



Основи програмування. Частина 2. Методології програмування

Робоча програма дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	<i>12 Інформаційні технології</i>
Спеціальність	<i>121 Програмна інженерія</i>
Освітня програма	<i>Інженерія програмного забезпечення мультимедійних та інформаційно-пошукових систем</i>
Статус дисципліни	<i>Нормативна</i>
Форма навчання	<i>очна(денна)</i>
Рік підготовки, семестр	<i>1 рік навчання, 2 семестр</i>
Обсяг дисципліни	<i>Лекції: 36 год., Комп'ютерний практикум: 18 год., Лабораторні роботи: 36 год., СРС: 75 год</i>
Семестровий контроль/ контрольні заходи	<i>Екзамен, модульна контрольна робота, календарний контроль</i>
Розклад занять	<i>roz.kpi.ua</i>
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	<i>Лектор: к.т.н., ас. Погорелов В.В., volodymyr.pogorelov@gmail.com Практичні: к.т.н., ас. Погорелов В.В., volodymyr.pogorelov@gmail.com Лабораторні: к.т.н., ас. Погорелов В.В., volodymyr.pogorelov@gmail.com</i>
Розміщення курсу	<i>Google classroom: https://classroom.google.com/u/0/c/MzE5ODIyMjA4MsdFTA3?hl=ua</i>

Програма навчальної дисципліни

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

Навчальна дисципліна «Основи програмування. Частина 2. Методології програмування» спрямована на вивчення теоретичних та методологічних основ побудови програм мовою програмування C#, оволодіння засобами створення програмного забезпечення, отримання практичних навичок розробки програмного забезпечення при вирішенні практичних задач. Така теоретична та практична підготовка формує базові навички з програмування і є основою для успішного

опановування фахових дисциплін.

Метою дисципліни є формування у студентів здатності розробляти програмне забезпечення для вирішення прикладних задач різного ступеня складності мовою програмування C#.

Вивчення дисципліни «**Основи програмування. Частина 2. Методології програмування**» формує у здобувачів освіти **загальні (ЗК) та фахові компетентності (ФК):**

ЗК01 Здатність до абстрактного мислення, аналізу та синтезу.

ЗК06 Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

ФК01 Здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення.

ФК02 Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.

ФК03 Здатність розробляти архітектури, модулі та компоненти програмних систем.

ФК07 Володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних.

ФК08 Здатність застосовувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення.

ФК10 Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя.

ФК11 Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розроблення програмного забезпечення.

ФК13 Здатність обґрунтовано обирати та освоювати інструментарій з розроблення та супроводження програмного забезпечення.

ФК14 Здатність до алгоритмічного та логічного мислення.

Програмні результати навчання (ПРН) дисципліни «Основи програмування. Частина 2. Методології програмування» за освітньою програмою:

ПРН01 Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.

ПРН03 Знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення.

ПРН06 Уміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення.

ПРН07 Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.

ПРН13 Знати і застосовувати методи розроблення алгоритмів, конструювання програмного забезпечення та структур даних і знань.

ПРН15 Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення.

ПРН18 Знати та вміти застосовувати інформаційні технології обробки, зберігання та передачі даних.

ПРН38 Вміти застосовувати технології програмування для розроблення програмного забезпечення мультимедійних та інформаційно-пошукових систем.

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

Успішному засвоєнню дисципліни «*Основи програмування. Частина 2. Методології програмування*» передують вивчення дисциплін «*Основи програмування. Частина 1. Базові конструкції*», «*Алгоритми та структури даних*», «*Комп'ютерна дискретна математика*»

Отримані при засвоєнні дисципліни «*Основи програмування. Частина 2. Методології програмування*» теоретичні знання та практичні уміння забезпечують засвоєння дисциплін «*Програмування*», «*Бази даних*», успішне проходження переддипломної практики, виконання курсових та дипломних проєктів за спеціальністю 121 Інженерія програмного забезпечення.

3. Зміст навчальної дисципліни

Розділ 1. Колекції.

Тема 1.1. Введення в колекції.

Тема 1.2. Неузагальнені колекції.

Тема 1.3. Узагальнені колекції.

Тема 1.4. Черга. Стек. Словник. Індиксатори і створення колекцій.

Розділ 2. Робота з потоками і файловою системою.

Тема 2.1. Робота з дисками.

Тема 2.2. Робота з каталогами.

Розділ 3. Серіалізація.

Тема 3.1. Введення в серіалізацію об'єктів.

Тема 3.2. Формати серіалізації.

Розділ 4. Збір сміття, управління пам'яттю і вказівниками.

Тема 4.1. Збирач сміття в C#.

Тема 4.2. Вказівники.

Тема 4.3. Роль збірок в додатках .NET.

Розділ 5. Рефлексія.

Тема 5.1. Введення в рефлексію.

Тема 5.2. Застосування рефлексії і дослідження типів.

Тема 5.3. Атрибути в .NET.

4. Навчальні матеріали та ресурси

Базова література

1. Навчальні матеріали з дисципліни «*Основи програмування*». Використати для опанування теоретичних знань та практичних умінь з дисципліни. Матеріали знаходяться в *Google classroom*. Доступ надається зареєстрованим студентам.

2. Керніган Б., Річі Д. Мова програмування С. Друге видання: пер.В. Цибуляк. 2012. 222 с.
 3. Коваленко І.В. Програмування мовою С# 6.0. Тернопіль, ТНТУ. 2016. 227с.

Додаткова література

1. Настенко Д.В., Нестерко А.Б. Об'єктно-орієнтоване програмування. Частина 1. Основи об'єктно-орієнтованого програмування на мові С#. Київ, НТУУ «КПІ». 2016. 76 с.

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

№ з/п	Тип навчального заняття	Опис навчального заняття
1	Лекція 1.	Введення в колекції. Неузагальнені колекції. Узагальнені колекції. Завдання на СРС: п.6 №1.
2	Лабораторне заняття 1 (частина 1)	Запрограмувати калькулятор з використанням Windows Forms. Завдання на СРС: п.6 №2.
3	Практине заняття 1	Використання колекцій мови С# для вирішення прикладних задач. Завдання на СРС: п.6 №3.
4	Лекція 2.	Черга. Стек. Словник. Завдання на СРС: п.6 №4.
5	Лабораторне заняття 1 (частина 2)	Запрограмувати калькулятор з використанням Windows Forms. Завдання на СРС: п.6 №5.
6	Лекція 3.	Індексатори і створення колекцій. Завдання на СРС: п.6 №6.
7	Лабораторне заняття 1 (частина 3)	Запрограмувати калькулятор з використанням Windows Forms. Завдання на СРС: п.6 №7.
8	Практине заняття 2	Узагальнені та неузагальнені колекції. Завдання на СРС: п.6 №8.
9	Лекція 4.	Основи роботи з файловою системою та потоками. Завдання на СРС: п.6 №9.
10	Лабораторне заняття 2 (частина 1)	Запрограмувати власний текстовий редактор з використанням Windows Forms. Завдання на СРС: п.6 №10.
11	Лекція 5.	Основні вбудовані методи для роботи з дисками та каталогами. Завдання на СРС: п.6 №11.
12	Лабораторне заняття 2 (частина 2)	Запрограмувати власний текстовий редактор з використанням Windows Forms. Завдання на СРС: п.6 №12.

13	Практине заняття 3	Створення колекцій та робота з індексерами. Завдання на СРС: п.6 №13.
14	Лекція 6.	Поняття серіалізації об'єктів та застосування у прикладних задачах. Завдання на СРС: п.6 №14.
15	Лабораторне заняття 2 (частина 3)	Запрограмувати власний текстовий редактор з використанням Windows Forms. Завдання на СРС: п.6 №15.
16	Лекція 7.	Формати серіалізації та десеріалізації об'єктів. Завдання на СРС: п.6 №16.
17	Лабораторне заняття 3	Реалізувати власний графічний редактор з використанням Windows Forms. Завдання на СРС: п.6 №17.
18	Практине заняття 4	Робота з потоками, файлами та дисками Завдання на СРС: п.6 №18.
19	Лекція 8.	Способи організації роботи з пам'яттю в різних мовах програмування Завдання на СРС: п.6 №19.
20	Лабораторне заняття 4 (частина 1)	Модифікувати код іншої лабораторної роботи (однієї з попередніх або з іншої дисципліни) для збереження даних програми між запусками за допомогою серіалізації класів. Завдання на СРС: п.6 №20.
21	Лекція 9.	Збирач сміття в C# Завдання на СРС: п.6 №21.
22	Лабораторне заняття 4 (частина 2)	Модифікувати код іншої лабораторної роботи (однієї з попередніх або з іншої дисципліни) для збереження даних програми між запусками за допомогою серіалізації класів. Завдання на СРС: п.6 №22.
23	Практине заняття 5	Серіалізація об'єктів з використанням файлів різних форматів Завдання на СРС: п.6 №23.
24	Лекція 10.	Робота з вказівниками. Завдання на СРС: п.6 №24.
25	Лабораторне заняття 4 (частина 3)	Модифікувати код іншої лабораторної роботи (однієї з попередніх або з іншої дисципліни) для збереження даних програми між запусками за допомогою серіалізації класів. Завдання на СРС: п.6 №25.
26	Лекція 11.	Особливості використання збірок в додатках .NET. Завдання на СРС: п.6 №26.

27	Лабораторне заняття 4 (частина 4)	Модифікувати код іншої лабораторної роботи (однієї з попередніх або з іншої дисципліни) для збереження даних програми між запусками за допомогою серіалізації класів. Завдання на СРС: п.6 №27.
28	Практине заняття 6	Десериалізація об'єктів та особливості обробки помилок Завдання на СРС: п.6 №28.
29	Лекція 12.	Поняття рефлексії. Ревлексія в мові C# Завдання на СРС: п.6 №29.
30	Лабораторне заняття 5 (частина 1)	Додати до проекту іншої лабораторної роботи (однієї з попередніх або з іншої дисципліни) unit тести. Завдання на СРС: п.6 №30.
31	Лекція 13.	Рефлексія як основний інструмент для реалізації сучасних фреймворків. Завдання на СРС: п.6 №31.
32	Лабораторне заняття 5 (частина 2)	Додати до проекту іншої лабораторної роботи (однієї з попередніх або з іншої дисципліни) unit тести. Завдання на СРС: п.6 №32.
33	Практине заняття 7	Робота з різними складними типами даних із застосуванням рефлексії Завдання на СРС: п.6 №33.
34	Лекція 14.	Застосування рефлексії і дослідження типів. Завдання на СРС: п.6 №34.
35	Лабораторне заняття 5 (частина 3)	Додати до проекту іншої лабораторної роботи (однієї з попередніх або з іншої дисципліни) unit тести. Завдання на СРС: п.6 №35.
36	Лекція 15.	Атрибути в .NET Завдання на СРС: п.6 №36.
37	Лабораторне заняття 5 (частина 4)	Додати до проекту іншої лабораторної роботи (однієї з попередніх або з іншої дисципліни) unit тести. Завдання на СРС: п.6 №37.
38	Практине заняття 8	Використання фреймворків що базуються на рефлексії для розв'язку прикладних задач Завдання на СРС: п.6 №38.
39	Лекція 16.	Реалізація патерну інверсії контролю з використанням рефлексії. Завдання на СРС: п.6 №39.
40	Лабораторне заняття 6 (частина 1)	Функції. Створення програмного забезпечення з власнорозробленими функціями Завдання на СРС: п.6 №40.

41	Лекція 17.	Приклади застосування рефлексії для вирішення прикладних задач. Завдання на СРС: п.6 №41.
42	Лабораторне заняття 6 (частина 2)	Функції. Створення програмного забезпечення з власнорозробленими функціями Завдання на СРС: п.6 №42.
Модульна контрольна робота		

6. Самостійна робота студента

Дисципліна «Основи програмування. Частина 2. Методології програмування» ґрунтується на самостійних підготовках до аудиторних занять на теоретичні та практичні теми.

№ з/п	Назва теми, що виноситься на самостійне опрацювання	Кількість годин	Література
1	Підготовка до лекції 1	0.5	1-2
2	Підготовка лабораторного заняття 1 (частина 1)	1.5	1-2
3	Підготовка до практичного заняття 1	1	1-2
4	Підготовка до лекції 2	0.5	1-2
5	Підготовка лабораторного заняття 1 (частина 2)	1.5	1-2
6	Підготовка до лекції 3	0.5	1-2
7	Підготовка лабораторного заняття 1 (частина 3)	1.5	1-2
8	Підготовка до практичного заняття 2	1	1-2
9	Підготовка до лекції 4	0.5	1-2
10	Підготовка лабораторного заняття 2 (частина 1)	1.5	1-2
11	Підготовка до лекції 5	0.5	1-2
12	Підготовка лабораторного заняття 2 (частина 2)	1.5	1-2
13	Підготовка до практичного заняття 3	1	1-2
14	Підготовка до лекції 6	0.5	1-2
15	Підготовка лабораторного заняття 2 (частина 3)	1.5	1-2
16	Підготовка до лекції 7	0.5	1-2
17	Підготовка лабораторного заняття 3	1.5	1-2
18	Підготовка до практичного заняття 4	1	1-2
19	Підготовка до лекції 8	0.5	1-2
20	Підготовка лабораторного заняття 4 (частина 1)	1.5	1-2
21	Підготовка до лекції 9	0.5	1-2
22	Підготовка лабораторного заняття 4 (частина 2)	1.5	1-2

23	Підготовка до практичного заняття 5	1	1-2
24	Підготовка до лекції 10	0.5	1-2
25	Підготовка лабораторного заняття 4 (частина 3)	1.5	1-2
26	Підготовка до лекції 11	0.5	1-2
27	Підготовка лабораторного заняття 4 (частина 4)	1.5	1-2
28	Підготовка до практичного заняття 6	1	1-2
29	Підготовка до лекції 12	0.5	1-2
30	Підготовка лабораторного заняття 5 (частина 1)	1.5	
31	Підготовка до лекції 13	0.5	
32	Підготовка лабораторного заняття 5 (частина 2)	1.5	
33	Підготовка до практичного заняття 7	1	
34	Підготовка до лекції 14	0.5	
35	Підготовка лабораторного заняття 5 (частина 3)	1.5	
36	Підготовка до лекції 15	0.5	
37	Підготовка лабораторного заняття 5 (частина 4)	1.5	
38	Підготовка до практичного заняття 8	1	
39	Підготовка до лекції 16	0.5	
40	Підготовка лабораторного заняття 6 (частина 1)	1.5	
41	Підготовка до лекції 17	0.5	
42	Підготовка лабораторного заняття 6 (частина 2)	1.5	
43	Підготовка до МКР	4.5	
44	Підготовка до екзамену	30	

Політика та контроль

7. Політика навчальної дисципліни (освітнього компонента)

Відвідування занять. Відсутність на аудиторному занятті не передбачає нарахування штрафних балів, оскільки фінальний рейтинговий бал студента формується виключно на основі оцінювання результатів навчання. Разом з тим, обговорення результатів виконання тематичних завдань, а також презентація / публічний виступ та участь у обговореннях та доповнення на семінарах оцінюватимуться під час аудиторних занять. Для активної участі у роботі семінару студент готується за рекомендованою викладачем до певного семінарського заняття літературою. Участь у роботі семінару також передбачає підготування доповідей та співдоповідей у межах усіх занять.

Пропущені контрольні заходи оцінювання. Кожен студент має право відпрацювати пропущені з поважної причини (лікарняний, мобільність тощо) заняття за рахунок самостійної роботи. Детальніше за посиланням: <https://kpi.ua/files/n3277.pdf>.

Процедура оскарження результатів контрольних заходів оцінювання. Студент може підняти будь-яке питання, яке стосується процедури контрольних заходів та очікувати, що воно буде розглянуто згідно із наперед визначеними процедурами. Студенти мають право аргументовано оскаржити результати контрольних заходів, пояснивши з яким критерієм не погоджуються відповідно до оціночного. Календарний контроль проводиться з метою підвищення якості навчання студентів та моніторингу виконання студентом вимог силабусу.

Академічна доброчесність. 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>.

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>.

Інклюзивне навчання. Засвоєння знань та умінь в ході вивчення дисципліни «Науково-дослідна діяльність у комп'ютерній інженерії» може бути доступним для більшості осіб з особливими освітніми потребами, окрім здобувачів з серйозними вадами зору, які не дозволяють виконувати завдання за допомогою персональних комп'ютерів, ноутбуків та/або інших технічних засобів.

Навчання іноземною мовою. У ході виконання завдань студентам може бути рекомендовано звернутися до англомовних джерел. Призначення заохочувальних та штрафних балів Відповідно до Положення про систему оцінювання результатів навчання сума всіх заохочувальних балів не може перевищувати 10% рейтингової шкали оцінювання.

Всі студенти повинні відвідувати лекційні та практичні заняття, на яких потрібно активно працювати над засвоєнням навчального матеріалу. За об'єктивних причин (наприклад - хвороба, міжнародне стажування) навчання може відбуватись в он-лайн формі індивідуально за погодженням із керівником курсу.

Політика щодо дедлайнів та перескладання:

Роботи, які здаються із порушенням термінів без поважних причин, оцінюються на нижчу оцінку. Перескладання модулів відбувається із дозволу деканату за наявності поважних причин (наприклад, лікарняний).

Політика щодо академічної доброчесності:

Усі письмові роботи перевіряються на наявність плагіату і допускаються до захисту із коректними текстовими запозиченнями не більше 20%. Списування під час контрольних робіт заборонені (в т. ч. із використанням мобільних пристроїв).

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

8.1 Поточний контроль

Поточний контроль результатів навчання передбачає виконання студентами лабораторних робіт. Загальна кількість **лабораторних робіт** – 6. За кожну лабораторну роботу максимальна кількість балів -5 балів.

Бали нараховуються за:

- якість виконання лабораторної роботи (комп'ютерного практикуму): 0–3 бали;
- відповідь під час захисту лабораторної роботи (комп'ютерного практикуму): 0–2 бали;

Критерії оцінювання якості виконання роботи:

3 бали – робота виконана якісно, в повному обсязі;
1–2 бали – робота виконана якісно, в повному обсязі, але має недоліки;
0–0.5 бали – робота виконана не в повному обсязі, або містить суттєві помилки або відсутня.

Критерії оцінювання відповіді:

2 бали – відповідь повна, добре аргументована;
1–1,5 бали – в цілому відповідь вірна, але має недоліки або незначні помилки;
0–0,5 бали – у відповіді є суттєві помилки або вона відсутня;

Модульна контрольна робота проводиться у вигляді написання програмного забезпечення для вирішення завдання згідно з варіантом. Максимальна кількість балів за МКР - 20 балів.

17-20 балів – відповідь вірна, програмні розрахунки виконані у повному обсязі;
13-16 балів – відповідь вірна, але програмні розрахунки не завжди дають вірний результат;
9-12 балів – у відповіді є незначні помилки, які суттєво впливають на результати програмних розрахунків;
0-8 бали – у відповіді є суттєві помилки, немає відповіді або відповідь невірна.

Семестрова складова рейтингової шкали $R_C = 50$ балів.

Рейтингова шкала з дисципліни дорівнює: $R = R_{л.р.} + R_{мкр} = 30$ балів + 20 балів = 50 балів.

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

На першій атестації (8-й тиждень) студент отримує «зараховано», якщо його поточний рейтинг не менше 50 % від максимальної кількості балів, яку може отримати студент до першої атестації.

На другій атестації (14-й тиждень) студент отримує «зараховано», якщо його поточний рейтинг не менше 50 % від максимальної кількості балів, яку може отримати студент до другої атестації.

При семестровому рейтингу (R_C) не менше 30 балів та зарахуванні усіх практичних та лабораторних робіт студент отримує допуск до екзамену. **Екзаменаційна робота** складається з написання і налагодження програми відповідно до поставленої задачі та відповіді на теоретичне питання.

Максимальна кількість балів за практичне завдання – 30 балів.

25-30 балів – відповідь вірна, програмні розрахунки виконані у повному обсязі;
19-24 балів – відповідь вірна, але програмні розрахунки не завжди дають вірний результат;
13-18 балів – у відповіді є незначні помилки, які суттєво впливають на результати програмних розрахунків;
0-12 бали – у відповіді є суттєві помилки, немає відповіді або відповідь невірна.

Максимальна кількість балів за теоретичне питання – 20 балів.

Критерії оцінювання кожного теоретичного запитання роботи:

18-20 бали – відповідь вірна, повна, добре аргументована;
15-17 бали – відповідь вірна, розгорнута, але не дуже добре аргументована;
10-14 бали – відповідь не повна та неаргументована та має суттєві помилки, які не дають можливість отримати вірний результат до задачі / завдання;
0-9 бали – відповідь не вірна, відповіді немає.

Критерії оцінювання кожного теоретичного запитання контрольної роботи:

10 бали – відповідь вірна, повна, добре аргументована;

7-9 бали – відповідь вірна, розгорнута, але не дуже добре аргументована;

3-6 бали – відповідь не повна та неаргументована та має суттєві помилки, які не дають можливість отримати вірний результат до задачі / завдання;

0-2 бали – відповідь не вірна, відповіді немає.

Підсумкова оцінка формується за результатами оцінювання знань та навичок студента в семестрі та на екзамені за формулою: $R = RC + RE$. Підсумкова оцінка переводиться до залікової оцінки згідно з таблицею (Таблиця відповідності рейтингових балів оцінкам за університетською шкалою).

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

<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

Робочу програму навчальної дисципліни (силабус):

Складено : к.т.н. ас. Погорелов В.В.

Ухвалено кафедрою ПЗКС (протокол № 12 від 26.04.2023 р.)

Погоджено Методичною комісією факультету прикладної математики (протокол № 10 від 26.05.2023 р.)