»: розбиття на класи еквівалентності, аналіз граничних значень, тестування таблиць рішень, тестування станів і переходів, тестування пар значень; методи «білого ящика»: тестування виразів, гілок, умов, циклів та інші, рівень покриття тестами програмного коду; тестування, засноване на досвіді; управління тестуванням: рівні тестування, ролі і відповідальності, планування та виконання тестування; артефакти тестування: тест-план, матриця відповідності вимог, тест-кейси та тестові сценарії, звіти про помилки, підсумковий звіт про тестування; метрики оцінки якості тестування.
Чому це цікаво/треба вивчати	Тестування програмного забезпечення стає все більш важливим в IT-галузі, що відображає зростаючу важливість якості програмного забезпечення в сучасному світі. Знання, отримані в цьому курсі, будуть корисними для тих, хто хоче пов’язати свою професійну дальність з розробкою програмного забезпечення, але найбільш цінними вони будуть для тих, хто бажає займатися тестування ПЗ, команд тестувальників та їх керівників.
Чому можна навчитися	Оволодіти термінологією тестування ПЗ, проводити аналіз вимог до ПЗ, розуміти принципи застосування тестування в різних життєвих циклах ПЗ; проводити статичний аналіз програмного коду; розробляти тест-кейси на основі методів «чорного» та «білого» ящику; документувати знайдені в програмах дефекти; створювати супутні до тестування документи; розуміти доцільність проведення тестування, доцільність автоматизації тестування.
Як можна користуватися набутими знаннями і уміннями	Набуті знання й навички дозволять студентам бути більш кваліфікованими спеціалістами в області розробки й тестування програмного забезпечення, навчитися не тільки тому, як правильно написати ту чи іншу програму, але й переконатися, що вона дійсно працює коректно.
Інформаційне забезпечення дисципліни	Монографії та онлайн-підручники. Репозиторій лекцій-презентацій, матеріали для виконання лабораторних робіт, тестові завдання Силабус
Вид семестрового контролю	залік

8. Забезпечення якості програмного забезпечення	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	2, 4
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Базові знання з дисциплін «Програмування», «Дискретна математика», «Алгоритми і структури даних»
Що буде вивчатися	Моделі якості, життєвий цикл ПЗ, методи забезпечення якості програм на кожному етапі життєвого циклу, метрики якості, принципи і методи тестування ПЗ, супутня документація, основи автоматизації тестування
Чому це цікаво/треба вивчати	З кожним роком істотно зростає рівень складності і важливості програмного забезпечення, при цьому забезпечення якості є невід'ємною частиною процесу розробки на кожному етапі створення ПЗ
Чому можна навчитися	Методи забезпечення якості на ранніх етапах розробки: проводити аналіз вимог до ПЗ, знаходити помилки в специфікаціях, чітко формулювати вимоги до власних проектів; методи оцінки якості програмного коду на основі метрик та існуючих програмних засобів; пошук та документування дефектів в програмах; виконання тест-дизайну на основі стратегій «білого» і «чорного ящика»; документування тест-кейсів та тестових сценаріїв; тестування програм на мові Python за допомогою бібліотеки unittest; тестування веб-сторінок за допомогою Selenium IDE
Як можна користуватися набутими знаннями і уміннями	По закінченню курсу студенти будуть в змозі оцінити якість програмного забезпечення, провести аналіз вимог до програмного забезпечення, на основі різних стратегій провести його верифікацію та валідацію, правильно створити супутню документацію, забезпечити бездоганну роботу власних програм на мові Python
Інформаційне забезпечення дисципліни	Монографії та онлайн-підручники. Репозиторій лекцій-презентацій, посібник і матеріали для виконання лабораторних робіт, тестові завдання Силабус
Вид семестрового контролю	залік

9. Автоматизоване тестування програмного забезпечення	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	2, 4
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Базові знання з дисциплін «Програмування», «Дискретна математика», «Алгоритми і структури даних»
Що буде вивчатися	інструментальні засоби автоматизації тестування
Чому це цікаво/треба вивчати	Тестування програмного забезпечення є важливою і невід'ємною частиною сучасного процесу створення програмного продукту. Тестування вручну з багатьох причин повинно бути зведено до мінімуму, що призводить до необхідності організації автоматизованого тестування. Тому знання і вміння з автоматизації тестування є затребуваними на ринку праці
Чому можна навчитися	Робота з вимогами до програмного забезпечення і постановкою задач; різновиди автоматизованого тестування; стратегії «білого» і «чорного ящика» при автоматизації тестування; рівні тестування, модульне та приймальне тестування; використання бібліотеки unittest як демонстрація автоматизації модульного тестування; Selenium як інструмент приймального тестування; практичні роботи з повного тестування проекту
Як можна користуватися набутими знаннями і уміннями	По закінченню курсу студенти будуть в змозі безпосередньо приймати участь в розробці проектів як працівники з автоматизації тестування, зокрема, розробляти та реалізовувати програмно юніт-тести, а також проводити приймальне тестування з використанням Selenium
Інформаційне забезпечення дисципліни	Монографії та онлайн-підручники. Репозиторій лекцій-презентацій, матеріали для виконання лабораторних робіт, тестові завдання Силабус
Вид семестрового контролю	залік

Анотації вибіркових дисциплін для 3 курсу

10. Дослідження операцій	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	3, 5
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Базові знання з дисциплін «Алгебра та геометрія», «Математичний аналіз», «Чисельні методи», «Програмування», «Алгоритми і структури даних»
Що буде вивчатися	Основні поняття та методи дослідження операцій. Теоретичні основи алгоритмів методів дослідження операцій, їх застосування та реалізація для вирішення широкого спектру практичних задач.
Чому це цікаво/треба вивчати	Важливим завданням сучасності є керування складними системами, оптимізація їх структури, розвитку й функціонування з метою досягнення максимальної ефективності. Дослідження операцій є основним інструментом при прийнятті рішень в найрізноманітніших областях людської діяльності, який передбачає застосування математичних методів для знаходження оптимальних рішень з використанням сучасної обчислювальної техніки.
Чому можна навчитися	Студенти будуть вміти самостійно ставити та розв'язувати оптимізаційні задачі за допомогою аналітичних та чисельних методів. Володіти вміннями розв'язування задач лінійного, дискретного, опуклого та нелінійного програмування.
Як можна користуватися набутими знаннями і уміннями	Використовувати набуті теоретичні знання та практичні навички при побудові та дослідженні математичних моделей обчислювальних та інформаційних процесів, пов'язаних з функціонуванням об'єктів професійної діяльності з використанням комп'ютерної техніки.
Інформаційне забезпечення дисципліни	Слайди презентацій, класичні підручники, методичні посібники, онлайн підручник Силабус
Вид семестрового контролю	Залік

11. Задачі дискретної оптимізації	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	3, 5
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Базові знання з дисциплін «Алгебра та геометрія», «Математичний аналіз», «Чисельні методи», «Програмування», «Алгоритми і структури даних»
Що буде вивчатися	Основні поняття та теоретичні відомості лінійного програмування (ЛП), методи розв'язування задач ЛП, теорія двоїстості в ЛП. Основні поняття та теоретичні відомості дискретного програмування, методи розв'язування задач цілочисельного лінійного програмування.
Чому це цікаво/треба вивчати	Методи і моделі лінійного програмування та лінійного цілочисельного програмування широко застосовуються при оптимізації процесів у всіх галузях народного господарства: при розробці виробничої програми підприємства, розподілі її по виконавцях, при розміщенні замовлень між виконавцями і по тимчасових інтервалах, при визначенні найкращого асортименту продукції, що випускається, в задачах перспективного, поточного та оперативного планування і управління; при плануванні вантажопотоків, визначенні плану товарообігу і його розподіл; в задачах розвитку і розміщення продуктивних сил, баз і складів матеріальних ресурсів і т.д.
Чому можна навчитися	Студенти будуть вміти самостійно ставити та розв’язувати оптимізаційні задачі за допомогою аналітичних та чисельних методів. Володіти вміннями розв’язування транспортних задач (збалансованих, незбалансованих, з обмеженнями пропускної здатності), задач про призначення та задач комівояжера
Як можна користуватися набутими знаннями і уміннями	Використовувати набуті теоретичні знання та практичні навички при вирішенні багаточисельних проблем управління і планування виробничих процесів, в задачах проектування та перспективного планування, при організації функціонування та розвитку соціальних процесів, їх координації з господарськими та економічними процесами.
Інформаційне забезпечення дисципліни	Слайди презентацій, класичні підручники , методичні посібники, онлайн підручник Силабус
Вид семестрового контролю	Залік

12. Лінійне програмування та теорія ігор	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	3, 5
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Базові знання з дисциплін «Алгебра та геометрія», «Математичний аналіз», «Чисельні методи», «Програмування», «Алгоритми і структури даних»
Що буде вивчатися	Основні поняття та теоретичні відомості лінійного програмування (ЛП), методи розв'язування задач ЛП, теорія двоїстості в ЛП та аналіз моделі ЛП на чутливість. Основні поняття та теоретичні відомості теорії ігор. Розгляд типових ігрових ситуацій та їх математичний опис, методи розв’язання ігрових задач.
Чому це цікаво/треба вивчати	Методи і моделі лінійного програмування широко застосовуються при оптимізації процесів у всіх галузях народного господарства: при розробці виробничої програми підприємства, розподілі її по виконавцях, при розміщенні замовлень між виконавцями і по тимчасових інтервалах, при визначенні найкращого асортименту продукції, що випускається, в задачах перспективного, поточного та оперативного планування і управління; при плануванні вантажопотоків, визначенні плану товарообігу і його розподіл; в задачах розвитку і розміщення продуктивних сил, баз і складів матеріальних ресурсів і т.д. Теорія ігор – це розділ математики, в якому досліджуються математичні моделі прийняття рішень в умовах конфліктів, тобто в умовах зіткнення сторін.
Чому можна навчитися	Студенти будуть мати знання теорії лінійного програмування, методів розв’язування задач лінійного програмування, основних особливостей та умов використання методів, проведення аналізу отриманого оптимального рішення. Застосування методів і моделей теорії ігор в процесі розв’язування теоретичних і практичних задач.
Як можна користуватися набутими знаннями і уміннями	Використовувати набуті теоретичні знання та практичні навички при побудові та дослідженні математичних моделей обчислювальних та інформаційних процесів методами лінійного програмування та теорії ігор.
Інформаційне забезпечення дисципліни	Слайди презентацій, класичні підручники, методичні посібники, онлайн підручник, силабус
Вид семестрового контролю	Залік

13. Математичне програмування	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	3, 5
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 54 аудиторних годин, 66 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Базові знання з дисциплін «Алгебра та геометрія», «Математичний аналіз», «Чисельні методи», «Програмування», «Алгоритми і структури даних»
Що буде вивчатися	Класи задач, які розглядає математичне програмування. Основні поняття математичного програмування. Методи розв'язання задач математичного програмування. Теоретичні основи алгоритмів методів дослідження операцій.
Чому це цікаво/треба вивчати	Ефективність роботи сучасних підприємств, які є складними системами, залежить від якості організаційного управління. Процеси прийняття рішень лежать в основі будь-якої цілеспрямованої діяльності. При формуванні стратегічних і тактичних рішень керівник повинен враховувати безліч часом суперечливих міркувань, спиратися на складні критерії ефективності шляхів досягнення кінцевих цілей. У зв'язку з цим виникла необхідність застосовувати для аналізу і синтезу економічних ситуацій і систем математичні методи і сучасну обчислювальну техніку. Такі методи об'єднуються під загальною назвою – математичне програмування. Завдання математичного програмування знаходять застосування в різних областях людської діяльності, де необхідний вибір одного з можливих варіантів дій. Оптимальні (ефективні) рішення дозволяють досягати мети при мінімальних витратах трудових, матеріальних і сировинних ресурсів.
Чому можна навчитися	Студенти будуть вміти самостійно ставити та розв'язувати оптимізаційні задачі за допомогою аналітичних та чисельних методів. Володіти вміннями розв'язування задач лінійного, дискретного, опуклого та нелінійного програмування, транспортних задач (збалансованих, незбалансованих, з обмеженнями пропускних спроможностей комунікацій), оптимізаційних моделей на мережах (задача про найкоротший шлях та максимальний потік)
Як можна користуватися набутими знаннями і уміннями	Використовувати набуті теоретичні знання та практичні навички при побудові та дослідженні математичних моделей обчислювальних та інформаційних процесів, пов'язаних з функціонуванням об'єктів професійної діяльності з використанням комп'ютерної техніки. Наприклад при вирішенні багаточисельних проблем управління і планування виробничих процесів, в задачах проектування та перспективного планування, при організації функціонування та розвитку соціальних процесів, їх координації з господарськими та економічними процесами.
Інформаційне забезпечення дисципліни	Слайди презентацій, класичні підручники, методичні посібники, онлайн підручник, силабус
Вид семестрового контролю	Залік

14. Практичні аспекти чисельної оптимізації	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	3, 5
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 54 аудиторних годин, 66 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Базові знання з дисциплін «Алгебра та геометрія», «Математичний аналіз», «Чисельні методи», «Програмування», «Алгоритми і структури даних»
Що буде вивчатися	Проведення наукового дослідження, планування експерименту, накопичення і обробки емпіричного матеріалу та його аналізу, формулювання відповідних висновків та рекомендацій. Формалізування задачі багатопараметричної оптимізації. Методи розв'язання задач багатопараметричної оптимізації
Чому це цікаво/треба вивчати	Налаштування моделі алгоритмів за даними - це завдання оптимізації, від ефективності вирішення якої залежить практична застосовність методу машинного навчання. В епоху великих даних багато класичних алгоритмів оптимізації стають непридатними, тому що тут потрібно вирішувати завдання оптимізації функцій за час менше, ніж необхідно для обчислення значення функції в одній точці. Таким вимогам можна задовольнити в разі грамотного комбінування відомих підходів в оптимізації з урахуванням конкретної специфіки задачі, яка розв'язується.
Чому можна навчитися	Студенти будуть мати знання о ролі методів оптимізації в прикладних науках і розв'язанні практичних задач, основних особливостей методів оптимізації для задач нелінійного програмування, умов використання методів залежно від особливостей задачі, можливостей адаптації методів при вирішенні конкретних практичних задач.
Як можна користуватися набутими знаннями і уміннями	При проведенні самостійного наукового практичного дослідження студенти будуть вміти аналізувати поставлену задачу оптимізації, обрати найбільш ефективний для її розв'язання метод, реалізовувати обраний метод та одержати практичні результати.
Інформаційне забезпечення дисципліни	Слайди презентацій, класичні підручники , методичні посібники, онлайн підручник, силабус
Вид семестрового контролю	Залік

15. Нелінійне програмування	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	3, 5
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 54 аудиторних годин, 66 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Базові знання з дисциплін «Алгебра та геометрія», «Математичний аналіз», «Чисельні методи», «Програмування», «Алгоритми і структури даних»
Що буде вивчатися	Вибір оптимального варіанту рішення технічних задач є однією з актуальних проблем в інженерній діяльності. За змістом задачі оптимізації можуть бути дуже різними. Але математично вони являють собою задачі мінімізації (максимізації) деякого функціоналу, який залежить від характеристик технічної системи. Чисельне рішення оптимізаційних задач на комп'ютері зводиться до пошуку екстремуму функції багатьох змінних за допомогою детермінованих методів пошуку та методів випадкового пошуку. Це широкий клас методів, який найбільш повно використовує можливості сучасної обчислювальної техніки.
Чому це цікаво/треба вивчати	У сучасному виробництві часто виникають задачі синтезу систем автоматичного управління складними об'єктами. Одним з етапів моделювання таких систем є оптимізація нелінійної цільової функції. В основі рішення багатьох задач комп'ютерних наук лежать методи нелінійного програмування .Наприклад, у машинному навчанні задачу оптимізації необхідно вирішувати кожен раз при налаштуванні якоїсь моделі алгоритмів за даними, причому від ефективності вирішення відповідної задачі оптимізації залежить практична застосовність самого методу машинного навчання.
Чому можна навчитися	Студенти будуть мати знання основних понять теорії оптимізації, різних класів методів розв'язування задач нелінійного програмування, ролі методів оптимізації в прикладних науках і розв'язанні практичних задач, основних особливостей методів оптимізації для задач нелінійного програмування, умов використання методів залежно від особливостей задачі, можливостей адаптації методів при розв'язанні конкретних практичних задач.
Як можна користуватися набутими знаннями і уміннями	Вибір того чи іншого методу обумовлений складністю обчислення критерію оптимальності і складністю обмежуючих умов, необхідної точністю рішення, потужністю наявної обчислювальної машини, тому студенти будуть вміти аналізувати поставлену задачу оптимізації, щоб обрати найбільш ефективний для її розв'язання метод, реалізувати обраний метод та одержати практичні результати.
Інформаційне забезпечення дисципліни	Слайди презентацій, класичні підручники , методичні посібники, онлайн підручник, силабус
Вид семестрового контролю	Залік

16. Випадкові процеси	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	3, 5
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 54 аудиторних годин, 66 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Базові знання з дисциплін «Математичний аналіз», «Теорія ймовірностей», «Математична статистика»
Що буде вивчатися	Основи теорії випадкових процесів, Марківські випадкові процеси, ланцюги Маркова, аналітичні властивості випадкових процесів, перетворення випадкових процесів, стаціонарні випадкові процеси
Чому це цікаво/треба вивчати	Теорія випадкових процесів лежить в основі стохастичних моделей та алгоритмів, що використовуються в аналізі даних, зокрема й інтелектуальному аналізі даних
Чому можна навчитися	Використовувати випадкові процеси у практичних застосуваннях, зокрема, інженерній практиці, аналізі даних та інтелектуальному аналізі даних. Розуміти принципи роботи з стохастичними моделями даних. Реалізовувати аналіз даних з використанням бібліотек сучасних програмних засобів.
Як можна користуватися набутими знаннями і уміннями	По закінченні курсу студенти будуть у змозі використовувати основи випадкових процесів при розв'язанні прикладних задач (в т. ч. для аналізу даних) та застосовувати програмні засоби для їх вирішення.
Інформаційне забезпечення дисципліни	Монографії та онлайн-підручники. Відеолекції. Онлайн-курси за методикою змішаного навчання. Силабус.
Вид семестрового контролю	залік

17. Статистичне моделювання	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	3, 5
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 54 аудиторних годин, 66 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Базові знання з дисциплін «Математичний аналіз», «Теорія ймовірностей», «Математична статистика»
Що буде вивчатися	Основи випадкових процесів; методи та алгоритми статистичного моделювання випадкових величин, векторів, процесів; використання методів статистичного моделювання при розв'язанні нестандартних прикладних задач
Чому це цікаво/треба вивчати	Методи статистичного моделювання лежать в основі організації та проведення обчислювальних експериментів, імітаційного моделювання, а також при розв'язанні різноманітних прикладних задач, зокрема, слабо формалізованих та структурованих
Чому можна навчитися	Моделювати реалізації випадкових величин, векторів та процесів і ефективно їх використовувати в сучасних інформаційних технологіях
Як можна користуватися набутими знаннями і уміннями	По закінченні курсу студенти будуть у змозі використовувати методи статистичного моделювання для розв'язання прикладних задач (в т.ч. для аналізу даних), застосовувати програмні засоби для їх вирішення. Будуть вміти організовувати та проводити обчислювальний експеримент
Інформаційне забезпечення дисципліни	Монографії та онлайн-підручники. Відеолекції. Онлайн-курси за методикою змішаного навчання. Силабус.
Вид семестрового контролю	Залік

18. Марківські моделі	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	3, 5
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 54 аудиторних годин, 66 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Базові знання з дисциплін «Математичний аналіз», «Теорія ймовірностей», «Математична статистика»
Що буде вивчатися	Основи теорії випадкових процесів, марківські випадкові процеси, марківські ланцюги, приховані марківські моделі
Чому це цікаво/треба вивчати	Стохастичні марківські моделі - одні з основних в аналізі даних та інтелектуальному аналізі даних.
Чому можна навчитися	Розуміти принципи роботи з стохастичними моделями даних марківського типу. Проводити аналіз даних та вибирати програмні засоби для його реалізації. Реалізовувати аналіз даних з використанням бібліотек сучасних програмних засобів.
Як можна користуватися набутими знаннями і уміннями	Будувати стохастичні моделі, що виникають при аналізі даних. Проводити аналіз даних. Вміти використовувати бібліотеки сучасних програмних систем (Python, R Studio) для аналізу даних.
Інформаційне забезпечення дисципліни	Монографії та онлайн-підручники. Відеолекції. Онлайн-курси за методикою змішаного навчання. Силабус.
Вид семестрового контролю	Залік

19. Рівняння математичної фізики	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	3, 6
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 54 аудиторних годин, 66 годин самостійної роботи
Мова викладання	українська
Вимоги до початку вивчення дисципліни	Базові знання з дисциплін «Математичний аналіз», «Диференціальні рівняння», «Алгебра та геометрія», «Функціональний аналіз»
Що буде вивчатися	Диференціальні рівняння в частинних похідних. Задачі Коші та початково-крайові задачі для рівнянь в частинних похідних. Методи розв’язування задач для рівнянь в частинних похідних.
Чому це цікаво/треба вивчати	Математична фізика розвивалася паралельно розвитку теоретичної математики і фізики, її предметна область — це вивчення фізичних процесів таких як коливання, дифузія у неперервних середовищах, явища теплопровідності методами теорії диференціальних операторів в частинних похідних. Апарат математичної фізики є основним способом дослідження фізичних полів та процесів в неперервних середовищах і широко застосовується в обчислювальному експерименті та математичному моделюванні.
Чому можна навчитися	Будувати математичні моделі фізичних явищ, визначати тип математичної моделі (класифікувати рівняння) та обирати метод аналітичного розв'язання, розв’язувати задачі Коші для хвильових рівнянь та рівнянь теплопровідності, розв’язувати крайові задачі для еліптичних, гіперболічних та параболічних рівнянь
Як можна користуватися набутими знаннями і уміннями	По закінченню курсу студенти зможуть проводити аналіз математичних моделей різноманітних процесів, що описуються лінійними рівняннями в частинних похідних, будувати розв’язки відповідних крайових задач, проводити аналіз отриманих результатів.
Інформаційне забезпечення дисципліни	Онлайн-підручники, тестові завдання. Силабус
Вид семестрового контролю	Залік

20. Інтегральні перетворення і диференціальні рівняння	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	3, 6
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 54 аудиторних годин, 66 годин самостійної роботи
Мова викладання	українська
Вимоги до початку вивчення дисципліни	Базові знання з дисциплін «Математичний аналіз», «Диференціальні рівняння», «Алгебра та геометрія», «Функціональний аналіз»
Що буде вивчатися	Інтегральні перетворення Фур'є, Лапласа, Ханкеля. Лінійні диференціальні рівняння в частинних похідних (ДРЧП). Застосування інтегральних перетворень до розв'язування граничних задач для ДРЧП.
Чому це цікаво/треба вивчати	ДРЧП описують велике коло фізичних задач (коливання, теплообмін, стаціонарні поля) і мають широке застосування в математичному моделюванні. Інтегральні перетворення використовуються як в теоретичних дослідженнях граничних задач для рівнянь в частинних похідних, так і для побудови їх розв'язків.
Чому можна навчитися	Класифікувати рівняння в частинних похідних, ставити граничні та задачі Коші для ДРЧП, застосовувати інтегральні перетворення до диференціальних операторів, розв'язувати граничні задачі використовуючи інтегральні перетворення.
Як можна користуватися набутими знаннями і уміннями	По закінченню курсу студенти зможуть використовувати апарат інтегральних перетворень для дослідження та розв'язування граничних задач для диференціальних рівнянь в частинних похідних.
Інформаційне забезпечення дисципліни	Навчальні посібники, підручники, збірники задач, тестові завдання. Силабус
Вид семестрового контролю	Залік

21. Класична та сучасна теорії параболічних задач	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	3, 6
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 54 аудиторних годин, 66 годин самостійної роботи
Мова викладання	українська
Вимоги до початку вивчення дисципліни	Базові знання з дисциплін «Математичний аналіз», «Диференціальні рівняння», «Алгебра та геометрія», «Функціональний аналіз»
Що буде вивчатися	Диференціальні рівняння в частинних похідних параболічного типу. Задача Коші та крайова задача. Простори Соболєва. Класичні та узагальнені розв'язки параболічних задач. Методи розв'язування параболічних крайових задач.
Чому це цікаво/треба вивчати	Диференціальні рівняння в частинних похідних параболічного типу описують широке коло фізичних процесів, зокрема, тепломасообмін, дифузія у неперервних середовищах тощо. Перехід від класичних до узагальнених розв'язків дозволяє істотно послабити умови на коефіцієнти і праві частини крайових задач, що є важливим з практичної точки зору.
Чому можна навчитися	Ставити початково-крайову та задачу Коші для параболічних рівнянь. Досліджувати розв'язність таких задач в класах неперервно-диференційовних функцій та в класах функціональних просторів Соболєва. Будувати розв'язки параболічних задач.
Як можна користуватися набутими знаннями і уміннями	По закінченню курсу студенти зможуть використовувати елементи теорії узагальнених функції для дослідження крайових задач для рівнянь в частинних похідних; досліджувати існування і єдиність класичних та узагальнених розв'язків параболічних крайових задач та будувати їх розв'язки.
Інформаційне забезпечення дисципліни	Навчальні посібники, підручники, збірники задач, тестові завдання. Силабус
Вид семестрового контролю	Залік

22. Front-end розробка	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	3, 6
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Розуміння принципів об'єктно-орієнтованої розробки. Розуміння однієї з С-подібних об'єктно-орієнтованих мов (C\C++, Java, C#). Розуміння принципів роботи комп'ютерних мереж.
Що буде вивчатися	Протокол HTTP. Розробка клієнт-орієнтованого коду мовою JavaScript для виконання у браузері. Технологія AJAX. Мови розмітки HTML та CSS. Платформа Node.js. Фреймворк React.
Чому це цікаво/треба вивчати	Веброзробка — одна з найбільших частин нинішнього ринку застосунків для користувачів. Кожен має браузер і доступ в інтернет майже кожної хвилини, відтак може запускати веб-застосунки та сайти, тому вони стають все більш функціональними. В цьому курсі ми розглянемо основи роботи з клієнтською частиною вебзастосунків — від верстки вебсторінок до програмування реактивної поведінки на дії користувача, користуючись популярним фреймворком React від Facebook.
Чому можна навчитися	Розуміти принципи роботи протоколу HTTP. Розробляти дизайн вебсторінок за допомогою мов HTML та CSS. Розробляти динамічну поведінку вебсторінок за допомогою мови JavaScript. Користуватися платформою Node.js для запуску JavaScript коду локально. Користуватися фреймворком React та спеціальним діалектом JSX.
Як можна користуватися набутими знаннями і уміннями	По завершенню курсу, студенти будуть у змозі реалізувати клієнтську частину вебзастосунку: від верстки до динамічної взаємодії із сервером, за допомогою традиційних засобів HTML/CSS/JavaScript та фреймворк React.
Інформаційне забезпечення дисципліни	Слайди презентацій та онлайн-репозиторій лекційних прикладів. Сайти документації розглянутого програмного забезпечення. Онлайн-курси за методикою змішаного навчання. Силабус.
Вид семестрового контролю	Залік

23. Побудова REST-сервісів	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	3, 6
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Розуміння принципів роботи комп'ютерних мереж. Практичні навички програмування на Python чи іншій об'єктно-орієнтованій мові. Практичні навички роботи з базами даних.
Що буде вивчатися	Протокол HTTP. Принципи реалізації серверних застосунків. Мова програмування PHP, фреймворки Django та Flask. Реалізація взаємодії із клієнтським застосунком засобами протоколу HTTP (REST).
Чому це цікаво/треба вивчати	Більшість сучасних сайтів та вебзастосунків підтримують постійний контакт із відповідними серверними застосунками для підвантаження даних та оновлення доступного користувачеві контенту. Чим більша частина програмних рішень переходить на використання веб-інтерфейсів, тим більшим попитом будуть користуватися спеціалісти, що можуть правильно організувати клієнт-серверну взаємодію.
Чому можна навчитися	Розуміти принципи роботи протоколу HTTP. Розробляти серверне програмне забезпечення комунікації з клієнтським застосунком. Розуміти принципи роботи сервера, основи розміщення застосунків у хмарі. Розуміти синтаксис та принципи роботи мови програмування PHP. Користуватися фреймворками Django та Flask.
Як можна користуватися набутими знаннями і уміннями	По завершенню курсу, студенти будуть у змозі реалізувати серверну частину вебзастосунку: від програмування до розміщення у хмарі, за допомогою мови PHP чи фреймворків Django чи Flask.
Інформаційне забезпечення дисципліни	Слайди презентацій та онлайн-репозиторій лекційних прикладів. Сайти документації розглянутого програмного забезпечення. Онлайн-курси за методикою змішаного навчання. Силабус
Вид семестрового контролю	Залік

24. Веб-технології візуалізації даних	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	3, 6
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Розуміння принципів роботи комп'ютерних мереж. Навички програмування мовою Python чи іншою об'єктно-орієнтованою мовою. Навички роботи з базами даних.
Що буде вивчатися	Вебсистеми збору, обробки і візуалізації даних з використанням JavaScript: Plotly, Seaborn, d3.js. Технологія AJAX. Фреймворк Flask.
Чому це цікаво/треба вивчати	Збір та візуалізація даних — важливі інструменти в руках будь-якого data science спеціаліста: уміння швидко опитувати віддалені джерела даних, аналізувати інформацію та представляти її в якомога наочнішому вигляді стають все більш широко затребуваним умінням в часи, коли все більше інформації та бізнес-рішень переходить в онлайн.
Чому можна навчитися	Аналізувати структуру даних та обирати найбільш вдалий спосіб їх репрезентації. Використовувати хмарні інструменти аналізу даних. Створювати комплексні системи збору даних з різних джерел. Проводити перетворення даних на клієнтському застосунку та на сервері. Візуалізовувати дані за допомогою одного із вибраних фреймворків.
Як можна користуватися набутими знаннями і уміннями	По закінченню курсу, студенти будуть у змозі створювати онлайн-застосунки для збору інформації за допомогою технології AJAX, обробляти її та відображати візуальні дашборди з використанням мови програмування JavaScript та фреймворку Flask.
Інформаційне забезпечення дисципліни	Слайди презентацій та онлайн-репозиторій лекційних прикладів. Сайти документації розглянутого програмного забезпечення. Онлайн-курси за методикою змішаного навчання. Силабус
Вид семестрового контролю	Залік

Анотації вибіркових дисциплін для 4 курсу

25. Системи глибинного навчання	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	4, 7
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 54 аудиторних годин, 66 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Базові знання з дисциплін: «Математична логіка та теорія алгоритмів», «Алгоритми і структури даних», «Математична статистика», «Аналіз даних», «Архітектура обчислювальних систем», «Програмування»
Що буде вивчатися	Системи глибинного навчання (Deep Learning) - це міждисциплінарна прикладна галузь, що вивчає системи і процеси, які призначені для аналізу і обробки даних за допомогою найбільш сучасних типів глибоких нейронних мереж
Чому це цікаво/треба вивчати	Deep Learning, це сучасна тенденція розвитку методів і засобів обробки і аналізу даних із застосуванням глибоких нейронних мереж. Оволодіння інструментами створення і використання глибоких нейронних мереж є невід'ємним елементом підготовки висококваліфікованих спеціалістів у різних галузях національного господарства України та світу
Чому можна навчитися	Знати основні положення теорії штучних нейронних мереж, методологію побудови та використання глибоких нейронних мереж, методи застосування глибоких нейронних мереж для вирішення практичних задач. Уміння розробляти, тестувати та реалізовувати нейромережеві засоби з використанням мови програмування Python та бібліотек TensorFlow і Keras. Навички та досвід розв'язання типових задач розпізнавання образів
Як можна користуватися набутими знаннями і уміннями	Здатність розв'язувати професійні задачі аналізу та обробки даних за допомогою систем Deep Learning. Здатність розроблювати, експлуатувати та обслуговувати програмне забезпечення для інформаційних та інформаційно-управляючих систем Deep Learning різного призначення
Інформаційне забезпечення дисципліни	Монографії та онлайн-підручники, матеріали для виконання лабораторних робіт, тестові завдання
Вид семестрового контролю	Залік

26. Обчислювальний інтелект	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	4, 7
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 54 аудиторних годин, 66 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Базові знання з дисциплін: «Математична логіка та теорія алгоритмів», «Алгоритми і структури даних», «Математична статистика», «Аналіз даних», «Архітектура обчислювальних систем», «Програмування»
Що буде вивчатися	Дисципліна обіймає проблематику вивчення сучасного стану технологій обчислювального інтелекту, що використовуються для формалізації та обробки знань в технологіях функціонування систем, вивчення сучасних програмних засобів і технологій проектування і реалізації інтелектуальних систем та їх налагодження і дослідження
Чому це цікаво/треба вивчати	Проблема аналізу інформації з використанням технології обчислювального інтелекту є дуже актуальною в сьогочасних умовах широкого використання інформаційних технологій
Чому можна навчитися	Після вивчення дисципліни студенти зможуть використати набуті знання та вміння в галузі обчислювального інтелекту при проектуванні, моделюванні та налагодженні інформаційних систем, комп'ютеризованих систем управління, систем і технологій розпізнавання образів та прийняття рішень, використовуючи для цього сучасні програмні середовища
Як можна користуватися набутими знаннями і уміннями	Метою вивчення дисципліни є формування у майбутніх фахівців знань та вмінь застосування сучасних методів та засобів проектування, розробки та дослідження інтелектуальних систем, що базуються на технологіях обчислювального інтелекту, і одержання навиків використання таких систем та технологій у своїй практичній роботі.
Інформаційне забезпечення дисципліни	Монографії та онлайн-підручники, матеріали для виконання лабораторних робіт, тестові завдання
Вид семестрового контролю	Залік

27. Програмування систем штучного інтелекту	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	4, 7
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 54 аудиторних годин, 66 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Базові знання з дисциплін: «Математична логіка та теорія алгоритмів», «Алгоритми і структури даних», «Математична статистика», «Аналіз даних», «Архітектура обчислювальних систем», «Програмування»
Що буде вивчатися	Дисципліна обіймає проблематику вивчення засобів розробки програмного забезпечення систем штучного інтелекту, що використовуються для формалізації та обробки інформації в інформаційних системах різноманітного призначення
Чому це цікаво/треба вивчати	Створення систем штучного інтелекту, що призначені для застосування в універсальних інформаційних системах, є однією з найбільш актуальних проблем в сьогочасних умовах широкого використання інформаційних технологій
Чому можна навчитися	Після вивчення дисципліни студенти зможуть використати набуті знання та вміння в галузі розробки програмного забезпечення систем штучного інтелекту, що застосовуються в різних галузях народного господарства
Як можна користуватися набутими знаннями і уміннями	Метою вивчення дисципліни є формування у майбутніх фахівців знань та вмінь застосування сучасних методів та засобів проектування та розробки програмного забезпечення для реалізації систем штучного інтелекту і одержання навиків використання такого програмного забезпечення в своїй практичній роботі
Інформаційне забезпечення дисципліни	Монографії та онлайн-підручники, матеріали для виконання лабораторних робіт, тестові завдання
Вид семестрового контролю	Залік

28. Інформаційні системи	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	4, 7
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Базові знання об'єктно-орієнтованого програмування, основи баз даних, основи веб-розробки та тестування, git
Що буде вивчатися	Підходи до побудови інформаційних систем та програмних продуктів від бізнес проблеми до готової системи. Методології роботи з функціональними та нефункціональними вимогами. Побудова архітектури системи на основі актуальних архітектурних стилів (наприклад, мікросервісів) та шаблонів. Підходи для забезпечення стійкості та надійності систем. Способи розгортання інформаційних систем на хмарних хостингах (cloud) та on-prem.
Чому це цікаво/треба вивчати	Правильно побудована архітектура системи та процесів розробки дозволяє суттєво зменшити витрати на створення, розвиток та експлуатацію програмного продукту, а іноді відрізняє вдалий проект чи стартап від того який вже не існує.
Чому можна навчитися	Студенти можуть опанувати основні найбільш популярні підходи до побудови систем, навчитись обирати найбільш ефективні підходи для різних типів проектів та обґрунтовувати свої рішення. Отримають досвід командної роботи та навчаться застосовувати гнучкі методології розробки. Матимуть змогу опанувати актуальні практики побудови мікросервісних систем та сучасні підходи до побудови стійких та надійних систем з високою доступністю. Отримають досвід побудови CI/CD процесів.
Як можна користуватися набутими знаннями і уміннями	Набуті знання можуть бути застосовані як при побудові стартапу, так і при розробці корпоративних систем (для невеликих локальних бізнесів та міжнародних корпорацій)
Інформаційне забезпечення дисципліни	Монографії та онлайн-підручники. Репозиторій лекцій-презентацій, посібник і матеріали для виконання лабораторних робіт, тестові завдання
Вид семестрового контролю	Залік

29. Забезпечення стійкості інформаційних систем	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	4, 7
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Знання об'єктно-орієнтованого програмування (бажано на основі мови Java), основи баз даних, основи веб-розробки та тестування, git
Що буде вивчатися	Загальні принципи побудови клієнт-серверних систем. Підходи для забезпечення стійкості та надійності систем. Способи розгортання інформаційних систем на хмарних хостингах (cloud) та on-prem.
Чому це цікаво/треба вивчати	В сучасному світі від корпоративних інформаційних систем очікують вже не здатність працювати протягом робочого дня робочого тижня (8/5), а цілодобово протягом всього тижня 24/7
Чому можна навчитися	Імплементувати та застосовувати при побудові клієнт-серверних систем основі шаблони забезпечення стійкості. Загальні підходи до підтримки систем та аналізу інцидентів.
Як можна користуватися набутими знаннями і уміннями	Набуті знання можуть бути застосовані при розробці інформаційних систем різної складності, а також при підтримці legacy-систем.
Інформаційне забезпечення дисципліни	Монографії та онлайн-підручники. Репозиторій лекцій-презентацій, посібник і матеріали для виконання лабораторних робіт, тестові завдання
Вид семестрового контролю	Залік

30. Інтеграційні шаблони в інформаційних системах	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	4, 7
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Знання об'єктно-орієнтованого програмування (бажано на основі мови Java), основи баз даних, основи веб-розробки та тестування, розуміння принципів роботи розподілених та корпоративних систем, git
Що буде вивчатися	Загальні принципи побудови клієнт-серверних корпоративних систем. Способи розгортання інформаційних систем на хмарних хостингах (cloud) та on-prem. Шаблони та підходи інтеграції різних компонентів системи або різних систем.
Чому це цікаво/треба вивчати	В сучасному світі від корпоративних інформаційних систем очікують можливість надійної інтеграції як з іншими системами в рамках підприємства так і з системами постачальників та клієнтів.
Чому можна навчитися	Імплементувати та застосовувати при побудові клієнт-серверних систем основі інтеграційні шаблони.
Як можна користуватися набутими знаннями і уміннями	Набуті знання можуть бути застосовані при розробці інформаційних систем різної складності, а також при підтримці legacy-систем.
Інформаційне забезпечення дисципліни	Монографії та онлайн-підручники. Репозиторій лекцій-презентацій, посібник і матеріали для виконання лабораторних робіт, тестові завдання
Вид семестрового контролю	Залік

31. Розподілені і хмарні обчислення	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	4, 7
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 54 аудиторних годин, 66 годин самостійної роботи
Мова викладання	українська
Вимоги до початку вивчення дисципліни	Знання з дисциплін «Архітектура обчислювальних систем», «Програмування», «Алгоритми і структури даних»
Що буде вивчатися	Основні поняття та концепції розподілених і хмарних обчислень, методів та підходів до розв'язання прикладних задач за допомогою розподілених і хмарних обчислень.
Чому це цікаво/треба вивчати	Завдяки об'єднанню ресурсів і непостійного характеру споживання з боку споживачів, розподілені і хмарні обчислення дозволяють економити на масштабах, використовуючи менші апаратні ресурси, ніж були потрібні б при виділених апаратних потужностях для кожного користувача.
Чому можна навчитися	- мета побудови розподілених і хмарних систем; - поняття розподілених обчислень та розподіленої системи, хмарних обчислень та хмарних систем; - вимоги до розподілених і хмарних систем; - класифікація розподілених і хмарних систем та їх призначення; - переваги та недоліки розподілених і хмарних обчислень; - основні парадигми застосування розподілених і хмарних обчислень; - моделі розподілених і хмарних архітектур. - паралельні чисельні алгоритми для вирішення типових задач обчислювальної математики.
Як можна користуватися набутими знаннями і уміннями	- розробляти моделі розподілених обчислень; - проводити аналіз комунікаційної трудомісткості розподілених алгоритмів; - аналізувати та обирати оптимальні рішення щодо залучення засобів розподілених і хмарних обчислень у напрямку їх застосування для проведення наукових досліджень, а також для створення єдиного обчислювального середовища організації, підприємства; - застосовувати базові знання стандартів в області інформаційних технологій під час розробки та впровадження розподілених обчислювальних систем на базі хмарних технологій та сервісів; - проектувати компоненти програмного забезпечення для роботи в якості сервісів у складі розподілених обчислювальних систем та комплексів й хмарних обчислень.
Інформаційне забезпечення дисципліни	Слайди презентацій, класичні підручники, методичні посібники, онлайн підручник. Силабус
Вид семестрового контролю	Залік

32. Програмне забезпечення розподілених систем	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	4, 7
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 54 аудиторних годин, 66 годин самостійної роботи
Мова викладання	українська
Вимоги до початку вивчення дисципліни	Знання з дисциплін «Архітектура обчислювальних систем», «Програмування», «Алгоритми і структури даних»
Що буде вивчатися	Основні поняття та концепцій побудови програмного забезпечення для розподілених систем та розподілених обчислень, методи та алгоритми рішення математичних задач за допомогою розподілених алгоритмів.
Чому це цікаво/треба вивчати	Вивчення теоретичних основ та практичних аспектів використання розподілених систем для вирішення складних прикладних задач з великим об’ємом обчислень.
Чому можна навчитися	- мета побудови розподілених систем; - поняття розподілених обчислень та розподіленої системи, математичний апарат і моделі паралельних і розподілених обчислень; - вимоги до розподілених систем; - системи розробки паралельних програм; - теоретичні основи організації паралельних і розподілених обчислювальних процесів, розпаралелювання алгоритмів, перетворення послідовних програм в паралельні; - проблематика організації паралельних та розподілених обчислень; - особливості розробки програмного забезпечення для роботи у складі розподілених обчислювальних систем.
Як можна користуватися набутими знаннями і уміннями	- розробляти моделі розподілених обчислень; - проводити аналіз комунікаційної трудомісткості розподілених алгоритмів; - застосовувати технологію розробки розподілених програм для багатопроцесорних систем з розподіленою пам'яттю (стандарт передачі повідомлень MPI); - оцінювати ефективність розроблених паралельних програм, враховуючи основні показники ефективності: час виконання програми, прискорення, ефективність та вартість; - використовувати розподілену парадигму проектування програмного забезпечення; - вміти адаптувати паралельні програми для виконання на кластерних системах; - встановлювати порядок виконання операцій та обміну даними; - використовувати інструментальні засоби для організації розподілених обчислювальних процесів.
Інформаційне забезпечення дисципліни	Слайди презентацій, класичні підручники, методичні посібники, онлайн підручник, силабус.
Вид семестрового контролю	Залік

33. Операційні системи	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	4, 7
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 54 аудиторних годин, 66 годин самостійної роботи
Мова викладання	українська
Вимоги до початку вивчення дисципліни	Знання з дисциплін «Архітектура обчислювальних систем», «Програмування», «Алгоритми і структури даних»
Що буде вивчатися	Принципи роботи операційних систем, способи керування ресурсами обчислювальної системи, технології взаємодії з прикладним програмним забезпеченням. Класифікація операційних систем для вирішення певних завдань та її налаштовування, керування розподіленими ресурсами обчислювальної системи.
Чому це цікаво/треба вивчати	Сучасний світ неможливо уявити без різних «гаджетів» та побутових пристроїв, всі вони працюють під керівництвом операційних систем. Важливо розуміти можливості та недоліки кожної з них для розв'язання прикладних задач.
Чому можна навчитися	- основи побудови операційних систем, їхньої архітектури, вимоги до них, історію їх розвитку і сучасні підходами до їх реалізації; - базовий склад компонентів операційної системи, основні функції ядра і системного програмного забезпечення; - методи і алгоритми керування локальними ресурсами комп'ютера: процесором, пам'яттю, пристроями введення-виведення, розподілюваними ресурсами; - способи і засоби розв'язання проблем синхронізації і взаємних блокувань у багатозадачних і багатопотокових операційних системах; - принципи реалізації файлових систем, структуру сучасних файлових систем; - проблеми реалізації мережних функцій операційних систем і способи організації віддаленого виклику процедур і розподілених файлових систем; - підходи до реалізації зазначених вище механізмів у сучасних операційних системах Linux, Mac і Windows;
Як можна користуватися набутими знаннями і уміннями	- здійснювати інсталяцію сучасних операційних системам Linux і Windows, в тому числі у віртуалізованих середовищах; - виконувати базові настроювання операційних систем і вирішувати задачі адміністрування їх; - використовуючи системні засоби розробляти сценарії для автоматизації задач адміністрування; - формулювати вимоги до операційної системи для вирішення певних прикладних завдань;
Інформаційне забезпечення дисципліни	Слайди презентацій, класичні підручники, методичні посібники, онлайн підручник, силабус.
Вид семестрового контролю	Залік

34. Навчання з підкріпленням	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	4, 7
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 54 аудиторних годин, 66 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Базові знання з дисциплін: Математична логіка та теорія алгоритмів, Алгоритми і структури даних, Випадкові процеси, Аналіз даних, Вступ до баз даних та інформаційних систем, Основи машинного навчання, Методи оптимізації
Що буде вивчатися	Парадигми, моделі та методи навчання з підкріпленням
Чому це цікаво/треба вивчати	Навчання з підкріпленням (Reinforcement Learning) є актуальним напрямком Машинного Навчання (Machine Learning), оскільки дозволяє проводити навчання моделей у режимі постійної адаптації до середовища і відсутності навчальної вибірки як такої. Це зумовлює використання навчання з підкріпленням» різних галузях, таких як автономні системи, робототехніка, фінанси, медицина, енергетика, Інтернет Речей.
Чому можна навчитися	Розуміти складові частини методів навчання з підкріпленням. Реалізовувати методи навчання з підкріплення для відповідної прикладної задачі.
Як можна користуватися набутими знаннями і уміннями	Здатність розв’язувати професійні задачі збору, аналізу та обробки даних в умовах реального часу та відсутності навчальної вибірки.
Інформаційне забезпечення дисципліни	Слайди презентацій, монографії, наукові статті, методичні посібники, онлайн підручники, силабус, матеріали для виконання лабораторних робіт, тестові завдання
Вид семестрового контролю	Залік

35. Навчання з підкріпленням в глибинному навчанні	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	4, 7
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 54 аудиторних годин, 66 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Базові знання з дисциплін: Математична логіка та теорія алгоритмів, Алгоритми і структури даних, Випадкові процеси, Аналіз даних, Вступ до баз даних та інформаційних систем, Основи машинного навчання, Системи глибинного навчання, Методи оптимізації
Що буде вивчатися	Парадигми, моделі та методи глибинного навчання з підкріпленням
Чому це цікаво/треба вивчати	Навчання з підкріпленням (Reinforcement Learning) використовується є актуальним напрямком в області Глибинного Навчання, оскільки дозволяє проводити навчання моделей у режимі постійної адаптації до середовища і відсутності навчальної вибірки як такої.
Чому можна навчитися	Розуміти складові частини методів глибинного навчання з підкріпленням. Реалізовувати методи навчання з підкріплення для відповідної прикладної задачі і відповідною архітектурою нейронної мережі.
Як можна користуватися набутими знаннями і уміннями	Здатність розв’язувати професійні задачі збору, аналізу та обробки даних в умовах реального часу та відсутності навчальної вибірки.
Інформаційне забезпечення дисципліни	Слайди презентацій, монографії, наукові статті, методичні посібники, онлайн підручники, силабус, матеріали для виконання лабораторних робіт, тестові завдання
Вид семестрового контролю	Залік

36. Методи пошуку аномалій	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	4, 7
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 54 аудиторних годин, 66 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Базові знання з дисциплін: Математична логіка та теорія алгоритмів, Алгоритми і структури даних, Випадкові процеси, Аналіз даних, Вступ до баз даних та інформаційних систем, Основи машинного навчання, Системи глибинного навчання, Методи оптимізації
Що буде вивчатися	Методи пошуку різних типів аномалій.
Чому це цікаво/треба вивчати	Актуальність пошуку аномалій (аномалійне виявлення) в різних галузях продовжує зростати, особливо в контексті великих обсягів даних та розвитку технологій штучного інтелекту. Ось кілька причин, які визначають актуальність цього напряму: Безпека Інформації: Аномалійне виявлення є ключовим елементом систем безпеки інформації. Виявлення незвичайних або підозрілих активностей може вказувати на можливі атаки або порушення безпеки. Фінанси та Шахрайство: В області фінансів актуальність пошуку аномалій полягає в виявленні шахрайських операцій, аномальних транзакцій та інших невідповідностей, що можуть вказувати на фінансові ризики. Медицина: В області медицини аномалійне виявлення може використовуватися для ранньої діагностики захворювань, виявлення аномальних паттернів у пацієнтських даних чи медичних зображеннях.
Чому можна навчитися	Розрізняти типи задач пошуку аномалій і розробляти необхідних метод пошук аномалій у відповідності до прикладної задачі
Як можна користуватися набутими знаннями і уміннями	Здатність розв’язувати професійні задачі збору, аналізу та обробки даних пов’язані з пошуком і виявленням аномалій в задачах інтелектуального аналізу даних
Інформаційне забезпечення дисципліни	Слайди презентацій, монографії, наукові статті, методичні посібники, онлайн підручники, силабус, матеріали для виконання лабораторних робіт, тестові завдання
Вид семестрового контролю	Залік

37. Системний аналіз	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	4, 8
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	опанування курсів теорії ймовірностей, математичної статистики, методів оптимізації, дослідження операцій, математичного моделювання
Що буде вивчатися	Принципи, процедури, методи та методології системного аналізу. Дослідження складних та надскладних систем. Системно-методологічні аспекти моделювання.
Чому це цікаво/треба вивчати	Одна з найактуальніших проблем нової епохи – проблема ефективного управління складними системами, зокрема, інформаційними системами. Завдання, що постають в процесі створення та функціонування складних та надскладних систем, вимагають володіння системним мисленням, спеціально розробленими методами та методологіями.
Чому можна навчитися	Знання предметної області та основних понять і методів системного аналізу. Уміння розкривати невизначеності; знаходити раціональний компроміс; проводити інформаційний аналіз системних задач і структурно-функціональний аналіз складних ієрархічних систем; передбачати та прогнозувати поведінку систем. Досвід аналізу систем, проектування та розробки додатків для розв'язання системних задач.
Як можна користуватися набутими знаннями і уміннями	Проєктувати та досліджувати складні системи. Створювати моделі складних систем. Діяти в умовах нестачі інформації, суперечливості інформації, надлишку інформації. Приймати складні рішення щодо функціонування систем.
Інформаційне забезпечення дисципліни	Онлайн підручники та курси лекцій, слайди презентацій, підручники, методичні посібники, силабус
Вид семестрового контролю	Залік

38. Основи теорії прийняття рішень	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	4, 8
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	опанування курсів теорії ймовірностей, математичної статистики, методів оптимізації, дослідження операцій, математичного моделювання
Що буде вивчатися	Принципи, процедури, методи теорії прийняття рішень. Прийняття рішень в складних та надскладних системах; в умовах нестачі інформації, суперечливості інформації, надлишку інформації.
Чому це цікаво/треба вивчати	Процеси прийняття рішень лежать в основі будь-якої цілеспрямованої діяльності людини. При проектуванні та розробці технологій побудови та експлуатації машин, приладів, пристроїв, будівель; в соціальній сфері для організації функціонування та розвитку соціальних процесів, їх координації з господарськими та економічними. Сьогодні для вироблення такого рішення потрібний науковий підхід – занадто великі втрати, пов'язані з помилками. Оптимальні рішення дозволяють досягти мети при мінімальних витратах трудових, матеріальних і сировинних ресурсів.
Чому можна навчитися	Знання предметної області та основних понять і методів теорії прийняття рішень. Уміння будувати моделі оптимізації рішень одночасно за кількома критеріями, визначати вигляд принципу оптимальності; розкривати невизначеності; знаходити раціональний компроміс. Досвід прийняття рішень, проектування та розробки додатків для підтримки прийняття рішень.
Як можна користуватися набутими знаннями і уміннями	Будувати моделі оптимізації рішень одночасно за кількома критеріями для складних та надскладних систем. Приймати складні рішення щодо функціонування систем. Приймати рішення в умовах нестачі інформації, суперечливості інформації, надлишку інформації. Розробляти додатки для підтримки прийняття рішень.
Інформаційне забезпечення дисципліни	Онлайн підручники та курси лекцій, слайди презентацій, підручники, методичні посібники, силабус
Вид семестрового контролю	Залік

39. Основи теорії інформаційних систем	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	4, 8
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	опанування курсів теорії ймовірностей, математичної статистики, методів оптимізації, дослідження операцій, математичного моделювання
Що буде вивчатися	Принципи, процедури, методи та методології побудови та дослідження складних та надскладних інформаційних систем. Архітектура та керування інформаційними системами
Чому це цікаво/треба вивчати	Сучасну епоху з повним правом можна назвати епохою складних систем. Інформаційні системи – складні організаційні, програмно-апаратні та телекомунікаційні комплекси. Їх проєктування, створення та управління вимагають володіння спеціально розробленими методами та методологіями.
Чому можна навчитися	Знання предметної області та основних понять і методів аналізу інформаційних систем (ІС). Уміння знаходити раціональний компроміс; проводити аналіз архітектур існуючих ІС та створювати нові архітектури та динамічні моделі ІС; передбачати та прогнозувати поведінку ІС. Досвід аналізу, проектування та розробки інформаційних систем.
Як можна користуватися набутими знаннями і уміннями	Проєктувати та досліджувати складні інформаційні системи. Створювати моделі складних та надскладних інформаційних систем. Діяти в умовах нестачі інформації, суперечливості інформації, надлишку інформації. Приймати складні рішення щодо функціонування інформаційних систем.
Інформаційне забезпечення дисципліни	Онлайн підручники та курси лекцій, слайди презентацій, підручники, методичні посібники, силабус
Вид семестрового контролю	Залік

40. Теорія оптимального керування	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	4, 8
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Базові знання з дисциплін: Математичний аналіз, лінійна алгебра, диференціальні рівняння, теорія функцій комплексної змінної, методи оптимізації, числові методи, алгоритми і системи комп'ютерної математики, програмування.
Що буде вивчатися	Теорія керування – це наука про принципи й методи спрямування функцій різних систем, процесів та об’єктів. Її сутність полягає в побудові математичної моделі об’єкта керування на основі його системного аналізу та в синтезі алгоритму для досягнення бажаних характеристик функціонування або цілей керування.
Чому це цікаво/треба вивчати	У даний час ця теорія слугує єдиною науковою основою розв’язування задач керування динамічними системами різної природи (фізичної, хімічної, біологічної та соціальної тощо) на базі розвинених математичних методів їх дослідження, зокрема їх аналізу та синтезу. Сучасна теорія керування є послідовницею кібернетики, науки що вивчає управління різноманітними системами живої та неживої природи на основі принципу зворотного зв’язку. Теорія керування моє глибокі зв’язки з теорією нейронних мереж.
Чому можна навчитися	Будувати моделі динамічних систем у вигляді диференційних або ітеративних рівнянь та систем рівнянь, ідентифікувати ці моделі на даних, будувати критерії їх оптимального функціонування, знаходити стратегії оптимального керування, чисельно розв’язувати задачі оптимального керування.
Як можна користуватися набутими знаннями і уміннями	Здатність розв’язувати професійні задачі проектування керованих технічних систем, в тому числі автономних керованих систем (роботів, дронів) в програмних середовищах математичного моделювання (Matlab, Octava, Scilab і т.і.).
Інформаційне забезпечення дисципліни	Конспекти лекцій, підручники та монографії. Силабус Новицький І.В. Сучасна теорія керування: навч. посіб. / І.В. Новицький, С.А. Ус, м-во освіти і науки України, Нац. гірн. ун-т. – Дніпро : НГУ, 2017. – 263 с. Пічкур В. В. Лекції з теорії керування. Київ: Київський національний університет імені Тараса Шевченка, Факультет комп’ютерних наук та кібернетики, 2017. – 232 с. Paluszek M., Thomas S. MATLAB Machine Learning Recipes: A Problem-Solution Approach. 2nd ed. APRESS, 2019.
Вид семестрового контролю	Залік

41. Теорія автоматичного керування	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	4, 8
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Базові знання з дисциплін: Математичний аналіз, лінійна алгебра, диференціальні рівняння, теорія функцій комплексної змінної, методи оптимізації, числові методи, алгоритми і системи комп'ютерної математики, програмування.
Що буде вивчатися	Теорія автоматичного керування (ТАК) - наукова дисципліна, предметом вивчення якої є системи, які складаються з об’єкта та пристрою керування (автоматичного регулятора) і допоміжних елементів. ТАК виявляє загальні закономірності функціонування, які притаманні автоматичним системам різної природи, і на основі цього розробляє принципи побудови ефективних систем для керування об’єктами різного призначення. При цьому широко використовуються математичні методи – теорія диференціальних рівнянь, теорія функцій комплексної змінної, перетворення Лапласа і Фур’є, матриці і т.д.
Чому це цікаво/треба вивчати	Автоматизація виробництва – один з важливих факторів науково-технічного процесу, за рахунок якої відбувається інтенсифікація процесів випуску продукції та зменшення витрат енергетичних та матеріальних ресурсів. В теорії автоматичного керування розглядаються також питання оптимізації технологічних режимів, а також можливості пристосування (адаптації) систем до змінюваних умов роботи.
Чому можна навчитися	Теорія автоматичного керування – одна з навчальних дисциплін, яка формує у студентів знання про загальні принципи і процеси функціонування автоматичних систем керування. Мета дисципліни – формування у студентів знань, умінь та навичок з розв’язання задач аналізу та синтезу автоматичних систем керування за умов діяння детермінованих та випадкових збурень.
Як можна користуватися набутими знаннями і уміннями	Здатність розв’язувати професійні задачі проектування автоматичних керованих технічних систем, в тому числі автономних керованих систем (роботів, дронів) в програмних середовищах математичного моделювання (Matlab, Simulink, Octava, Scilab і т.і.).
Інформаційне забезпечення дисципліни	Конспекти лекцій, підручники та монографії. Силабус Іванов А. О. Теорія автоматичного керування: Підручник. — Дніпропетровськ: Національний гірничий університет. — 2003. — 250 с. Новицький І.В. Сучасна теорія керування: навч. посіб. / І.В. Новицький, С.А. Ус, м-во освіти і науки України, Нац. гірн. ун-т. – Дніпро : НГУ, 2017. – 263 с.
Вид семестрового контролю	Залік

42. Теорія динамічної оптимізації	
Кафедра, яка забезпечує викладання	прикладної математики
Рівень вищої освіти	Перший (бакалаврський)
Курс, семестр	4, 8
Обсяг дисципліни та розподіл годин аудиторної та самостійної роботи	4 кредити ЄКТС: 72 аудиторні години, 48 годин самостійної роботи
Мова викладання	Українська
Вимоги до початку вивчення дисципліни	Базові знання з дисциплін: Математичний аналіз, лінійна алгебра, диференціальні рівняння, теорія функцій комплексної змінної, методи оптимізації, числові методи, алгоритми і системи комп'ютерної математики, програмування.
Що буде вивчатися	Теорія динамічної оптимізації – це наука про оптимізацію поведінки динамічних систем, що описуються диференційними або ітеративними рівняннями руху. Ця теорія є складовою частиною теорії оптимізації у багатовимірних або нескінченно-вимірних просторах (гільбертових або банахових просторах).
Чому це цікаво/треба вивчати	Теорія нескінченно-вимірної оптимізації є теоретичною математичною основою теорії оптимального керування.
Чому можна навчитися	Аналізувати нескінченно-вимірні задачі оптимізації, апроксимувати ці задачі скінчено-вимірними, досліджувати збіжність апроксимацій та будувати методи розв’язання таких задач.
Як можна користуватися набутими знаннями і уміннями	Здатність обґрунтовувати постановки задач оптимізації динамічних систем, досліджувати властивості задач, розробляти методи їх розв’язання з урахуванням структури систем, публікувати результати теоретичних досліджень у фахових виданнях.
Інформаційне забезпечення дисципліни	Конспекти лекцій, підручники та монографії. Силабус Моклячук, М. П. Варіаційне числення. Екстремальні задачі. Підручник. – К. : Видавничо-поліграфічний центр "Київський університет", 2009. – 380 с. Пічкур В. В. Лекції з теорії керування. Київ: Київський національний університет імені Тараса Шевченка, Факультет комп’ютерних наук та кібернетики, 2017. – 232 с.
Вид семестрового контролю	Залік