

**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 року)



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

Силабус навчальної дисципліни

**«Об'єктно-орієнтоване програмування»**

що викладається в межах ОПП “Комп’ютерна алгебра, криптологія і теорія ігор”, “Комп’ютерний аналіз математичних моделей”, “Математика. Математична економіка та економетрика”

першого (бакалаврського) рівня вищої освіти для здобувачів зі спеціальності 111 Математика

**Львів 2022**

<b>Назва дисципліни</b>	<b>Об'єктно-орієнтоване програмування</b>
<b>Адреса викладання дисципліни</b>	Львівський національний університет імені Івана Франка, вул. Університетська 1, м. Львів, Україна, 79000
<b>Факультет та кафедра, за якою закріплена дисципліна</b>	Факультет прикладної математики та інформатики, кафедра програмування
<b>Галузь знань, шифр та назва спеціальності</b>	11 – математика та статистика 111 – математика
<b>Викладачі дисципліни</b>	Ярошко Сергій Адамович, к. ф.-м. н., доцент, завідувач кафедри програмування; Пасічник Тимофій Васильович, к. ф.-м. н., доцент, доцент кафедри програмування
<b>Контактна інформація викладачів</b>	Електронна пошта: <a href="mailto:serhiy.yaroshko@lnu.edu.ua">serhiy.yaroshko@lnu.edu.ua</a> , <a href="mailto:tymofiy.pasichnyk@lnu.edu.ua">tymofiy.pasichnyk@lnu.edu.ua</a> веб-сторінки: <a href="https://ami.lnu.edu.ua/employee/yaroshko">https://ami.lnu.edu.ua/employee/yaroshko</a> <a href="https://ami.lnu.edu.ua/employee/pasichnyk-t-v">https://ami.lnu.edu.ua/employee/pasichnyk-t-v</a>
<b>Консультації з питань навчання по дисципліні відбуваються</b>	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.
<b>Сторінка курсу</b>	<a href="https://ami.lnu.edu.ua/course/basics-of-programming-informatics">https://ami.lnu.edu.ua/course/basics-of-programming-informatics</a>
<b>Інформація про дисципліну</b>	Курс "Об'єктно-орієнтоване програмування" є вибірковою дисципліною зі спеціальності 111 Математика для освітніх програм "Комп'ютерна алгебра, криптологія і теорія ігор", "Комп'ютерний аналіз математичних моделей", "Математика. Математична економіка та економетрика", яку викладають у п'ятому та шостому семестрі в обсязі 8 кредитів (за Європейською кредитно-трансферною системою ECTS)
<b>Коротка анотація дисципліни</b>	Фокус уваги курсу спрямовано на оволодіння базовими інструментами середовища .Net та особливості мови програмування C#: властивості, події, індикатори, методи розширення, використання інтерфейсів, серіалізація, життєвий цикл об'єктів, засоби рефлексії, LINQ, багатопотокове програмування. Розглянуто засоби Windows Forms для побудови застосунків.
<b>Мета та цілі дисципліни</b>	Метою дисципліни «Об'єктно-орієнтоване програмування» є ознайомити студента з базовими можливостями платформи .Net, навчити його використовувати мову C# для створення ефективних застосунків.
<b>Література для вивчення дисципліни</b>	<i>Основна література</i> 1. Learn C#. OFFICIAL COLLECTION: <a href="https://learn.microsoft.com/en-us/users/dotnet/collections/yz26f8y64n7k07?WT.mc_id=dotnet-35129-website">https://learn.microsoft.com/en-us/users/dotnet/collections/yz26f8y64n7k07?WT.mc_id=dotnet-35129-website</a> 2. C# programming guide <a href="https://learn.microsoft.com/en-us/dotnet/csharp/programming-guide/?WT.mc_id=dotnet-35129-website">https://learn.microsoft.com/en-us/dotnet/csharp/programming-guide/?WT.mc_id=dotnet-35129-website</a> 3. Object-oriented programming (C#) <a href="https://learn.microsoft.com/en-us/dotnet/csharp/fundamentals/tutorials/oo?source=recommendations">https://learn.microsoft.com/en-us/dotnet/csharp/fundamentals/tutorials/oo?source=recommendations</a> 4. Inheritance in C# and .NET <a href="https://learn.microsoft.com/en-us/dotnet/csharp/fundamentals/tutorials/inheritance">https://learn.microsoft.com/en-us/dotnet/csharp/fundamentals/tutorials/inheritance</a> 5. Language Integrated Query (LINQ) <a href="https://learn.microsoft.com/en-us/dotnet/csharp/programming-guide/concepts/linq/">https://learn.microsoft.com/en-us/dotnet/csharp/programming-guide/concepts/linq/</a>

<b>Обсяг курсу</b>	8 кредитів ЄКТС – 240 годин. З них 48 години лекцій, 64 години лабораторних занять та 128 годин самостійної роботи
<b>Очікувані результати навчання</b>	Після завершення цього курсу студент буде: <i>знати</i> <ul style="list-style-type: none"> <li>• засади функціонування платформи .Net;</li> <li>• синтаксис мови програмування C#, прийоми об'єктно-орієнтованого програмування мовою C#;</li> <li>• можливості та правила використання стандартних класів, зокрема, контейнерів, стандартних інтерфейсів;</li> <li>• призначення та влаштування компонент бібліотеки Windows Forms.</li> </ul> <i>вміти</i> <ul style="list-style-type: none"> <li>• оголошувати та використовувати у програмах мовою C# ієрархії класів, реалізовувати стандартні та власні інтерфейси;</li> <li>• налагоджувати взаємодію об'єктів через події;</li> <li>• серіалізувати об'єкти, програмно взаємодіяти з файловою системою;</li> <li>• використовувати LINQ для опрацювання колекцій даних;</li> <li>• будувати застосунки за допомогою Windows Forms;</li> <li>• створювати багатопотокові застосунки.</li> </ul>
<b>Ключові слова</b>	Платформа .Net, мова програмування C#, клас, інтерфейс, властивість, подія, делегат, LINQ, серіалізація, багатопотоковість.
<b>Формат курсу</b>	Очний: проведення лекцій, лабораторних робіт та консультацій в приміщеннях університету, а в умовах карантину – онлайнний на платформі Microsoft Teams
<b>Теми</b>	
<b>Підсумковий контроль, форма</b>	заліки в кінці семестрів
<b>Пререквізити</b>	Для вивчення курсу студенти потребують знань з попередніх семестрів навчання: основи мови програмування «Python».
<b>Навчальні методи та техніки, які використовують під час викладання курсу</b>	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, додаткових навчальних посібників, розміщених у хмарному сховищі (Moodle, Microsoft Teams, Google Classroom). Обговорення теоретичного та практичного матеріалу в онлайн сервісах.
<b>Необхідне обладнання</b>	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.
<b>Критерії оцінювання (окремо для кожного виду навчальної діяльності)</b>	<b>Оцінювання</b> проводиться за 100-бальною шкалою. 100 балів нараховують за виконання лабораторних завдань і контрольних робіт. Лабораторні завдання можуть бути індивідуальні та командні. Упродовж семестру студент виконує не менше 10 лабораторних робіт, кожен з яких оцінюють 5-12 балів залежно від складності. Контрольні роботи направлені на перевірку засвоєння теоретичного матеріалу, оцінюються сумарно до 20 балів, виконуються у формі тестувань, колоквиумів тощо. Завдання до курсу можуть передбачати самостійне опрацювання актуальних публікацій щодо програмування для .Net. <b>Відвідання занять</b> 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

	<p>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.</p> <p><b>Академічна доброчесність:</b> 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.</p>
<b>Опитування</b>	Анкету-оцінку з метою оцінювання якості курсу буде надано після завершення курсу.

### Схема курсу

Тижд	Тема, план, короткі тези	Форма заняття	Тривалість, год	Оцінювання
<b>5-ий семестр</b>				
1	Л.1. Архітектура, призначення, можливості платформи .Net. Система типів, операції мови С#. Типи-значення та типи-посилання С#. Консольне введення-виведення. Конструкції керування мови С#.	Лекція	2	
	Побудова консольних програм в середовищі Visual Studio засобами мови С#. Форматоване виведення.	Лабораторна робота	2	5
2	Послідовні, галужені, циклічні алгоритми.	Лабораторна робота	2	5
3	Л.2. Методи System. Object. Перелік - найпростіший тип користувача. Масиви С#: одно- та багатовимірні, зубчасті. Текстові змінні.	Лекція	2	
	Послідовні, галужені, циклічні алгоритми з використанням вбудованих типів.	Лабораторна робота	2	5
4	Обробка текстової інформації	Лабораторна робота	2	5
5	Л.3. Основні принципи об'єктно-орієнтованого програмування. Оголошення класів і структур мовою С#: поля, методи, властивості. Передавання аргументів методам. Конструювання, копіювання об'єктів.	Лекція	2	
	Оголошення та використання переліків.	Лабораторна робота	2	7
6	Створення та робота з масивами в С#.	Лабораторна робота	2	7
7	Л.4. Наслідування. Множинне наслідування. Віртуальні, сховані методи. Абстрактні класи.	Лекція	2	
	Оголошення простих класів. Побудова та використання властивостей.	Лабораторна робота	2	7
8	Побудова ітерацій для заданого n.	Лабораторна робота		7
9	Л.5. Статичні члени класу, статичні класи. Методи, що розширюють клас. Закриті класи і методи. Часткові класи.	Лекція	2	
	Ітераційні обчислення з точністю $\epsilon > 0$ .	Лабораторна робота	2	7
10	Оголошення та використання простих класів. Побудова та використання властивостей.	Лабораторна робота	2	7
11	Операції мови С#: is, as, поглинання null та інші спеціальні. Перетворення типів. Перевірка рівності об'єктів. Перевантаження операцій. Методи-індексатори.	Лекція	2	
	Оголошення та використання структур. Побудова та використання методів.	Лабораторна робота	2	7

12	Інтерфейси: означення, можливості використання. Стандартні інтерфейси IComparable, IEnumerable, ICloneable. Наслідування класів та інтерфейсів.	Лабораторна робота		7
13	Розробка узагальнених типів. Узагальнення класів. Узагальнення інтерфейсів. Обмеження узагальнень. Засоби узагальнених класів. Питання обсягу коду та продуктивності.	Лекція	2	
	Операції мови C#: is, as, поглинання null та інші спеціальні. Перетворення типів	Лабораторна робота	2	7
14	Перевантаження операцій. Використання власних індексаторів. Порівняння об'єктів. Перетворення типів.	Лабораторна робота		7
15	Робота з файлами. Класи Path, FileStream, StreamReader, StreamWriter. Маніпулювання файлами і директоріями. Потoki даних. Формати серіалізації: двійковий, XML, SOAP.	Лекція	2	
	Контрольна робота	Контрольна робота	2	10
16	Захист лабораторних робіт. Підсумкова атестація.	Лабораторна робота		
<b>6-ий семестр</b>				
1	Простір імен System.Collection. Проблеми, пов'язані з неузагальненими колекціями.	Лекція	2	
	Оголошення та використання інтерфейсів. Оголошення класів, що реалізують стандартні інтерфейси.	Лабораторна робота	2	5
2	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	
	Побудова ієрархії класів. Оголошення і використання конструкторів, фіналізаторів, віртуальних методів.	Лабораторна робота	2	5
3	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	
	Робота з файлами. Створення файлів. Зчитування даних з файлів та запис даних у файли.	Лабораторна робота	2	6
4	Винятки та збірки Net. Опрацювання винятків. Побудова бібліотек класів.	Лекція	2	
	Робота зі списками, використовуючи List, LinkedList, Queue.	Лабораторна робота	2	6
5	Розширення мови C# та мова LINQ. Динамічне розширення мови.	Лекція	2	
	Синхронний та асинхронний виклики методів через делегата. Визначення події, організація взаємодії об'єктів через подію.	Контрольна робота	2	6
6	Побудова Windows-аплікацій з використанням Windows Forms. Стандартні елементи керування. Діалоги. Приклади побудови застосунків: перша програма, модель світлофора	Лекція	2	
	Розробка узагальнених класів. Створення класів винятків. Запуск, перехоплення та опрацювання винятків.	Лабораторна робота	2	6
7	Приклади побудови застосунків: тестування, редактор текстів, багатівіконні програми. Меню, діалоги, панелі інструментів, рядок стану.	Лекція	2	
	Використання інструментів рефлексії для програмного дослідження класів. Побудова та використання атрибутів користувача.	Лабораторна робота	2	6
8	Програмний пошук папок, файлів. Зберігання до файла (у різних форматах) та завантаження об'єктів з файла.	Лекція	2	
	Контрольна робота.	Контрольна робота	2	10
9	Побудова Windows-аплікацій зі стандартними елементами керування Windows Forms. Файлові діалоги.	Лекція	2	
	Використання колекцій .Net. Робота з рядками, текстом.	Лабораторна робота	2	6
10	Створення складних Windows Forms User Control, забезпечення підтримки етапу проектування аплікації	Лекція	2	
	Побудова багатопотокових Windows-аплікацій: окремий потік для взаємодії з GUI, окремий для обчислень тощо.	Лабораторна робота	2	6

11	Багатопотоковість і синхронізація. Асинхронні делегати. Класи Thread, Task. Синхронізація. Багатопотокові аплікації	Лекція	2		
	Взаємодія з менеджером купи. Реалізація фіналізаторів. Реалізація інтерфейсу IDisposable.	Лабораторна робота	2	7	
12	Життєвий цикл об'єктів. Finalize. IDisposable. Менеджер керованої купи, збирання сміття.	Лекція	2		
	LINQ: розширення типів, лямбда вирази, LINQ- запити	Лабораторна робота	2	7	
13	Основи WPF та XAML. Властивості й атрибути. Властивості та елементи. Контейнери.	Лекція	2		
	Побудова Windows-аплікацій з нестандартними елементами керування Windows Forms.	Лабораторна робота	2	7	
14	Класи Geometry, Drawing та Visual. Комбінування фігур. Міні-мова опису геометрії.	Лекція	2		
	Побудова WPF-аплікацій. Створення фігур та їх анімація.	Лабораторна робота	2	7	
15	Основи анімації WPF. Класи анімації. Реалізація анімації в кодї C#. Декларативна анімація і розкадровки. Роль тригерів подій.	Лекція	2		
	Контрольна робота з WPF.	Контрольна робота	2	10	
16	Тривимірні графіка в WPF. Геометрична модель і поверхні. Джерела світла. Положення камери. Затінення та нормалі.	Лекція	2		
	Захист лабораторних робіт. Підсумкова атестація	Лабораторна робота	2		