



Основи програмування мовою Python

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	17 Електроніка та телекомунікації
Спеціальність	172 Телекомунікації та радіотехніка
Освітня програма	Інформаційно-комунікаційні технології
Статус дисципліни	Вибіркова
Форма навчання	очна(денна)
Рік підготовки, семестр	рік четвертий, весняний семестр
Обсяг дисципліни	4 кредити ЄКТС, з них лекції 36 годин, практичні заняття 18 годин
Семестровий контроль/ контрольні заходи	Модульна контрольна робота Залік
Розклад занять	Згідно з розкладом
Мова викладання	Українська
Інформація про керівника курсу / викладачів	Лектор: к.т.н. Педан С.І., 066-120-04-65, stas.pedan@gmail.com Практичні заняття: к.т.н. Педан С.І.
Розміщення курсу	https://classroom.google.com/u/1/c/NjUwNjU3NzcxMjY5

Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Під час вивчення дисципліни студентам надаються знання базового синтаксису мови Python, навички роботи зі змінними, колекціями даних (списки, словники, кортежі, множини), функціями, класами, написання циклів та операторів розгалуження, робота з класами та обробки файлів. Важливим аспектом дисципліни є отримання навичок об'єктно-орієнтованого програмування та тестування на мові Python в результаті виконання серії самостійних практичних робіт.

Вивчення дисципліни базується на знанні студентами матеріалів, передбачуваних навчальними планами дисциплін:

- «Прикладне програмування в інформаційно-комунікаційних системах»;
- «Інформатика-1»;
- «Інформатика-2» та інші.

Дисципліна забезпечує студентам вивчення таких дисциплін:

- «Технології побудови web-орієнтованих систем»;
- «Системне проектування телекомунікаційних мереж»;
- «Інтелектуальна обробка інформації»;

- «Моделювання в інфо-комунікаційних системах»;
- «Технології інтернет»;
- «Програмування мікрокомп'ютерних систем обміну даних»;
- «Сучасні технології програмування».

Цілі дисципліни	<p>Метою навчальної дисципліни є формування у студентів знань з загальних питань, які використовуються для забезпечення наступних процесів телекомунікацій:</p> <p>підготовка фахівця, який має базові компетенції, що засновані на системи знань в області системного проектування у системах, обладнанні і програмному забезпеченні систем телекомунікацій;</p> <p>формування та розвиток загальних і професійних компетентностей з впровадження та застосування технологій телекомунікацій і радіотехніки, що сприяють соціальній стійкості та мобільності випускника на ринку праці;</p> <p>формування у здобувачів освіти логічного мислення, розвиток їх інтелекту та здібностей;</p> <p>формування знань, вмінь і навичок, необхідних для розуміння сучасних технологій програмування телекомунікаційних систем та мереж.</p>
------------------------	--

Згідно з вимогами освітньо-професійної програми студенти після засвоєння навчальної дисципліни мають продемонструвати такі результати навчання:

знання:

Здатність використовувати базові методи, способи та засоби отримання, передавання, обробки та зберігання інформації (ФК-3);

Здатність здійснювати комп'ютерне моделювання пристроїв, систем і процесів з використанням універсальних пакетів прикладних програм (ФК-4);

Здатність здійснювати розробку, тестування, використання та супровід спеціалізованого програмного забезпечення з дотриманням атрибутів якості, програмування прикладних задач, створення WEB-сайтів, моделювання та віртуалізацію інфокомунікаційних процесів, систем, мереж (ФК-16);

уміння:

Мати навички оцінювання, інтерпретації та синтезу інформації і даних (ПРН5);

Описувати принципи та процедури, що використовуються в телекомунікаційних системах, інформаційно-телекомунікаційних мережах та радіотехніці (ПРН8);

Знаходити, оцінювати і використовувати інформацію з різних джерел, необхідну для розв'язання професійних завдань, включаючи відтворення інформації через електронний пошук (ПРН18);

Здійснювати аналіз вимог до програмного забезпечення, створювати сценарії для його тестування, програмувати прикладні задачі, створювати власні математичні моделі, програмні модулі для структурування даних, аналізу, моделювання, віртуалізації, розробки та оптимізації інфокомунікаційних засобів та процесів (ПРН23).

2. Пререквізити та постреквізити дисципліни

Перелік дисциплін або знань та умінь, володіння якими необхідні здобувачу вищої освіти для успішного засвоєння дисципліни	Перелік дисциплін, які базуються на результатах навчання з даної дисципліни
Дисципліна вивчається на основі ступеня бакалавра базуючись на знаннях таких дисциплін: - «Прикладне програмування в інформаційно-комунікаційних системах»; - «Інформатика-1»; - «Інформатика-2» та інші.	- «Технології побудови web-орієнтованих систем»; - «Системне проектування телекомунікаційних мереж»; - «Інтелектуальна обробка інформації»; - «Моделювання в інфо-комунікаційних системах»; - «Технології інтернет»; - «Програмування мікрокомп'ютерних систем обміну даних»; - «Сучасні технології програмування».

3. Зміст навчальної дисципліни

Тема 1. Базові типи та структури даних

- 1.1 Властивості Python
- 1.2 Історія розробки Python
- 1.3 Сфера використання мови
- 1.4 Переваги та недоліки
- 1.5 Способи написання та запуску коду Python
- 1.6 Базові типи даних
- 1.7 Оператор ділення з остачею та зведення в ступінь
- 1.8 Динамічна типізація
- 1.9 Робота з символьними рядками
- 1.10 Списки (list)
- 1.11 Словники (dict)
- 1.12 Кортежі (tuple)
- 1.13 Множини (set)
- 1.14 Логічні дані (bool)
- 1.15 Операції порівняння

Тема 2. Реалізація функцій

- 2.1 Умовне розгалуження
- 2.2 Цикл for
- 2.3 Цикл while
- 2.4 break, continue, pass
- 2.5 Функції range, enumerate, zip
- 2.6 Модуль random
- 2.7 Генерація списків
- 2.8 Вбудовані методи
- 2.9 Структура функції
- 2.10 Функція обробки списку
- 2.11 Взаємодія функцій
- 2.12 Змінна кількість вхідних аргументів функції

- 2.13 Функція `map`
- 2.14 Функція `filter`
- 2.15 Лямбда функції
- 2.16 Видимість змінних

Тема 3. Об'єктно-орієнтоване програмування

- 3.1 Методології програмування
- 3.2 ООП в Python
- 3.3 Реалізація класу
- 3.4 Атрибути класу
- 3.5 Ініціалізація об'єкту
- 3.6 Методи класу
- 3.7 Реалізація наслідування
- 3.8 Реалізація інкапсуляції
- 3.9 Реалізація поліморфізму
- 3.10 Мова моделювання UML
- 3.11 Діаграма класів
- 3.12 Діаграма послідовності
- 3.13 Діаграма варіантів використання
- 3.14 Використання PlantUML для проектування дизайну архітектури

Тема 4. Модулі та пакети

- 4.1 Визначення винятку
- 4.2 Сценарії обробки винятків
- 4.3 Типи винятків
- 4.4 Обробка винятків з `try..except`
- 4.5 Генерація винятків
- 4.6 Системи контролю версій та Git
- 4.7 Збереження історії змін (`add`, `commit`)
- 4.8 Розгалуження розробки (`branch`, `checkout`, `merge`)
- 4.9 Платформа GitHub (`push`, `pull`, `fetch`, `pull request`, `fork`)
- 4.10 Робота з модулями
- 4.11 Пакети Python
- 4.12 PyPI репозиторій
- 4.13 Пошук та встановлення пакетів
- 4.14 Віртуальне середовище
- 4.15 Файли `__init__`, `__about__`, `__main__`
- 4.16 Реалізація пакету головного коду
- 4.17 Реалізація модульних тестів
- 4.18 Встановлення пакету з тестами
- 4.19 Запуск пакету та модульних тестів
- 4.20 Доступ до файлу
- 4.21 Читання файлу
- 4.22 Запис у файл
- 4.23 Робота з директоріями
- 4.24 Приклад пакету по перетворенню pdf у txt

4. Навчальні матеріали та ресурси

4.1 Базова література:

1. Основи програмування. Python. Частина 1 [Електронний ресурс]: підручник для студ. спеціальності 122 "Комп'ютерні науки", спеціалізації "Інформаційні технології в біології та медицині" / А. В. Яковенко ; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 1,59 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2018. – 195 с.
2. Копей В. Б. Мова програмування Python для інженерів і науковців : навчальний посібник / В. Б. Копей – ІваноФранківськ : ІФНТУНГ, 2019. – 272 с
3. Объектно-ориентированный анализ и проектирование. С примерами приложений. Буч Гради, Максимчук Роберт А., Энгел Майкл У., Янг Бобби Дж., Коаллен Джим, Хьюстон Келли А.: Пер. с англ. – 3-е изд. – М.: ООО «И.Д.Вильямс», 2008. – 720 с.
4. Фаулер М. UML. Основы. Краткое руководство по стандартному языку объектного моделирования [Текст] / М. Фаулер; пер. с англ. А. Петухова. - М.: Символ-Плюс, 2011. - 192 с.
5. Gamma, Erich, Helm, Richard, Johnson, Ralph and Vlissides, John M.. Design Patterns: Elements of Reusable Object-Oriented Software. 1 : Addison-Wesley Professional, 1994.
6. Scott Chacon, Ben Straub, "Pro Git. Second edition" (2014). Apress, <https://git-scm.com/book/uk/v2>
7. Bobby Iliev, "Introduction to Git and GitHub" (2023). <https://github.com/bobbyiliev/introduction-to-git-and-github-ebook>

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

5.1 Лекційні заняття – 36 годин

Тема 1. Базові типи та структури даних

Лекція 1. Вступ до дисципліни та знайомство з Python

Основні питання:

1. Структура та ціль дисципліни
2. Властивості Python
3. Історія розробки Python
4. Сфера використання мови
5. Переваги та недоліки
6. Способи написання та запуску коду Python

Дидактичні засоби: Слайди за темою лекції, навчально-методичні матеріали за темою лекції.

Література: [1]

Завдання на СРС: підготувати відповідь на контрольні запитання до л. 1.

Лекція 2. Базові типи даних Python та операції над ними

Основні питання:

1. Базові типи даних
2. Оператор ділення з остачею та зведення в ступінь
3. Динамічна типізація
4. Робота з символьними рядками

Дидактичні засоби: Слайди за темою лекції, навчально-методичні матеріали за темою лекції.

Література: [1-2].

Завдання на СРС: підготувати відповідь на контрольні запитання до л. 2.

Лекція 3. Базові структури даних Python (списки, словники, кортежі, множини, логічні дані)

Основні питання:

1. Списки (list)
2. Словники (dict)
3. Кортежі (tuple)

4. Множини (set)
5. Логічні дані (bool)
6. Операції порівняння

Дидактичні засоби: Слайди за темою лекції, навчально-методичні матеріали за темою лекції.

Література: [1-2].

Завдання на СРС: підготувати відповідь на контрольні запитання до л. 3.

Тема 2. Реалізація функцій

Лекція 4. Оператори Python (умовного розгалуження та циклу, функція range)

Основні питання:

1. Умовне розгалуження
2. Цикл for
3. Цикл while
4. break, continue, pass
5. Функції range, enumerate, zip

Дидактичні засоби: Комп'ютерний проектор з набором експозицій до л. 4.

Література: [2].

Завдання на СРС: підготувати відповідь на контрольні запитання до л. 4.

Лекція 5. Робота з функціями

Основні питання:

1. Модуль random
2. Генерація списків
3. Вбудовані методи
4. Структура функції
5. Функція обробки списку

Дидактичні засоби: Комп'ютерний проектор з набором експозицій до л. 5.

Література: [2].

Завдання на СРС: підготувати відповідь на контрольні запитання до л. 5

Лекція 6. Взаємодія функцій

Основні питання:

1. Взаємодія функцій
2. Змінна кількість вхідних аргументів функції
3. Функція map
4. Функція filter
5. Лямбда функції
6. Видимість змінних

Дидактичні засоби: Комп'ютерний проектор з набором експозицій до л. 6.

Література: [2].

Завдання на СРС: підготувати відповідь на контрольні запитання до л. 6.

Тема 3. Об'єктно-орієнтоване програмування

Лекція 7. Методології програмування та основи ООП

1. Методології програмування
2. ООП в Python
3. Реалізація класу
4. Атрибути класу

Дидактичні засоби: Комп'ютерний проектор з набором експозицій до л. 7.

Література: [3].

Завдання на СРС: підготувати відповідь на контрольні запитання до л. 7.

Лекція 8. Реалізація принципів ООП та мова UML

1. Ініціалізація об'єкту
2. Методи класу
3. Реалізація наслідування
4. Реалізація інкапсуляції
5. Реалізація поліморфізму

Дидактичні засоби: Комп'ютерний проектор з набором експозицій до л. 8.

Література: [4].

Завдання на СРС: підготувати відповідь на контрольні запитання до л. 8.

Лекція 9. Проектування дизайну системи та мова моделювання UML

1. Мова моделювання UML
2. Діаграма класів
3. Діаграма послідовності
4. Діаграма варіантів використання
5. Використання PlantUML для проектування дизайну архітектури

Дидактичні засоби: Комп'ютерний проектор з набором експозицій до л. 9.

Література: [4-5].

Завдання на СРС: підготувати відповідь на контрольні запитання до л. 9.

Тема 4. Модулі та пакети

Лекція 10. Обробка винятків

1. Визначення винятку
2. Сценарії обробки винятків
3. Типи винятків
4. Обробка винятків з try..except
5. Генерація винятків

Дидактичні засоби: Комп'ютерний проектор з набором експозицій до л. 10.

Література: [2].

Завдання на СРС: підготувати відповідь на контрольні запитання до л. 10.

Лекція 11. Система контролю версій Git та платформа GitHub

1. Системи контролю версій та Git
2. Збереження історії змін (add, commit)
3. Розгалуження розробки (branch, checkout, merge)
4. Платформа GitHub (push, pull, fetch, pull request, fork)

Дидактичні засоби: Комп'ютерний проектор з набором експозицій до л. 11.

Література: [6-7].

Завдання на СРС: підготувати відповідь на контрольні запитання до л. 11.

Лекція 12. Модулі та пакети

1. Робота з модулями
2. Пакети Python
3. PyPI репозиторій
4. Пошук та встановлення пакетів
5. Віртуальне середовище

Дидактичні засоби: Комп'ютерний проектор з набором експозицій до л. 12.

Література: [2].

Завдання на СРС: підготувати відповідь на контрольні запитання до л. 12.

Лекція 13. Практика розробки пакету Python

1. Файли `__init__`, `__about__`, `__main__`
2. Реалізація пакету головного коду
3. Реалізація модульних тестів
4. Встановлення пакету з тестами
5. Запуск пакету та модульних тестів

Дидактичні засоби: Комп'ютерний проектор з набором експозицій до л. 12.

Література: [2].

Завдання на СРС: підготувати відповідь на контрольні запитання до л. 12.

Лекція 14. Робота з файлами

1. Доступ до файлу
2. Читання файлу
3. Запис у файл
4. Робота з директоріями
5. Приклад пакету по перетворенню pdf у txt

Дидактичні засоби: Комп'ютерний проектор з набором експозицій до л. 14.

Література: [2].

Завдання на СРС: підготувати відповідь на контрольні запитання до л. 14.

5.2 Практичні заняття – 18 годин

Основні завдання циклу практичних занять ознайомитися з основними поняттями курсу Основ програмування мовою Python, та навчитися застосовувати ці поняття для розробки телекомунікаційних об'єктів та систем.

Тема 1. Базові типи та структури даних

Практичне заняття 1. Базові типи даних Python та операції над ними.

Основні питання:

1. Базові типи даних
2. Оператор ділення з остачею та зведення в ступінь
3. Динамічна типізація
4. Робота з символьними рядками

Практичне заняття 2. Базові структури даних Python.

1. Списки (list)
2. Словники (dict)
3. Кортежі (tuple)
4. Множини (set)
5. Логічні дані (bool)
6. Операції порівняння

Тема 2. Реалізація функцій

Практичне заняття 3. Оператори Python.

Основні питання:

1. Умовне розгалуження
2. Цикл for
3. Цикл while
4. break, continue, pass
5. Функції range, enumerate, zip

Практичне заняття 4. Робота з функціями та їх взаємодія.

Основні питання:

1. Модуль random
2. Генерація списків
3. Вбудовані методи
4. Структура функції
5. Функція обробки списку
6. Взаємодія функцій
7. Змінна кількість вхідних аргументів функції
8. Функція map
9. Функція filter
10. Лямбда функції
11. Видимість змінних

Тема 3. Об'єктно-орієнтоване програмування

Практичне заняття 5. Робота з об'єктами та класами, реалізація принципів ООП

Основні питання:

1. ООП в Python
2. Реалізація класу
3. Атрибути класу
4. Ініціалізація об'єкту
5. Методи класу
6. Реалізація наслідування
7. Реалізація інкапсуляції
8. Реалізація поліморфізму

Практичне заняття 6. Проектування дизайну системи та мова моделювання UML.

Основні питання:

1. Мова моделювання UML
2. Діаграма класів
3. Діаграма послідовності
4. Діаграма варіантів використання
5. Використання PlantUML для проектування дизайну архітектури

Тема 4. Модулі та пакети

Практичне заняття 7. Обробка винятків.

Основні питання:

1. Сценарії обробки винятків
2. Типи винятків
3. Обробка винятків з try..except
4. Генерація винятків

Практичне заняття 8. Поведінкові шаблони проектування ООП

Основні питання:

1. Системи контролю версій та Git
2. Збереження історії змін (add, commit)
3. Розгалуження розробки (branch, checkout, merge)
4. Платформа GitHub (push, pull, fetch, pull request, fork)

Практичне заняття 9. Огляд засобів програмування мобільних пристроїв на ОС Android

Основні питання:

1. Робота з модулями
2. Пакети Python
3. PyPI репозиторій
4. Пошук та встановлення пакетів

5. Віртуальне середовище
6. Файли `__init__`, `__about__`, `__main__`
7. Реалізація пакету головного коду
8. Реалізація модульних тестів
9. Встановлення пакету з тестами
10. Запуск пакету та модульних тестів

6. Самостійна робота студента

До самостійної роботи студентів включається підготовка до аудиторних занять шляхом опанування матеріалів лекцій, вивчення базової, додаткової літератури та виконання практичних завдань.

Тема 1. Базові типи та структури даних

Самостійна практична робота 1. Базові типи, структури даних та оператори

Ціль: отримання навичок використання базових типів та структур даних, а також роботи з операторами Python

Завдання: реалізація гри «Вгадай число» у відповідності по приведених правил та обмежень

Результат виконання завдання: звіт з програмним кодом реалізації завдання, а також скріншоти або історія повідомлень найуспішнішої та найдовшої реалізованої гри.

Тема 2. Реалізація функцій

Самостійна практична робота 2. Реалізація функцій та їх взаємодії

Ціль: отримання навичок реалізації функцій та їх організації їх взаємодії на мові Python

Завдання: реалізація гри «Хрестики-нулики» у відповідності по приведених правил та обмежень

Результат виконання завдання: звіт з програмним кодом реалізації завдання, а також скріншоти або історія повідомлень реалізованої гри

Тема 3. Об'єктно-орієнтоване програмування

Самостійна практична робота 3. Об'єктно-орієнтоване програмування

Ціль: отримання навичок проектування об'єктно-орієнтованого дизайну, а також використання принципів ООП при реалізації програм на мові Python

Завдання: реалізація ієрархії класів з предметної області, які пов'язані відношеннями частина-ціле з використанням шаблону «Компонувальник»

Результат виконання завдання: звіт з програмним кодом реалізації завдання, діаграма класів, а також скріншоти або історія повідомлень роботи реалізованої ієрархії класів

Тема 4. Модулі та пакети

Самостійна практична робота 4. Пакети та модулі Python

Ціль: отримання навичок розробки модулів та пакетів Python

Завдання: реалізація проекту з пакетом Python, який буде реалізовувати чотири математичні алгоритми, а також модульних тестів до них.

Результат виконання завдання: звіт з програмним кодом реалізації завдання, а також скріншоти або історія повідомлень роботи реалізованого пакету Python.

Політика та контроль

7. Політика навчальної дисципліни (освітнього компонента)

Дисципліна вивчається студентами в ході відвідування і прослуховування лекцій, розв'язання задач на практичних заняттях та виконання домашніх практичних робіт. В ході вивчення дисципліни передбачені поточні контрольні заходи результатів навчання у вигляді

опитування та тестування, модульна контрольна робота і підсумковий семестровий контроль у формі заліку. Поточні контрольні заходи здійснюються в основному у формі електронного тестування та опитування на практичних заняттях і в ході перевірки виконання домашніх практичних робіт. Підсумковий семестровий контроль проводиться у формі заліку. Загальна оцінка засвоєння студентами тем початкової дисципліни виставляється на підставі загального рейтингу кожного студента відповідно до рейтингової системи оцінювання результатів навчання.

8. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

8.1. Рейтинг студента з кредитного модуля розраховується виходячи із 100-бальної шкали і визначається за результатами виконання набору контрольних заходів (п.8.2.).

8.2. Контрольні заходи:

Лекційні заняття	– бал за роботу на лекції та написання конспекту – 1; За семестр проводиться 18 лекційних занять, кожен студент має можливість роботати на кожному з занять Максимальна кількість балів: $18 \times 1 = 18$
Практичні заняття	– бал за роботу на практичному занятті – 2; За семестр проводиться 9 практичних занять, кожен студент має можливість роботати на кожному з занять Максимальна кількість балів: $2 \times 9 = 18$. – бал за виконання аудиторного практичного завдання – 2; За семестр видається 5 аудиторних практичних завдань. Максимальна кількість балів: $2 \times 5 = 10$.
Самостійні практичні роботи	– бал за одну повну вірно виконану практичну роботу зі звітом в поставлені терміни – 5; – бал за виконану практичну роботу з запізненням – 2.5; Максимальна кількість балів: $5 \times 4 = 20$.
Модульна контрольна робота (МКР)	– бал за повну відповідь на всі запитання модульної контрольної роботи №1 – 14; – бал за повну відповідь на всі запитання модульної контрольної роботи №2 – 20; – бал за відсутність відповіді чи за неправильну відповідь на всі запитання – 0; Максимальна кількість балів: 34.

8.3. Підсумковий семестровий контроль проводиться у формі заліку з виставленням у відомість оцінки кожному із студентів на підставі поточного загального рейтингу студента, або проведення залікової контрольної роботи (співбесіди). Залікова контрольна робота (співбесіда) оцінюється із 100 балів. Контрольне завдання цієї роботи (співбесіди) складається з п'яти запитань з переліку, що надається для підготовки до заліку.

Кожне запитання оцінюється з 20 балів за такими критеріями:

- «відмінно» – повна відповідь (не менше 90% потрібної інформації), надані відповідні обґрунтування – 18-20 балів;
- «добре» – достатньо повна відповідь (не менше 75% потрібної інформації), є незначні неточності – 13-17 балів;
- «задовільно» – неповна відповідь (не менше 60% потрібної інформації), є деякі помилки – 7-12 балів;
- «незадовільно» – незадовільна відповідь – 0 балів.

8.4. Календарна проміжна атестація студентів проводиться на підставі поточного рейтингу студента на час проведення атестації. Якщо значення цього рейтингу не менше 30% від максимально можливого на час атестації, студент вважається атестованим. Умовою позитивної першої атестації є отримання не менше 10 балів, другої атестації – отримання не менше 50 балів.

8.5. Сума рейтингових балів, отриманих студентом протягом семестру, за умови зарахування МКР та всіх практичних робіт, переводиться до підсумкової оцінки згідно з таблицею

(п.8.7.). Якщо сума балів менша за 60, студент виконує залікову контрольну роботу (співбесіду). Умовою допуску до залікової контрольної роботи (співбесіди) при зарахуванні МКР та всіх лабораторних робіт, є сума рейтингових балів не менша 33.

8.6. Студент, який у семестрі отримав більше 60 балів, але бажає підвищити свій результат, може взяти участь у заліковій контрольній роботі (пройти співбесіду).

8.7. Таблиця переведення рейтингових балів до оцінок:

<i>Кількість балів</i>	<i>Оцінка</i>
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно
Не виконані умови допуску	Не допущено

9. Політика університету

9.1. Політика щодо академічної доброчесності

Політика та принципи академічної доброчесності визначені у розділі 3 Кодексу честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». Детальніше: <https://kpi.ua/code>

9.2. Норми етичної поведінки

Норми етичної поведінки студентів і працівників визначені у розділі 2 Кодексу честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». Детальніше: <https://kpi.ua/code>

10. Додаткова інформація з дисципліни (освітнього компонента)

Робочу програму навчальної дисципліни (силабус):

Складено к.т.н. Педан С.І.

Ухвалено кафедрою ІТТ НН ІТС (протокол №4 від 03 листопада 2023 року)