



ПРОГРАМУВАННЯ

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	<i>G Інженерія, виробництво та будівництво</i>
Спеціальність	<i>G7 Автоматизація, комп'ютерно-інтегровані технології та робототехніка</i>
Освітня програма	<i>Комп'ютерно-інтегровані технології та робототехніка</i>
Статус дисципліни	<i>Нормативна</i>
Форма навчання	<i>очна (денна)</i>
Рік підготовки, семестр	<i>Освітній компонент ПО 03.1.: 1 курс, 1 семестр (осінній семестр) Освітній компонент ПО 03.2.: 1 курс, 2 семестр (весняний семестр)</i>
Обсяг дисципліни	<i>10 кр. (300 год.) (5 кр. – I семестр, 5 кр. – II семестр)</i>
Семестровий контроль/ контрольні заходи	<i>Екзамен / МКР</i>
Розклад занять	<i>Згідно з розкладом на сайті https://schedule.kpi.ua/</i>
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	<i>Лектори: к.т.н., доц. Павловський Олексій Михайлович, a_pav[@]ukr.net, к.т.н., доц. Галаган Роман Михайлович r.galagan[@]kpi.ua, Практичні : к.т.н., доц. Павловський Олексій Михайлович к.т.н., доц. Лакоза Сергій Леонідович к.т.н., асистент Рупіч Сергій Сергійович к.т.н., доцент Сапегін Олександр Миколайович к.т.н., доцент Цибульник Сергій Олексійович ст. викладач Кравченко Ігор Володимирович к.т.н., доцент Галаган Роман Михайлович к.т.н., доцент Лисенко Юлія Юріївна асистент Драчук Олеся Олександрівна</i>
Розміщення курсу	<i>https://do.ipk.kpi.ua/course/view.php?id=2342 (ПО 3.1.) https://classroom.google.com/c/MTQ1OTE5MzE5NDc3 (ПО 3.2.)</i>

Програма навчальної дисципліни

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

В процесі освоєння навчальної дисципліни «Програмування» студенти отримують знання про сучасні комп'ютерні технології та програмування, методологію їх використання в практичній діяльності та здобувають навички роботи з комп'ютером та програмними продуктами, а також, створення власного програмного забезпечення для ОС Windows з використанням мов програмування C/C++ та Python.

Навчальна дисципліна Програмування належить до циклу професійної підготовки. Знання, які отримують студенти, вивчаючи цю дисципліну, можуть використовуватися у подальшому під час розробки алгоритмів функціонування комп'ютеризованих систем автоматизації, контролю, управління, діагностики, вимірювання; під час розробки математичного та прикладного програмного забезпечення вказаних систем.

Дисципліна «Програмування» складається із двох освітніх компонентів:

- *ПО 03.1. Програмування. Частина 1. Основи алгоритмізації та структурне програмування.*
- *ПО 03.2. Програмування. Частина 2. Об'єктно-орієнтоване програмування.*

Метою кредитного модуля є формування у студентів здатностей:

- *Здатність застосовувати знання у практичних ситуаціях.*
- *Навички використання інформаційних і комунікаційних технологій.*
- *Здатність використовувати для вирішення професійних завдань новітні технології у галузі автоматизації та комп'ютерно-інтегрованих технологій, зокрема, проектування багаторівневих систем керування, збору даних та їх архівування для формування бази даних параметрів процесу та їх візуалізації за допомогою засобів людино-машинного інтерфейсу.*
- *Здатність обґрунтовувати вибір технічної структури та вміти розробляти прикладне програмне забезпечення для мікропроцесорних систем керування на базі локальних засобів автоматизації, промислових логічних контролерів та програмованих логічних матриць і сигнальних процесорів.*
- *Здатність вільно користуватись сучасними комп'ютерними та інформаційними технологіями для вирішення професійних завдань, програмувати та використовувати прикладні та спеціалізовані комп'ютерно-інтегровані середовища для вирішення задач автоматизації.*

Основні завдання кредитного модуля.

Після засвоєння кредитного модуля студенти мають продемонструвати такі результати навчання:

- *Вміти застосовувати сучасні інформаційні технології та мати навички розробляти алгоритми та комп'ютерні програми з використанням мов високого рівня та технологій об'єктно-орієнтованого програмування, створювати бази даних та використовувати інтернет-ресурси.*

2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

Вивчення дисципліни базується на знаннях отриманих на попередньому рівні освіти. Знання, які отримують студенти при вивченні курсу "Програмування", використовуються у подальшому під час вивчення курсу "Технології розроблення програмного забезпечення", "Електроніка", "Основи цифрової схемотехніки", "Робототехніка" в курсовому і дипломному проектуванні, у практичній діяльності після закінчення навчання.

3. Зміст навчальної дисципліни

Розділ 1. Основи програмування та алгоритмізації задач.

o *Тема 1.1. Види мов програмування. Принципи розробки програм.*

Тема 1.2 Створення алгоритмів.

o *Тема 1.3. Системи числення, операції із числами, переведення із однієї системи в іншу*

Розділ 2. Алгоритмічна мова C++.

o *Тема 2.1. Типи даних і оператори.*

o *Тема 2.2. Базові конструкції структурного програмування.*

o *Тема 2.3. Функції.*

o *Тема 2.4. Масиви і масиви – рядки*

o *Тема 2.5. Вказівники.*

o Тема 2.6. Типи даних, що визначаються користувачем.

• **Розділ 3. Об'єктно-орієнтоване програмування. Мова Python.**

- o Тема 3.1. Основні відомості про мову програмування Python. Встановлення інтерпретатора Python, додаткових модулів та середовищ розробки IDE.
- o Тема 3.2. Основні характеристики мови програмування Python. Синтаксис. Типи даних. Змінні. Оператори.
- o Тема 3.3. Алгоритмічні структури Python.
- o Тема 3.4. Функції та модулі.
- o Тема 3.5. Парадигма об'єктно-орієнтованого програмування. Реалізація класів та екземплярів класів у Python. Інкапсуляція. Спадкування. Поліморфізм. Принципи SOLID.
- o Тема 3.6. Розробка програм з графічним інтерфейсом користувача із використанням бібліотеки Tkinter.

4. Навчальні матеріали та ресурси

Базова

1. Програмування. Комп'ютерний практикум [Електронний ресурс] : навч. посіб. для студ. спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології» / О.М. Павловський, С.Л. Лакоза, С.С. Рупіч; КПІ ім. Ігоря Сікорського, 2022. — 100 с.
2. Путівник мовою програмування Python [Електронний ресурс] – Режим доступу до ресурсу: <https://pythonguide.rozh2sch.org.ua/>

Додаткова

3. Paul Deitel, Deitel & Associates, Inc. C++ How to Program, 10th Edition. - Published by PEARSON INDIA (2017) ISBN 10: 9332585733 ISBN 13: 9789332585737
4. Ghitlur Swaroop. A Byte of Python [Електронний ресурс]. – Режим доступу до ресурсу: <https://python.swaroopch.com/>
5. Копей В. Б. Мова програмування Python для інженерів і науковців – Режим доступу до ресурсу: <https://vkopey.github.io/Python-for-engineers-and-scientists/>
6. Python documentation [Електронний ресурс] – Режим доступу до ресурсу: <http://docs.python.org/3/>

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

Основні форми навчання - лекції, комп'ютерні практикуми та самостійна робота студентів. На лекціях студентам викладаються теоретичні основи дисципліни.

Застосовується стратегія активного і колективного навчання, яка визначається інформаційно-комунікаційною технологією, що забезпечує проблемно-дослідницький характер процесу навчання та активізацію самостійної роботи студентів (електронні презентації для лекційних занять, використання аудіо- та відеопідтримки навчальних занять тощо).

Лекційні заняття:

Розділ 1. Основи програмування та алгоритмізації задач.

Тема 1.1. Види мов програмування. Принципи розробки програм.

Лекція 1. Вступ. Машинні мови, мови асемблера і мови високого рівня. Історія створення мов програмування. Класифікація програмного забезпечення. Класифікація програмістів на ринку праці.

Тема 1.2 Створення алгоритмів.

Лекція 2. Задача, алгоритм, програма, програмна система. Етапи розробки програм: Постановка задачі; аналіз, формалізований опис задачі і вибір моделі; вибір і розробка алгоритму вирішення задачі; проектування загальної структури програми; кодування; супровід програми. Основи алгоритмізації. Алгоритм і його властивості. Способи запису алгоритму. Зображення і функціональний зміст основних символів блок-схем. Приклади побудови алгоритмів.

Тема 1.3. Системи числення, операції із числами, переведення із однієї системи в іншу

Лекція 3. Поняття інформації та одиниці її вимірювання. Системи числення: позиційні і непозиційні системи числення. Правила переведення чисел із однієї системи у іншу.

Розділ 2. Алгоритмічна мова C++.

Тема 2.1. Типи даних і оператори.

Лекція 4. Концепція типу даних. Основні типи даних. Структура програми. Символи. Константи. Строкові константи. Керуючі символьні послідовності. Змінні. Ініціалізація змінної. Оператори. Арифметичні оператори. Інкремент і декремент. Оператори відношення і логічні оператори. Оператор присвоювання. Приведення типів.

Тема 2.2. Базові конструкції структурного програмування.

Лекція 5. Оператори вибору. Інструкція *if*. Вкладені *if*-інструкції. Конструкція *if-else-if*. Інструкція *switch*.

Лекція 6. Оператори циклу. Цикл *for*. Відсутність елементів у визначенні циклу. Цикли без тіла. Оголошення керуючої змінної циклу у заголовці інструкції *for*. Цикл *while*. Оператори переходу. Використання інструкції *break* для виходу з циклу. Використання інструкції *continue*. Інструкція *goto*.

Тема 2.3. Функції.

Лекція 7. Загальний формат C++ - функцій. Оголошення і визначення функцій. Використання аргументів. Використання інструкції *return*. Повернення значень. Використання функцій у виразах. Аргументи, що передаються функції за замовчуванням. Перегрузка функцій і неоднозначність.

Тема 2.4. Масиви і масиви – рядки

Лекція 8. Одномірні масиви. Ініціалізація масивів. Звернення до елементу масиву.

Лекція 9. Двовимірні та багатомірні масиви. Звернення до елементів масиву. Найпростіші дії та пошук даних.

Лекція 10. Масиви-рядки. Функції роботи з масивами-рядками. Поняття змінної строкового типу.

Лекція 11. Алгоритми пошуку. Простий (послідовний) пошук, пошук максимальних та мінімальних значень. Бінарний пошук. Інтерполяційний пошук.

Лекція 12. Сортування масивів. Дурне сортування. Метод бульбашки. Шейкерне сортування. Рекурсивні алгоритми сортування. Швидке сортування. Болотне сортування.

Тема 2.5. Вказівники.

Лекція 13. Вказівники. Оператори, що використовуються з вказівниками. Використання вказівників у виразах. Вказівники і масиви. Посилання. Використання вказівників із масивами і функціями.

Тема 2.6. Типи даних, що визначаються користувачем.

Лекція 14. Структури даних. Призначення та використання структур, створення структури та її елементів, основні принципи роботи із структурами.

Лекція 15. Заключна лекція. Підбиття підсумків, експрес огляд пройдених тем та обговорення основних моментів, пояснення змісту екзаменаційних білетів та умов допуску до екзамену.

Розділ 3. Об'єктно-орієнтоване програмування. Мова Python.

Тема 3.1. Основні відомості про мову програмування Python. Встановлення інтерпретатора Python, додаткових модулів та середовищ розробки IDE.

Лекція 1. Передмова. Мета та задачі курсу. Ознайомлення студентів з рейтинговою системою. Основні відомості про мову програмування Python. Встановлення інтерпретатора Python, додаткових модулів та середовищ розробки IDE.

Тема 3.2. Основні характеристики мови програмування Python. Синтаксис. Типи даних. Змінні. Оператори. Тема 3.3. Алгоритмічні структури Python. Тема 3.4. Функції та модулі.

Лекція 2. Синтаксис мови програмування Python. Типи даних: числа, списки, кортежі, рядки, словники. Змінні. Оператори. Арифметичні та логічні операції. Введення та виведення даних у командний рядок. Алгоритмічні структури Python: лінійні структури, розгалуження та цикли. Функції та модулі.

Тема 3.5. Парадигма об'єктно-орієнтованого програмування. Реалізація класів та екземплярів класів у Python. Інкапсуляція. Спадкування. Поліморфізм. Принципи SOLID.

Лекція 3. Об'єктно-орієнтоване програмування. Загальні терміни та визначення. Пояснення в загальному без прив'язки до мови програмування таких понять як спадкування, поліморфізм, інкапсуляція, абстракція, агрегація, композиція тощо.

Лекція 4. Створення класів та екземплярів класів у Python. Поля (атрибути-дані) та методи класу.

Лекція 5. Спадкування та множинне спадкування. Визначення, приклади застосування. Пояснення роботи алгоритму MRO C3. Метод super().

Лекція 6. Поліморфізм. Визначення, реалізація у Python, приклади застосування.

Лекція 7. Інкапсуляція. Визначення, реалізація у Python, приклади застосування.

Лекція 8. Методологія написання якісного та «чистого коду». Пояснення принципів SOLID. Підбиття підсумків та пояснення змісту екзаменаційних білетів та умов допуску до екзамену

Комп'ютерні практикуми:

ОК ПО 03.1. Програмування. Частина 1. Основи алгоритмізації та структурне програмування

1. Створення найпростішого додатку (2 год)
2. Дії з числами (4 год)
3. Програмування лінійних алгоритмів (4 год)
4. Створення додатку з декількома формами (4 год)
5. Знайомство з інструкцією вибору (6 год)
6. Програмування розгалужених алгоритмів (4 год)
7. Циклічні алгоритми з фіксованою кількістю повторень (4 год)
8. Цикли із невизначеною кількістю ітерацій (6 год)
9. Масиви. Пошук даних в одновимірних масивах (6 год)
10. Пошук даних в багатовимірних масивах (4 год)

Модульна контрольна робота виконується у вигляді двох контрольних робіт, проведення яких збігається із комп'ютерними практикумами №5 та №9.

ОК ПО 03.2. Програмування. Частина 2. Об'єктно-орієнтоване програмування

1. Загальна структура програми мовою Python (4 год)
2. Робота із різними типами даних (6 год)
3. Робота із розгалуженнями та циклами (6 год)
4. Створення функцій користувача та модулів (8 год)
5. Створення класів та екземплярів класів (8 год)
6. Використання механізму спадкування в ООП (4 год)
7. Використання механізму поліморфізму в ООП (4 год)
8. Використання механізму інкапсуляції в ООП (6 год)
9. Написання коду із використанням принципів SOLID (2 год)
10. Створення програм із графічним інтерфейсом користувача (10 год)

Модульна контрольна робота виконується на 29-му за розкладом занятті та обговорюється на 30-му (заключному) занятті.

Індивідуальні завдання:

ОК ПО 03.1. Програмування. Частина 1. Основи алгоритмізації та структурне програмування

В освітньому компоненті ПО 3.1. у якості індивідуального завдання передбачено проведення модульної контрольної роботи яка складається з двох контрольних робіт. Вона проводиться з метою встановлення рівня засвоєння студентами теоретичного матеріалу та визначення їх здатності застосовувати його для вирішення практичних задач.

ОК ПО 03.2. Програмування. Частина 2. Об'єктно-орієнтоване програмування

В освітньому компоненті ПО 3.1. у якості індивідуального завдання передбачено проведення модульної контрольної роботи. Це індивідуальне завдання передбачає розв'язання практичних завдань для закріплення теоретичного матеріалу. Модульна контрольна робота полягає в розв'язанні практичних завдань за темою 3.6 «Розробка програм з графічним інтерфейсом користувача із використанням бібліотеки Tkinter».

6. Самостійна робота студента

У відповідності до робочого навчального плану для освітнього компоненту ПО 3.1 та ПО 3.2 передбачено 76 год та 74 год самостійної роботи студентів відповідно, з яких 30 годин - на підготовку до екзамену, 2 години на підготовку до МКР, а інші - на опрацювання матеріалів лекцій, комп'ютерних практикумів та навчальної літератури відповідно до структури дисципліни.

Робота направлена на засвоєння та поглиблення вивченого матеріалу та на підготовку до занять та семестрового контролю. Самостійна робота студентів передбачає:

- закріплення знань, отриманих під час вивчення дисципліни;
- здобуття навичок самостійного вивчення матеріалу.

Політика та контроль

7. Політика навчальної дисципліни (освітнього компонента)

Перед студентом ставляться наступні вимоги:

- **правила відвідування занять:**

- у режимі очного навчання заняття відбуваються в аудиторії згідно розкладу занять;
- у режимі дистанційного навчання заняття відбуваються у вигляді онлайн-конференції у програмі Zoom - посилання на конференцію видається на початку семестру.
- **правила поведінки на заняттях:**
 - забороняється займатися будь-якою діяльністю, яка прямо не стосується дисципліни;
 - дозволяється використання засобів зв'язку лише для пошуку необхідної для виконання завдань інформації в інтернеті;
 - забороняється будь-яким чином зривати проведення занять.
- **правила захисту комп'ютерних практикумів:**
- **Для ОК ПО 03.1.:**
 - захист роботи проходить під час проведення лабораторної роботи (комп'ютерного практикуму) очно, або на платформі Zoom
 - викладач індивідуально задає запитання на які пропонується відповісти усно, або у вигляді написання коду. Для прискорення захистів, пропонується групування по бригадам по 2-5 студентів.
 - у окремих випадках допускається можливість захисту під час проведення консультацій;
- **Для ОК ПО 03.2.:**
 - виконані самостійні завдання мають бути захищеними (у випадку вчасного виконання завдань і якщо робота не містить ознак плагіату, то захист може й не потребуватись, однак у випадку невчасного виконання завдань без поважної причини захист є обов'язковим);
 - захист самостійних робіт на комп'ютерних практикумах проходить під час проведення комп'ютерного практикуму, а у випадку дистанційного навчання – у режимі онлайн-конференції на платформі Zoom; викладач індивідуально задає запитання, на які пропонується відповісти усно;
 - у окремих випадках допускається можливість захисту під час проведення консультацій.
- **правила призначення заохочувальних та штрафних балів:**
 - штрафні бали призначаються за несвоєчасний захист комп'ютерного практикуму, заохочувальні – за активну роботу на лекціях та ком. практикумах;
 - максимальна кількість заохочувальних та штрафних балів визначається відповідно до Положення про систему оцінювання результатів навчання в КПІ ім. Ігоря Сікорського, Положення про поточний, календарний та семестровий контролі результатів навчання в КПІ ім. Ігоря Сікорського, а також інших Положень та рекомендацій, які діють в КПІ ім. Ігоря Сікорського;
- **політика дедлайнів та перескладань:**
- **Для ОК ПО 03.1.:**
 - перескладання будь-яких контрольних заходів передбачено тільки за наявності документально підтверджених вагомих причин;
 - вчасним захист вважається в межах трьох занять після проведення комп'ютерного практикуму;
 - невчасним вважається захист робіт з затримкою більше ніж на три заняття, порушення даного дедлайну призводить до зменшення кількості балів за роботи та оцінюється на 1 бал нижче, ніж вказано п.8 «Види контролю та рейтингова система оцінювання результатів навчання» за кожні наступні три заняття.

- **Для ОК ПО 03.2.:**

- перескладання будь-яких контрольних заходів передбачено тільки за наявності документально підтверджених вагомих причин відсутності на занятті;
- захист виконаних самостійних робіт на комп'ютерних практикумах вважається вчасним, якщо він відбувається у межах того самого заняття, коли виконується самостійна робота;
- перескладань самостійних робіт для підвищення балів не передбачено.

- **політика округлення рейтингових балів:**

- округлення рейтингового балу відбувається до цілого числа за правилами округлення.

- **політика оцінювання контрольних заходів:**

- оцінювання контрольних заходів відбувається відповідно до Положення про систему оцінювання результатів навчання в КПІ ім. Ігоря Сікорського, Положення про поточний, календарний та семестровий контролю результатів навчання в КПІ ім. Ігоря Сікорського, а також інших Положень та рекомендацій, які діють в КПІ ім. Ігоря Сікорського;
- нижня межа позитивного оцінювання кожного контрольного заходу має бути не менше 60% від балів, визначених для цього контрольного заходу;
- негативний результат оцінюється в 0 балів.

- **політика використання інструментів штучного інтелекту (ШІ)**

- Використання генеративного ШІ (наприклад: ChatGPT, GitHub Copilot, Gemini тощо) дозволено виключно як допоміжного інструменту (для пошуку ідей, оптимізації готового коду, пошуку помилок або редагування тексту звітів). Інструменти ШІ не можуть бути вказані як співавтори результатів навчання.
- Повне копіювання згенерованого ШІ коду, архітектури програмного забезпечення або текстового вмісту звітів без власного аналізу та маркування заборонено.
- У разі виявлення прихованого використання ШІ (без декларування), надання вигаданих ШІ даних/бібліотек або повної нездатності пояснити логіку роботи програми під час захисту, робота не зараховується. Студент отримує 0 балів за дане контрольне завдання з правом повторного виконання та захисту за умови повного його перероблення.

Академічна доброчесність

Політика та принципи академічної доброчесності визначені у розділі 3 Кодексу честі КПІ ім. Ігоря Сікорського. Детальніше: <https://kpi.ua/code>.

Норми етичної поведінки

Норми етичної поведінки студентів і працівників визначені у розділі 2 Кодексу честі КПІ ім. Ігоря Сікорського. Детальніше: <https://kpi.ua/code>.

Оскарження результатів контрольних заходів

Студенти мають право підняти будь-яке питання, яке стосується процедури проведення або оцінювання контрольних заходів та очікувати, що воно буде розглянуто комісією.

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

- **Для ОК ПО 03.1.:**

Поточний контроль:

№	Назва контрольного заходу	Кількість	Ваговий бал	Усього
1	виконання та захист комп'ютерних практикумів (комп'ютерні практикуми №1-10 відповідно до п.5 «Методика опанування навчальної дисципліни (освітнього компонента)» силабусу для ОК ПО 3.1.)	10	5	50

2	Виконання модульної контрольної роботи	1	10	10
			Усього:	60

Критерії оцінювання комп'ютерних практикумів:

5-4 балів – завдання виконано в повному обсязі, здано вчасно та захищене, результат роботи програми правильний, студент дає впевнені та правильні пояснення коду та логіки роботи програми, можливі незначні помилки;

3 бали – завдання виконано не в повному обсязі або присутні невеликі неточності, пояснення в цілому вірні, проте не повністю розкривають зміст питання, або присутні помилки у відповіді;

0 балів – завдання не виконано, пояснення відсутні; або завдання надане викладачу на перевірку, але результат роботи програми неправильний, при цьому студент не може знайти помилку самостійно або відмовляється виправляти код.

Критерії оцінювання МКР:

10 балів – відповідь на всі питання правильна та повна;

9-8 балів – відповіді правильні, присутня незначна кількість помилок або неточностей ;

7-6 балів – більше 60% правильних відповідей, допускається наявність неточностей та помилок;

0 балів – менше 60% правильних відповідей.

Календарний контроль: провадиться двічі на семестр як моніторинг поточного стану виконання вимог силабусу.

Для отримання «зараховано» з першої проміжної атестації (7 тиждень) студент матиме на менш ніж 16 балів (за умови, якщо на початок 7 тижня згідно з календарним планом «ідеальний» студент має отримати 24 бали).

Для отримання «зараховано» з другої проміжної атестації (13 тиждень) студент матиме не менш 30 балів (за умови, якщо на початок 13 тижня згідно з календарним планом «ідеальний» студент має отримати 44 бали).

Семестровий контроль: екзамен

Умови допуску до семестрового контролю: необхідною умовою допуску до екзамену є виконання і захист не менш ніж 7 комп'ютерних практикумів, оцінка з модульної контрольної роботи не менше 6 балів, а також стартовий рейтинг не менший 30 балів.

Екзамен проводиться у змішаній формі. Змішана форма повторює риси усної форми проведення екзамену, із відмінністю у тому, що практичне завдання студенти виконують на комп'ютері і показують результат екзаменатору, тому на початок екзамену в аудиторії знаходиться вся група, що є рисою письмової форми проведення екзамену. Такий підхід зумовлений необхідністю демонстрації набутих навичок на вмінь набутих протягом навчального семестру. Кожний білет містить три теоретичних питання та одне практичне (написання коду). Екзамен проводиться у комп'ютерному класі. Студену пропонується розпочати виконання завдання білету з реалізації практичного завдання, для чого може бути використано персональний комп'ютер з подальшою демонстрацією викладачеві працездатності коду. Відповіді на теоретичні питання даються в усній формі. Кожне теоретичне питання оцінюється у 8 балів, а практичне – 16 балів, тобто сумарний максимальний бал за екзаменаційну контрольну роботу складає Рекз =40 балів.

Система оцінювання теоретичних питань:

- повна відповідь (не менше 90% потрібної інформації) – 7балів;
достатньо повна відповідь (не менше 75% потрібної інформації, або незначні неточності) – 5-6 балів;
- неповна відповідь (не менше 60% потрібної інформації та деякі помилки) – 4 бали;
незадовільна відповідь – 0 балів.

Система оцінювання практичного запитання:

- повне безпомилкове розв'язування завдання (написання коду) та його демонстрація – 14-16 балів;
- повне розв'язування завдання з несуттєвими неточностями – 12-13 балів;
- завдання виконане з певними недоліками, код в цілому правильний, проте присутні помилки, які студент самотійно не здатен виправити – 10-11 балів;
- «незадовільно», завдання не виконано – 0 балів.

• **Для ОК ПО 03.2.:**

Оцінювання результатів навчання здійснюється за 100-бальною шкалою з подальшим переведенням до оцінок за університетською шкалою. РСО складається з двох складових:

- 1) стартової RC – призначена для оцінювання заходів поточного контролю впродовж семестру;
- 2) екзаменаційної RE – призначена для оцінювання окремих запитань (завдань) на екзамені.

Стартова шкала становить RC=50 балів, відповідно, екзаменаційна – RE=50 балів.

Стартовий рейтинг (протягом семестру) складається з балів, що студент отримує за:

- 1) виконання та захист 6 індивідуальних завдань на комп'ютерних практикумах;
- 2) виконання однієї МКР.

Поточний контроль.:

№	Назва контрольного заходу	Кількість	Ваговий бал	Усього
1	виконання та захист комп'ютерних практикумів (комп'ютерні практикуми №1-3 відповідно до п.5 для ОК ПО 3.2.)	3	5	15
2	виконання та захист комп'ютерних практикумів (комп'ютерні практикуми №4-6 відповідно до п.5 для ОК ПО 3.2.)	3	8	24
3	Виконання модульної контрольної роботи	1	11	11
Усього:				50

Важливо! На результат оцінювання виконаного завдання впливатиме не лише його правильність, але й дотримання вимог стандартів з написання коду мовою Python (зокрема, PEP8 «[Style Guide for Python Code](#)»), акуратність оформлення коду, дотримання принципів іменування, дотримання вимог завдання тощо.

Критерії оцінювання КП 1-3:

5 балів – завдання виконано в повному обсязі, здано вчасно та захищене, результат роботи програми правильний, оформлення відповідає стандартам, студент дає впевнені та правильні пояснення коду та логіки роботи програми, можливі незначні помилки;

4 бали – завдання виконано в повному обсязі, результат роботи програми правильний, але погано виконана оптимізація або є незначні помилки в коді, під час захисту пояснення невпевнені, студент плутається в поясненнях, не на всі додаткові запитання дає правильні відповіді;

3 бали – завдання виконано не в повному обсязі (наприклад, виконані не всі розрахунки), пояснення невпевнені або неправильні; або завдання надане викладачу на перевірку, код є робочим, але студент не пояснює і не захищає завдання;

0 балів – завдання не виконано, пояснення відсутні; або завдання надане викладачу на перевірку, але результат роботи програми неправильний, при цьому студент не може знайти помилку самотійно або відмовляється виправляти код.

Критерії оцінювання КП 4-6:

8 балів – завдання виконано в повному обсязі, здано вчасно та захищене, результат роботи програми правильний, оформлення відповідає стандартам, студент дає впевнені та правильні пояснення коду та логіки роботи програми, можливі незначні помилки;

6-7 балів – завдання виконано в повному обсязі, результат роботи програми правильний, але погано виконана оптимізація або є незначні помилки в коді, під час захисту пояснення невпевнені, студент плутається в поясненнях, не на всі додаткові запитання дає правильні відповіді;

5 балів – завдання виконано не в повному обсязі (наприклад, виконані не всі розрахунки), пояснення невпевнені або неправильні; або завдання надане викладачу на перевірку, код є робочим, але студент не пояснює і не захищає завдання;

0 балів – завдання не виконано, пояснення відсутні; або завдання надане викладачу на перевірку, але результат роботи програми неправильний, при цьому студент не може знайти помилку самостійно або відмовляється виправляти код.

Критерії оцінювання МКР:

11 балів – завдання виконано в повному обсязі, здано вчасно та захищене, результат роботи програми правильний, оформлення відповідає стандартам, студент дає впевнені та правильні пояснення коду та логіки роботи програми, можливі незначні помилки;

9-10 балів – завдання виконано в повному обсязі, результат роботи програми правильний, але погано виконана оптимізація або є незначні помилки в коді, під час захисту пояснення невпевнені, студент плутається в поясненнях, не на всі додаткові запитання дає правильні відповіді;

6-8 балів – завдання виконано не в повному обсязі (наприклад, виконані не всі розрахунки), пояснення невпевнені або неправильні; або завдання надане викладачу на перевірку, код є робочим, але студент не пояснює і не захищає завдання;

0 балів – завдання не виконано, пояснення відсутні; або завдання надане викладачу на перевірку, але результат роботи програми неправильний, при цьому студент не може знайти помилку самостійно або відмовляється виправляти код.

Обов'язковою складовою виставлення балів за комп'ютерні практикуми та МКР є їх усний захист. Якщо студент відмовляється захищати завдання, відповідаючи на додаткові запитання викладача та пояснюючи код програми, він не може претендувати на максимальний бал.

Оцінка за комп'ютерні практикуми є остаточною та підвищенню не підлягає, окрім випадків, описаних у наступному абзаці.

Випадок 1. Якщо студент виконав деякі комп'ютерні практикуми на 0 балів, то в кінці семестру він може один (але не більше) із таких практикумів спробувати написати повторно, щоб отримати вищий бал. При цьому він може розраховувати максимум тільки на половину від загальної кількості балів за цей практикум, навіть якщо напише усе правильно.

Випадок 2. Студент наприкінці семестру за бажанням може переписати будь-який один комп'ютерний практикум для підвищення балу (не обов'язково ті, що написані на 0 балів), але якщо напише на меншу оцінку, то вона буде остаточною.

Важливо! Випадок 1 та випадок 2 не можуть бути обрані одночасно – або перший, або другий.

Штрафні та заохочувальні бали:

- за участь у факультетській/університетській/всеукраїнській/закордонній олімпіаді з програмування, розробці власної програми мовою Python або модернізації комп'ютерних практикумів надається від +1 до +3 балів;

- за активну участь у виконанні самостійних домашніх завдань, ведення конспекту тощо надається +2 бали;
- за проходження додаткових дистанційних курсів, що пов'язані з ООП та мовою Python, може бути надано від +1 до +5 балів.

Заохочувальні бали також можуть надаватись за активну участь в обговоренні та розв'язку завдань на лекційних заняттях та комп'ютерних практикумах. Сума заохочувальних балів визначається викладачем (це може бути +0,5 або +1 бал (але не більше) за одне заняття). Сума штрафних балів не може перевищувати «-5», сума заохочувальних балів не може перевищувати «+5».

Календарний контроль:

проводиться двічі на семестр як моніторинг поточного стану виконання вимог силабусу.

Для отримання «зараховано» з першої проміжної атестації (8 тиждень) студент матиме на менш ніж 8 балів.

Для отримання «зараховано» з другої проміжної атестації (14 тиждень) студент матиме не менш 15 балів.

Семестровий контроль: екзамен

Умови допуску до семестрового контролю:

Необхідною умовою допуску до екзамену: є виконання та захист не менш ніж 4 комп'ютерних практикумів, виконання модульної контрольної роботи, стартовий рейтинг $RC \geq 25$ балів (з урахуванням всіх штрафних та заохочувальних балів).

Критерії екзаменаційного оцінювання.

Екзаменаційний білет в базовому варіанті складається з 3 теоретичних запитань та 2 практичних завдань, кожне з яких оцінюється в 10 балів. Максимальна кількість балів, яку можна отримати за правильні відповіді на всі запитання та розв'язання завдань становить 10 балів $\times 5 = 50$ балів.

Критерії оцінювання кожного окремого запитання:

9-10 балів – повна і правильна письмова відповідь з усним поясненням (не менше 90% необхідної інформації); дані правильні визначення термінам; якщо запитання складається з кількох окремих запитань, то повна відповідь дана на усі окремі запитання; якщо в запитанні стоїть вимога привести приклад коду – то такий приклад наведений і є правильним;

7-8 балів – письмова відповідь із деякими неточностями (від 70 до 89% правильної інформації); якщо запитання складається з кількох окремих запитань, а повна відповідь дана не на усі окремі запитання; якщо в запитанні стоїть вимога привести приклад коду – то такий приклад наведений, але є неправильним;

6 балів – відповідь неповна (від 50 до 69% правильної інформації), містить суттєві неточності, визначення дані не всім термінам і поняттям; відповідь стосується теми запитання, але по суті вона є загальною та поверхневою; якщо в запитанні стоїть вимога привести приклад коду – то такий приклад не наведений;

0 балів – відповідь відсутня або невірна; дані неправильні визначення всім термінам і поняттям; відповідь містить тільки матеріал, що не стосується теми запитання; відповідь складається із одного або декількох речень, які не розкривають запитання по суті.

Під час проведення екзамену студент зобов'язаний дотримуватись норм, регламентованих Положенням про організацію навчального процесу в КПІ ім. Ігоря Сікорського, зокрема тих, про які йде мова у статті за цим посиланням <https://kpi.ua/behavior>.

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

Кількість балів	Оцінка
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
Порушення принципів академічної доброчесності або морально-етичних норм поведінки	Усунений

9. Додаткова інформація з дисципліни (освітнього компонента)

У рамках опанування дисципліни «Програмування» допускається можливість зарахування сертифікатів проходження дистанційних чи онлайн курсів за відповідною тематикою (за попереднім узгодженням викладачем).

Робочу програму навчальної дисципліни (силабус):

Складено доцент кафедри КІОНС, к.т.н., доцент Павловський Олексій Михайлович,

доцент кафедри АСНК, к.т.н., доцент Галаган Роман Михайлович

Ухвалено кафедрою КІТВП (протокол № 20 від 24.06.2026 року)

кафедрою КІОНС (протокол № 17 від 24.06.2026 року)

кафедрою АСНК (протокол № 22 від 18.06.2026 року)

Погоджено Методичною комісією факультету робототехніки та приладобудування (протокол № 6/26 від 26.06.2026 року)