

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

Затверджено

На засіданні кафедри програмування
факультету прикладної математики
Львівського національного університету
імені Івана Франка
(протокол № 1 від 31 серпня 2022 р.)



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

Силабус навчальної дисципліни
«Програмування» (IV семестр),
що викладається в межах ОПП Комп'ютерні науки
першого (бакалаврського) рівня вищої освіти
для здобувачів зі спеціальності 122 Комп'ютерні науки (Інформатика)

Львів 2022 р.

Назва дисципліни	Програмування
Адреса викладання дисципліни	Львівський національний університет імені Івана Франка, вул. Університетська 1, м. Львів, Україна, 79000
Факультет та кафедра, за якою закріплена дисципліна	Факультет прикладної математики та інформатики, кафедра програмування
Галузь знань, шифр та назва спеціальності	Галузь знань: 12 Інформаційні технології Спеціальність: 122 Комп'ютерні науки Спеціалізація: Інформатика
Викладачі дисципліни	Ярошко Сергій Адамович, к. ф.-м. н., доцент, завідувач кафедри програмування; Муzychuk Анатолій Омелянович, к. ф.-м. н., доцент, доцент кафедри програмування
Контактна інформація викладачів	Електронна пошта: serhiy.yaroshko@lnu.edu.ua , anatoliy.muzychuk@lnu.edu.ua веб-сторінки: https://ami.lnu.edu.ua/employee/yaroshko https://ami.lnu.edu.ua/employee/muzychuk
Консультації з питань навчання по дисципліні відбуваються	Консультації проводять раз на тиждень згідно з оприлюдненим розкладом консультацій викладача. Можливі он-лайн консультації через Microsoft Teams. Для погодження часу он-лайн консультацій слід писати на електронну пошту викладача.
Сторінка курсу	https://ami.lnu.edu.ua/course/basics-of-programming-informatics
Інформація про дисципліну	Курс "Програмування" (мовою C#) є нормативною дисципліною зі спеціальності 122 Комп'ютерні науки (інформатика) для освітньої програми Комп'ютерні науки, яку викладають у четвертому семестрі в обсязі 4 кредити (за Європейською кредитно-трансферною системою ECTS)
Коротка анотація дисципліни	Фокус уваги курсу спрямовано на оволодіння базовими інструментами середовища .Net та особливості мови програмування C#: властивості, події, індексатори, методи розширення, використання інтерфейсів, серіалізація, життєвий цикл об'єктів, засоби рефлексії, LINQ, багатопотокове програмування. Розглянуто засоби Windows Forms для побудови застосунків.
Мета та цілі дисципліни	Метою нормативної дисципліни «Програмування» є ознайомити студента з базовими можливостями платформи .Net, навчити його використовувати мову C# для створення ефективних застосунків, приготувати до вивчення курсу "Програмна інженерія".
Література для вивчення дисципліни	<p><i>Основна література</i></p> <ol style="list-style-type: none"> 1. Learn C#. OFFICIAL COLLECTION: https://learn.microsoft.com/en-us/users/dotnet/collections/yz26f8y64n7k07?WT.mc_id=dotnet-35129-website 2. C# programming guide https://learn.microsoft.com/en-us/dotnet/csharp/programmingguide/?WT.mc_id=dotnet-35129-website 3. Object-oriented programming (C#) https://learn.microsoft.com/en-us/dotnet/csharp/fundamentals/tutorials/oop?source=recommendations 4. Inheritance in C# and .NET https://learn.microsoft.com/en-us/dotnet/csharp/fundamentals/tutorials/inheritance 5. Language Integrated Query (LINQ) https://learn.microsoft.com/en-us/dotnet/csharp/programmingguide/concepts/linq 6. Troelsen Andrew, Japikse Phil. Pro C# 10 with .NET 6: Foundational Principles and Practices in Programming – APress, 2022 – 1640 pp. 7. Christian Nagel Professional C# and .NET, 2021st Edition – Wrox, 2021 – 1800 pp. <p><i>Додаткова література</i></p> <ol style="list-style-type: none"> 8. Mark J. Price C# 10 and .NET 6 – Modern Cross-Platform Development – Packt Publishing, 2021 – 826 pp. 9. Joseph Albahari C# 10 in a Nutshell: The Definitive Reference – O'Reilly Media, 2022 – 1000 pp.

Обсяг курсу	4 кредити ЄКТС – 120 годин. З них 32 години лекцій, 32 години лабораторних занять та 56 годин самостійної роботи
Очікувані результати навчання	<p>Після завершення цього курсу студент буде:</p> <p><i>знати</i></p> <ul style="list-style-type: none"> • засади функціонування платформи .Net; • синтаксис мови програмування C#, прийоми об'єктно-орієнтованого програмування мовою C#; • можливості та правила використання стандартних класів, зокрема, контейнерів, стандартних інтерфейсів; • призначення та влаштування компонент бібліотеки Windows Forms. <p><i>вміти</i></p> <ul style="list-style-type: none"> • оголошувати та використовувати у програмах мовою C# ієрархії класів, реалізовувати стандартні та власні інтерфейси; • налагоджувати взаємодію об'єктів через події; • серіалізувати об'єкти, програмно взаємодіяти з файловою системою; • використовувати LINQ для опрацювання колекцій даних; • будувати застосунки за допомогою Windows Forms; • створювати багатопотокові застосунки.
Компетентності	<p><i>Інтегральна:</i> Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми у галузі комп'ютерних наук або у процесі навчання, що передбачають застосування теорій та методів інформаційних технологій і характеризується комплексністю та невизначеністю умов.</p> <p><i>Загальні (ЗК):</i></p> <p>ЗК1. Здатність до абстрактного мислення, аналізу та синтезу.</p> <p>ЗК2. Здатність застосовувати знання у практичних ситуаціях.</p> <p>ЗК3. Знання та розуміння предметної області та розуміння професійної діяльності.</p> <p>ЗК7. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.</p> <p>ЗК8. Здатність генерувати нові ідеї (креативність).</p> <p>ЗК9. Здатність працювати в команді.</p> <p>ЗК12. Здатність оцінювати та забезпечувати якість виконуваних робіт.</p> <p>СК2,3,4,6,7,8,10,11,16</p> <p><i>Спеціальні (фахові, предметні) компетентності (СК):</i></p> <p>СК3. Здатність до логічного мислення, побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проектування, розроблення й аналізу алгоритмів, оцінювання їх ефективності та складності, розв'язності та нерозв'язності алгоритмічних проблем для адекватного моделювання предметних областей і створення програмних та інформаційних систем.</p> <p>СК7. Здатність застосовувати теоретичні та практичні основи методології та технології моделювання для дослідження характеристик і поведінки складних об'єктів і систем, проводити обчислювальні експерименти з обробкою й аналізом результатів.</p> <p>СК8. Здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління.</p> <p>СК10. Здатність застосовувати методології, технології та інструментальні засоби для управління процесами життєвого циклу інформаційних і програмних систем, продуктів і сервісів інформаційних технологій відповідно до вимог замовника.</p>
Програмні результати навчання	<p>ПР1. Застосовувати знання основних форм і законів абстрактно-логічного мислення, основ методології наукового пізнання, форм і методів вилучення, аналізу, обробки та синтезу інформації в предметній області комп'ютерних наук.</p>

	<p>ПР5. Проектувати, розробляти та аналізувати алгоритми розв’язання обчислювальних та логічних задач, оцінювати ефективність та складність алгоритмів на основі застосування формальних моделей алгоритмів та обчислюваних функцій.</p> <p>ПР6. Використовувати методи чисельного диференціювання та інтегрування функцій, розв'язання звичайних диференціальних та інтегральних рівнянь, особливостей чисельних методів та можливостей їх адаптації до інженерних задач, мати навички програмної реалізації чисельних методів.</p> <p>ПР9. Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв’язання задач в галузі комп’ютерних наук.</p> <p>ПР11. Володіти навичками управління життєвим циклом програмного забезпечення, продуктів і сервісів інформаційних технологій відповідно до вимог і обмежень замовника, вміти розробляти проектну документацію (техніко-економічне обґрунтування, технічне завдання, бізнес-план, угоду, договір, контракт).</p> <p>ПР15. Застосовувати знання методології та CASE-засобів проектування складних систем, методів структурного аналізу систем, об’єктно-орієнтованої методології проектування при розробці і дослідженні функціональних моделей організаційно-економічних і виробничо-технічних систем.</p>																																														
Ключові слова	Платформа .Net, мова програмування C#, клас, інтерфейс, властивість, подія, делегат, LINQ, серіалізація, багатопотоковість, Windows Forms.																																														
Формат курсу	Очний: проведення лекцій, лабораторних робіт та консультацій в приміщеннях університету, а в умовах карантину – онлайн-овий на платформі Microsoft Teams																																														
Теми	<table><tr><th>Тиждень</th><th>Тема, план, короткі тези</th><th>Форма заняття</th><th>Тривалість, год</th><th>Термін виконання</th></tr><tr><td rowspan="2">1</td><td>Архітектура, призначення, можливості платформи .Net. Система типів, операції мови C#. Типи-значення та типи-посилання C#. Консольне введення-виведення. Конструкції керування мови C#. Методи System. Object. Перелік - найпростіший тип користувача. Масиви C#: одно- та багатовимірні, зубчасті.</td><td>Лекція</td><td>2</td><td></td></tr><tr><td>Побудова консольних програм в середовищі Visual Studio засобами мови C#. Форматоване виведення. Послідовні, галужені, циклічні алгоритми з використанням вбудованих типів.</td><td>Лабораторна робота</td><td>2</td><td>Наступне лабораторне заняття</td></tr><tr><td rowspan="2">2</td><td>Оголошення класів і структур мовою C#: поля, методи, властивості. Передавання аргументів методам. Конструювання, копіювання об’єктів. Статичні члени класу, статичні класи. Методи, що розширюють клас.</td><td>Лекція</td><td>2</td><td></td></tr><tr><td>Опрацювання масивів, у тому числі зубчастих. Оголошення та використання переліків.</td><td>Контрольна робота</td><td>2</td><td></td></tr><tr><td rowspan="2">3</td><td>Операції мови C#: is, as, поглинання null та інші спеціальні. Перетворення типів. Перевірка рівності об’єктів. Перевантаження операцій. Методи-індексатори.</td><td>Лекція</td><td>2</td><td></td></tr><tr><td>Оголошення та використання структур. Оголошення та використання простих класів. Побудова та використання властивостей.</td><td>Лабораторна робота</td><td>2</td><td>Наступне лабораторне заняття</td></tr><tr><td rowspan="2">4</td><td>Інтерфейси: означення, можливості використання. Стандартні інтерфейси IComparable, IEnumerable, ICloneable. Наслідування класів та інтерфейсів. Абстрактні класи. Віртуальні та перекриті методи. Конструктори підкласів. Закриті класи і методи. Часткові класи.</td><td>Лекція</td><td>2</td><td></td></tr><tr><td>Перевантаження операцій. Використання власних індексаторів. Порівняння об’єктів. Перетворення типів.</td><td>Лабораторна робота</td><td>2</td><td>Наступне лабораторне заняття</td></tr><tr><td>5</td><td>Делегати, події в мові C#: визначення типу делегата, використання простого та групового делегатів, масиву делегатів, анонімні методи і лямбда-вирази. Визначення події, взаємодія об’єктів через подію.</td><td>Лекція</td><td>2</td><td></td></tr></table>	Тиждень	Тема, план, короткі тези	Форма заняття	Тривалість, год	Термін виконання	1	Архітектура, призначення, можливості платформи .Net. Система типів, операції мови C#. Типи-значення та типи-посилання C#. Консольне введення-виведення. Конструкції керування мови C#. Методи System. Object. Перелік - найпростіший тип користувача. Масиви C#: одно- та багатовимірні, зубчасті.	Лекція	2		Побудова консольних програм в середовищі Visual Studio засобами мови C#. Форматоване виведення. Послідовні, галужені, циклічні алгоритми з використанням вбудованих типів.	Лабораторна робота	2	Наступне лабораторне заняття	2	Оголошення класів і структур мовою C#: поля, методи, властивості. Передавання аргументів методам. Конструювання, копіювання об’єктів. Статичні члени класу, статичні класи. Методи, що розширюють клас.	Лекція	2		Опрацювання масивів, у тому числі зубчастих. Оголошення та використання переліків.	Контрольна робота	2		3	Операції мови C#: is, as, поглинання null та інші спеціальні. Перетворення типів. Перевірка рівності об’єктів. Перевантаження операцій. Методи-індексатори.	Лекція	2		Оголошення та використання структур. Оголошення та використання простих класів. Побудова та використання властивостей.	Лабораторна робота	2	Наступне лабораторне заняття	4	Інтерфейси: означення, можливості використання. Стандартні інтерфейси IComparable, IEnumerable, ICloneable. Наслідування класів та інтерфейсів. Абстрактні класи. Віртуальні та перекриті методи. Конструктори підкласів. Закриті класи і методи. Часткові класи.	Лекція	2		Перевантаження операцій. Використання власних індексаторів. Порівняння об’єктів. Перетворення типів.	Лабораторна робота	2	Наступне лабораторне заняття	5	Делегати, події в мові C#: визначення типу делегата, використання простого та групового делегатів, масиву делегатів, анонімні методи і лямбда-вирази. Визначення події, взаємодія об’єктів через подію.	Лекція	2	
Тиждень	Тема, план, короткі тези	Форма заняття	Тривалість, год	Термін виконання																																											
1	Архітектура, призначення, можливості платформи .Net. Система типів, операції мови C#. Типи-значення та типи-посилання C#. Консольне введення-виведення. Конструкції керування мови C#. Методи System. Object. Перелік - найпростіший тип користувача. Масиви C#: одно- та багатовимірні, зубчасті.	Лекція	2																																												
	Побудова консольних програм в середовищі Visual Studio засобами мови C#. Форматоване виведення. Послідовні, галужені, циклічні алгоритми з використанням вбудованих типів.	Лабораторна робота	2	Наступне лабораторне заняття																																											
2	Оголошення класів і структур мовою C#: поля, методи, властивості. Передавання аргументів методам. Конструювання, копіювання об’єктів. Статичні члени класу, статичні класи. Методи, що розширюють клас.	Лекція	2																																												
	Опрацювання масивів, у тому числі зубчастих. Оголошення та використання переліків.	Контрольна робота	2																																												
3	Операції мови C#: is, as, поглинання null та інші спеціальні. Перетворення типів. Перевірка рівності об’єктів. Перевантаження операцій. Методи-індексатори.	Лекція	2																																												
	Оголошення та використання структур. Оголошення та використання простих класів. Побудова та використання властивостей.	Лабораторна робота	2	Наступне лабораторне заняття																																											
4	Інтерфейси: означення, можливості використання. Стандартні інтерфейси IComparable, IEnumerable, ICloneable. Наслідування класів та інтерфейсів. Абстрактні класи. Віртуальні та перекриті методи. Конструктори підкласів. Закриті класи і методи. Часткові класи.	Лекція	2																																												
	Перевантаження операцій. Використання власних індексаторів. Порівняння об’єктів. Перетворення типів.	Лабораторна робота	2	Наступне лабораторне заняття																																											
5	Делегати, події в мові C#: визначення типу делегата, використання простого та групового делегатів, масиву делегатів, анонімні методи і лямбда-вирази. Визначення події, взаємодія об’єктів через подію.	Лекція	2																																												

		Оголошення та використання інтерфейсів. Оголошення класів, що реалізують стандартні інтерфейси.	Лабораторн а робота	2	Наступне лабораторн е заняття
	6	Розробка узагальнених типів. Узагальнення класів. Узагальнення інтерфейсів. Обмеження узагальнень. Засоби узагальнених класів. Питання обсягу коду та продуктивності.	Лекція	2	
		Побудова ієрархії класів. Оголошення і використання конструкторів, фіналізаторів, віртуальних методів.	Контрольна робота	2	
	7	Колекції .Net, узагальнені колекції. Списки (зв'язні, сортовані), стек, черга, словник, бітовий масив. Перехоплення та опрацювання винятків. Класи винятків користувача, запуск винятків.	Лекція	2	
		Синхронний та асинхронний виклики методів через делегата. Визначення події, організація взаємодії об'єктів через подію.	Лабораторн а робота	2	Наступне лабораторн е заняття
	8	Простір System.Reflection. Програмний доступ до метаданих. Дослідження збірки, типу. Атрибути користувача та їх використання. Приклади. Доступ до приватних членів класу. Динамічний виклик методу.	Лекція	2	
		Розробка узагальнених класів. Створення класів винятків. Запуск, перехоплення та опрацювання винятків.	Лабораторн а робота	2	Наступне лабораторн е заняття
	9	Файлове введення-виведення і серіалізація об'єктів. Доступ до файлової системи. Маніпулювання файлами і директоріями. Потoki даних. Формати серіалізації: двійковий, XML, SOAP.	Лекція	2	
		Використання інструментів рефлексії для програмного дослідження класів. Побудова та використання атрибутів користувача.	Лабораторн а робота	2	Наступне лабораторн е заняття
	10	Побудова Windows-аплікацій з використанням Windows Forms. Стандартні елементи керування. Діалоги. Приклади побудови застосунків: перша програма, модель світлофора.	Лекція	2	
		Використання колекцій .Net. Робота з рядками, текстом.	Лабораторн а робота	2	Наступне лабораторн е заняття
	11	Приклади побудови застосунків: тестування, редактор текстів, багатовіконні програми. Меню, діалоги, панелі інструментів, рядок стану.	Лекція	2	
		Програмний пошук папок, файлів. Зберігання до файла (у різних форматах) та завантаження об'єктів з файла.	Лабораторн а робота	2	Наступне лабораторн е заняття
	12	Створення складних Windows Forms User Control, забезпечення підтримки етапу проектування аплікації	Лекція	2	
		Побудова Windows-аплікацій зі стандартними елементами керування Windows Forms. Файлові діалоги.	Лабораторн а робота	2	Наступне лабораторн е заняття
	13	Багатопотоковість і синхронізація. Асинхронні делегати. Класи Thread, Task. Синхронізація. Багатопотокові аплікації	Лекція	2	
		Побудова багатопотокових Windows-аплікацій: окремий потік для взаємодії з GUI, окремий для обчислень тощо.	Контрольна робота	2	
	14	Засоби мови, що підтримують LINQ. LINQ to objects. Блок ітератора. Приклади.	Лекція	2	
		LINQ: розширення типів, лямбда вирази, LINQ-запити	Лабораторн а робота	2	Наступне лабораторн е заняття
	15	Життєвий цикл об'єктів. Finalize. IDisposable. Менеджер керованої купи, збирання сміття.	Лекція	2	
		Взаємодія з менеджером купи. Реалізація фіналізаторів. Реалізація інтерфейсу IDisposable.	Лабораторн а робота	2	Наступне лабораторн е заняття
	16	Побудова та опрацювання рядків, використання String і StringBuilder.	Лекція	2	
		Побудова Windows-аплікацій з нестандартними елементами керування Windows Forms.	Контрольна робота	2	
Підсумковий контроль, форма		іспит в кінці семестру			

Пререквізити	Для вивчення курсу студенти потребують знань з попередніх семестрів навчання: основи ООП, використання винятків, взаємодія з потоками, використання контейнерів, модульне тестування. Одночасно з вивченням програмування студенти проходять навчальну обчислювальну практику, впродовж якої виконують завдання з програмування, поглиблюють набуті в курсі знання та удосконалюють навички.
Навчальні методи та техніки, які використовують під час викладання курсу	Лекції з мультимедійними презентаціями та з демонстрацією прийомів практичного використання середовища програмування; лабораторні заняття у вигляді проектування алгоритмів і програм, виконання практичних завдань, у тому числі у команді з 3-4 осіб; самостійне опрацювання навчальних матеріалів: підручників, конспектів лекцій, готових програм мовою C++, додаткових навчальних посібників, розміщених у хмарному сховищі (Moodle, Microsoft Teams, Google Classroom). Обговорення теоретичного та практичного матеріалу в онлайн сервісах, формулювання творчих завдань для студентів, виконання яких готує до вивчення нового теоретичного матеріалу.
Необхідне обладнання	Для проведення лекцій: комп'ютер, проектор, доступ до мережі інтернет. Для проведення лабораторних та виконання завдань: комп'ютер, ОС Windows, доступ до інтернету, середовище програмування мовою C# (Microsoft Visual Studio). Уся література, яку студенти не зможуть знайти самостійно, буде надана викладачем виключно в освітніх цілях без права її передачі третім особам. Студенти заохочуються до використання також й іншої літератури та джерел, яких немає серед рекомендованих.
Критерії оцінювання (окремо для кожного виду навчальної діяльності)	Оцінювання проводиться за 100-бальною шкалою. 50 балів нараховують за виконання лабораторних завдань і контрольних робіт, ще 50 балів – за виконання екзаменаційного завдання. Лабораторні завдання можуть бути індивідуальні та командні. Упродовж семестру студент виконує не менше 10 лабораторних робіт, кожен з яких оцінюють 3-5 балів залежно від складності. Оцінка за екзаменаційне завдання може бути поділена на дві частини: 20 балів за засвоєння теоретичного матеріалу, виставлені після опитувань упродовж семестру (у формі тестувань, колоквіумів тощо) та 30 за написання комп'ютерної програми (декількох програм). Завдання до курсу можуть передбачати самостійне опрацювання актуальних публікацій щодо програмування для .Net. Відвідання занять є важливою складовою навчання. Очікується, що всі студенти відвідають усі лекції і лабораторні заняття курсу. Активність під час проведення лекцій і лабораторних заохочується балами. У будь-якому випадку студенти зобов'язані дотримуватися усіх строків визначених для виконання усіх видів письмових робіт, передбачених курсом. Виконані роботи завантажують у відповідне хмарне сховище. Альтернативою відвідування лабораторних занять в університеті може бути дистанційна онлайн робота за розкладом проведення занять. Активність на лекціях і лабораторних ураховують при оцінюванні відповідного лабораторного завдання. Академічна доброчесність: очікується, що роботи студентів будуть їхнім оригінальними дослідженнями чи міркуваннями. Відсутність посилань на використані джерела, фабрикування джерел, списування, втручання в роботу інших студентів, здавання чужих комп'ютерних програм як своїх становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в письмовій роботі студента є підставою для її незарахування викладачем, незалежно від масштабів плагіату чи обману.
Опитування	Анкету-оцінку з метою оцінювання якості курсу буде надано після завершення курсу.