

Силабус навчальної дисципліни

№	Назва поля	Детальний контент, коментарі
1.	Назва факультету	Факультет Комп'ютерних наук
2.	Рівень вищої освіти	Перший (бакалаврський)
3.	Код і назва спеціальності	F3 Комп'ютерні науки
4.	Тип і назва освітньої програми	ОПП «Штучний інтелект»
5.	Назва дисципліни (інформація з ЦІСТ)	Об'єктно-орієнтоване програмування
6.	Кількість ЄКТС кредитів	5
7.	Структура дисципліни (розподіл за видами та годинами навчання)	Лекції – 30 годин, практичні заняття – 30 годин, консультації – 10 годин, самостійна робота – 80 годин. Семестровий контроль – екзамен.
8.	Графік (терміни) вивчення дисципліни	1-й рік, 2-й семестр
9.	Передумови для навчання за дисципліною	Раніше повинна бути вивчена дисципліна Алгоритмізація та програмування.
10.	Анотація (зміст) дисципліни	<p>Обов'язкова дисципліна циклу професійної та практичної підготовки за спеціальністю містить такі змістовні модулі:</p> <p>1. Концепція об'єктно-орієнтованого програмування Тема 1. Мета курсу. Передумови виникнення об'єктно-орієнтованого підходу. Співвідношення між об'єктно-орієнтованим аналізом, проектуванням та програмуванням. Тема 2. Поняття об'єкту. Поняття класу. Відносини між класами. Відносини між об'єктами. Тема 3. Складові частини об'єктної моделі: абстрагування, інкапсуляція, спадкування. Тема 4. Основи об'єктно-орієнтованого проектування мовою UML.</p> <p>2. Об'єктно-орієнтоване програмування мовою C++ Тема 1. Сутність абстрагування. Реалізація класів у C++. Призначення та особливості використання конструкторів і деструкторів. Тема 2. Сутність інкапсуляції. Реалізація приховування даних у C++. Особливості доступу до відкритих і закритих членів класу. Тема 3. Поняття інтерфейсу класа. Принцип відділення інтерфейсу класа від його реалізації. Тема 4. Класифікація функцій-елементів класу. Класифікація конструкторів класу. Призначення та характеристика різних типів конструкторів. Тема 5. Принцип найменших привілеїв у програмуванні. Константні та статичні члени класу. Тема 6. Поняття перевантаження операцій. Основні особливості перевантаження операцій у C++.</p>

		<p>Тема 7. Відношення спадкування між класами. Види спадкування в C++. Реалізація в C++ відкритого одиночного спадкування. Захищені елементи класу. Використання та перевизначення функцій базового класу в похідних.</p> <p>Тема 8. Поняття поліморфізму. Підтримка поліморфізму в C++. Віртуальні функції. Чисто віртуальні функції. Абстрактні базові класи.</p> <p>3. Використання бібліотек класів у об'єктно-орієнтованому програмуванні.</p> <p>Тема 1. Огляд та структура бібліотеки стандартних шаблонів.</p> <p>Тема 2. Основи програмування, керованого подіями.</p> <p>Тема 3. Особливості та порядок створення програмних застосувань з візуальним інтерфейсом у середовищі Visual Studio.</p>
11.	Компетентності, знання, вміння, розуміння, якими оволодіє здобувач вищої освіти в процесі навчання	<p>Дисципліна сприяє формуванню та розвитку таких компетентностей:</p> <p>ЗК2. Здатність застосовувати знання у практичних ситуаціях.</p> <p>ЗК4. Здатність спілкуватися державною мовою як усно, так і письмово.</p> <p>ЗК6. Здатність вчитися й оволодівати сучасними знаннями.</p> <p>ЗК11. Здатність приймати обґрунтовані рішення.</p> <p>ЗК12. Здатність оцінювати та забезпечувати якість виконуваних робіт.</p> <p>ФК8. 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>ФК10. 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>
12.	Результати навчання здобувача вищої освіти	<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>ПР14. Застосовувати знання методології та CASE-засобів проектування складних систем, методів структурного аналізу систем, об'єктно-орієнтованої методології проектування в процесі побудови і практичного застосування функціональних моделей організаційно-економічних і виробничо-технічних систем.</p>
13.	Система оцінювання	1.Індивідуальні завдання.

	відповідно до кожного завдання для складання заліку/екзамену	2.Для підсумкового контролю у формі екзамену для оцінювання роботи студента протягом семестру використовують підсумкову рейтингову оцінку $R_p = O_{sem} + O_{exam}$. Оцінку за семестр обчислюють як суму оцінок за різні види занять. Підсумкова рейтингова оцінка за семестр обчислюється у 100-бальній системі.
14.	Якість освітнього процесу	Дотримання принципів академічної доброчесності (http://lib.nure.ua/plagiat). Оновлення робочої програми дисципліни – 2020 р. Використовуються: мова програмування C++, середовище розробки Visual Studio 2022 та відповідні бібліотеки.
15.	Методичне забезпечення	1. Комплекс навчально-методичного забезпечення навчальної дисципліни «Об'єктно-орієнтоване програмування» для студентів усіх форм навчання за спеціальністю 122 Комп'ютерні науки, ОПП «Штучний інтелект» [Електронний ресурс] / ХНУРЕ; розроб. Губін В.О. -Х. ХНУРЕ, 2017.- 88 с. 2. Методичні вказівки до курсового проектування з дисципліни «Об'єктно-орієнтоване програмування» для студентів усіх форм навчання за спеціальністю 122 Комп'ютерні науки, ОПП «Штучний інтелект» [Електронний ресурс] / ХНУРЕ; розроб. Губін В.О. -Х. ХНУРЕ, 2021.- 20 с.
16.	Розробник силабусу(посада, ПІБ, ел. пошта)	В.О. Губін, ст. викл. каф. ШІ. e-mail: vadim.gubin@nure.ua