




## Парадигма функціонального програмування

Галузі знань: 10 Природничі науки, 11 Математика та статистика, 12 Інформаційні технології, 15 Автоматизація та приладобудування, 16 Хімічна та біоінженерія, 17 Електроніка та телекомунікації, 19 Архітектура та будівництво, 27 Транспорт

<b>Рівень вищої освіти</b>	<i>перший (бакалаврський)</i>
<b>Статус дисципліни</b>	<i>вибіркова (Дисципліна індивідуального вибору 2)</i>
<b>Обсяг дисципліни</b>	150 годин / 5 кредитів ЄКТС
<b>Мова викладання</b>	<i>українська</i>
<b>Що буде вивчатися (предмет вивчення)</b>	<p>Функціональне програмування є парадигмою програмування, у якій основна увага приділяється функціям як основним будівельним блокам програми. Мови програмування F# та Scala (і не тільки) є потужним інструментом для функціонального програмування та мають багатий набір функціональних можливостей.</p> <ol style="list-style-type: none"> <li><i>Вступ до функціонального програмування:</i> історія та еволюція функціонального програмування, мотивація та основні концепції, порівняння з іншими парадигмами програмування (імперативна, об'єктно-орієнтована).</li> <li><i>Чисті функції:</i> визначення та важливість чистих функцій, переваги чистих функцій для тестування та налагодження, приклади чистих функцій в різних мовах програмування.</li> <li><i>Імутабельність:</i> принципи незмінності даних, переваги імутабельності для паралельних обчислень та безпеки коду, використання імутабельних структур даних.</li> <li><i>Функції вищого порядку:</i> функції, що приймають інші функції як аргументи або повертають функції як результат, приклади функцій вищого порядку та їх застосування, каррінг (currying) та часткове застосування функцій.</li> <li><i>Лямбда-вирази та анонімні функції:</i> визначення та використання лямбда-виразів, приклади анонімних функцій у різних мовах, переваги та недоліки використання анонімних функцій.</li> <li><i>Рекурсія:</i> основи рекурсії та її відмінності від ітерації, приклади рекурсивних алгоритмів, оптимізація рекурсії за допомогою хвостової рекурсії.</li> <li><i>Функціональні структури даних:</i> списки, кортежі, множини та інші структури даних, використання функціональних структур даних для обробки інформації, порівняння з імперативними структурами даних.</li> <li><i>Зіставлення зі зразком:</i> основи співставлення з шаблоном (pattern matching), приклади використання співставлення з шаблоном для обробки складних структур даних, переваги співставлення з шаблоном для чистоти та зрозумілості коду.</li> <li><i>Ледаче обчислення:</i> принципи та механізми ледачого обчислення, переваги та недоліки ледачого обчислення, приклади застосування ледачого обчислення для підвищення продуктивності.</li> <li><i>Композиція функцій:</i> основи композиції функцій, приклади створення складних функцій з простих, переваги композиції функцій для модульності та повторного використання коду.</li> <li><i>Монадична обробка даних:</i> вступ до монад та їх використання, приклади монад в різних мовах програмування, використання монад для обробки побічних ефектів та асинхронних операцій</li> </ol>
<b>Чому це цікаво/треба вивчати (мета)</b>	<ol style="list-style-type: none"> <li>Розуміння різних підходів до програмування (функціональне, об'єктно-орієнтоване, імперативне) дозволяє обирати найбільш підходящі інструменти і методи для конкретних завдань.</li> <li>Функціональне програмування вчить мислити на більш високому рівні абстракції, що корисно для вирішення складних проблем.</li> <li>Багато нових ідей і методів програмування народжуються в академічних колах, де функціональне програмування є дуже популярним.</li> <li>Функціональне програмування може бути застосоване для розробки різних видів програм, включаючи веб-додатки, наукові обчислення, обробку даних та інші.</li> </ol>

	Наприклад, F# також інтегрується з платформою .NET, що дозволяє використовувати його в екосистемі Microsoft та взаємодіяти з іншими мовами програмування, бібліотеками та інструментами		
<b>Як можна користуватися набутими знаннями і вміннями (компетентності)</b>	<p><i>Розширення набору навичок.</i> Вивчення функціонального програмування дозволяє програмісту розширити свій набір навичок та підходів до розробки. Це дозволяє вирішувати завдання ефективніше, а також забезпечує нові інструменти та техніки, які можна застосовувати у своїй роботі.</p> <p><i>Розробка якіснішого коду.</i> Функціональне програмування звертає увагу на чистоту, ясність та безпеку коду. Вивчення функціонального програмування допомагає програмісту писати більш читаний, модульний та надійний код.</p> <p><i>Розробка більш гнучких рішень.</i> Функціональне програмування дозволяє створювати гнучкі та абстрактні рішення, використовуючи концепції, такі як функції вищих порядків та карування. Це дозволяє програмісту створювати код, який легко адаптується до змін вимог чи умов.</p> <p><i>Підвищення продуктивності та ефективності.</i> Функціональне програмування передбачає використання незмінних даних та уникнення побічних ефектів, що може сприяти підвищенню продуктивності та спрощенню тестування програмного забезпечення.</p> <p><i>Робота з паралельними та асинхронними операціями.</i> Функціональне програмування пропонує ефективні інструменти для роботи з паралельними та асинхронними операціями. Вивчення функціонального програмування дозволяє програмісту освоїти ці інструменти та створювати масштабовані та чуйні програми.</p> <p><i>Розуміння альтернативних підходів.</i> Вивчення функціонального програмування дозволяє програмісту отримати розуміння альтернативних підходів до програмування та розширити своє мислення у розробці програмного забезпечення</p>		
<b>Пререквізити</b>	Бажано «Основи програмування» та «Комп'ютерна дискретна математика»		
<b>Кореквізити</b>			
<b>Організація навчання</b>	<p>Види занять: лекції, практичні заняття.</p> <p>Форми здобуття освіти: денна, заочна, дистанційна, дуальна.</p> <p>Форми контролю: модульний контроль, іспит</p>		
<b>Кафедра</b>	Кафедра інженерії програмного забезпечення		
<b>Факультет</b>	Програмної інженерії та бізнесу		
<b>Викладач</b>		<b>ПІБ</b>	<b>Кузнєцова Юлія Анатоліївна</b>
		<b>Посада</b>	Доцент
		<b>Вчене звання</b>	Доцент
		<b>Науковий ступінь</b>	Кандидат технічних наук
		<b>e-mail</b>	<a href="mailto:y.kuznetsova@khai.edu">y.kuznetsova@khai.edu</a>
		<b>Персональна сторінка</b>	<a href="https://se.khai.edu/kuznecova/">https://se.khai.edu/kuznecova/</a>
<b>Посилання на електронні матеріали курсу</b>	<a href="https://mentor.khai.edu/course/view.php?id=8443">https://mentor.khai.edu/course/view.php?id=8443</a>		
<b>Посилання на робочу програму (силабус)</b>			