



Декан факультету інформаційних технологій

Тетяна ГОВОРУЩЕНКО
Ім'я, ПРІЗВИЩЕ

09 2025 р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Алгоритмізація та програмування

Назва дисципліни

Галузь знань – F Інформаційні технології
Спеціальність – F3 Комп'ютерні науки
Рівень вищої освіти – Перший (бакалаврський)
Освітньо-професійна програма – Комп'ютерні науки
Обсяг дисципліни – 13 кредитів ЄКТС
Шифр дисципліни – ОЗП.05
Мова навчання – Українська
Статус дисципліни: Обов'язкова (загальної підготовки)
Факультет – Інформаційних технологій
Кафедра – Комп'ютерних наук

Форма здобуття освіти	Курс	Семестр	Загальний обсяг		Кількість годин							Курсовий проєкт*	Курсова робота	Форма семестрового контролю	
					Аудиторні заняття					Самостійна робота, у т.ч. ІРС	Залік			Іспит	
			Кредити ЄКТС	Години	Разом	Лекції	Лабораторні роботи	Практичні заняття	Семінарські заняття						
Д	1	1	10	300	98	32	66			202				+	
Д	1	2	3	90	34	16	18			56	+		+		
Разом ДФН			13	390	132	48	84			258	2		2	1	

Примітка. З навчальної дисципліни у 2 семестрі передбачений курсовий проект, зміст та вимоги до виконання якого регулюються відповідними методичними рекомендаціями.

Робоча програма складена на основі освітньо-професійної програми «Комп'ютерні науки» за спеціальністю F3 «Комп'ютерні науки»

Робоча програма складена О. Собко
Підпис автора

д-р філософії Олена СОБКО

Науковий ступінь, вчене звання, Ім'я, ПРІЗВИЩЕ автора

Олександр Мазурець
Підпис автора

канд. техн. наук, доц. Олександр МАЗУРЕЦЬ

Науковий ступінь, вчене звання, Ім'я, ПРІЗВИЩЕ автора

Схвалена на засіданні кафедри Комп'ютерних наук

Протокол від 29.08.2025 № 1.

Зав. кафедри

Олександр Бармак
Підпис

Олександр БАРМАК
Ім'я, ПРІЗВИЩЕ

Робоча програма розглянута та схвалена вченою радою факультету інформаційних технологій

Голова вченої ради факультету

Тетяна Говорущенко
Підпис

Тетяна ГОВОРУЩЕНКО
Ім'я, ПРІЗВИЩЕ

ЛИСТ ПОГОДЖЕННЯ

Посада	Назва кафедри	Підпис	Ініціали, ПРИЗВИЩЕ
Завідувач кафедри, д-р. техн. наук, проф.	Комп'ютерних наук		Олександр БАРМАК
Гарант освітньо-професійної програми, канд. техн. наук, доц.	Комп'ютерних наук		Олександр МАЗУРЕЦЬ

3. Пояснювальна записка

Дисципліна «Алгоритмізація та програмування» є обов'язковою освітньою компонентою загальної підготовки здобувачів першого (бакалаврського) рівня вищої освіти, які навчаються за освітньо-професійними програмами в межах спеціальності ФЗ «Комп'ютерні науки». Двосеместрова дисципліна обсягом 15 кредитів ЄКТС, складається із теоретичного курсу, лабораторних робіт, індивідуальних завдань та курсового проєкту. Курсовий проєкт обсягом 2 кредити ЄКТС.

Пререквізити – вихідна

Кореквізити – Алгоритмізація та програмування (курсний проєкт) (ОЗП.06) Статистична обробка даних (ОЗП.08), Методи та системи штучного інтелекту (ОФП.01), Вебтехнології (ОФП.03), Інтелектуальний аналіз даних (ОФП.05), Об'єктно-орієнтоване проєктування (ОФП.08), Технології створення програмних продуктів (ОФП.09), Основи програмної інженерії та тестування програмного забезпечення (ОФП.10), Технології паралельних обчислень (ОФП.11), Системне та мережеве програмне забезпечення (ОФП.14), Кваліфікаційна робота (ОФП.18), КП Методи та системи штучного інтелекту (ОФП.02), КП Вебтехнології (ОФП.04), КП Інтелектуальний аналіз даних (ОФП.06), Професійна практика (ОФП.17). (це взято з окремої комірки АіП КП)

Відповідно до освітньої програми дисципліна сприяє забезпеченню:

компетентностей: здатність розв'язувати складні спеціалізовані задачі та практичні проблеми у галузі комп'ютерних наук або у процесі навчання, що передбачає застосування теорій та методів інформаційних технологій і характеризується комплексністю та невизначеністю умов (ІК); здатність до абстрактного мислення, аналізу та синтезу (ЗК 01); здатність застосовувати знання у практичних ситуаціях (ЗК 02); здатність та розуміння предметної області та розуміння професійної діяльності (ЗК 03); здатність спілкуватися державною мовою як усно, так і письмово (ЗК 04); здатність вчитися й оволодівати сучасними знаннями (ЗК 06); здатність до пошуку, оброблення та аналізу інформації з різних джерел (ЗК 07); здатність генерувати нові ідеї (креативність) (ЗК 08); здатність працювати в команді (ЗК 09); здатність бути критичним і самокритичним (ЗК 10); здатність приймати обґрунтовані рішення (ЗК 11); здатність оцінювати та забезпечувати якість виконуваних робіт (ЗК 12); здатність зберігати та примножувати моральні, культурні, наукові цінності і досягнення суспільства на основі розуміння історії та закономірностей розвитку предметної області, її місця у загальній системі знань про природу і суспільство та у розвитку суспільства, техніки і технологій, використовувати різні види та форми рухової активності для активного відпочинку та ведення здорового способу життя (ЗК 15); здатність ухвалювати рішення та діяти, дотримуючись принципу неприпустимості корупції та будь-яких інших проявів недобросовісності (ЗК 16); здатність до логічного мислення, побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проєктування, 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 (ФК 03); здатність проєктувати 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 (ФК 08).

програмих результатів навчання: 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 (ПРН 01); проєктувати, розробляти та аналізувати алгоритми розв'язання обчислювальних та логічних задач, оцінювати ефективність та складність алгоритмів на основі застосування формальних моделей алгоритмів та обчислюваних функцій (ПРН 05); 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 (ПРН 09); професійно розвиватися, опрацьовувати україномовні та англійськомовні джерела предметної області, усвідомлювати необхідність навчання впродовж усього життя з метою поглиблення набутих та здобуття нових фахових знань у галузі комп'ютерних наук, адаптуватися до роботи за конкретною професією, пропагувати ведення активного та здорового способу життя як ефективної складової професійного розвитку (ПРН 17).

Мета дисципліни. Формування знань, вмінь та навичок, необхідних для розробки алгоритмів та програм для задач оптимізації, розробки алгоритмів та програм аналізу даних, розробки прикладного програмного забезпечення інформаційних систем і технологій, створення та налагодження програмного інтерфейсу, візуальних компонент програмного забезпечення інформаційних систем і технологій для вирішення задач у галузі комп'ютерних наук.

Предмет дисципліни. Теоретичні принципи та практичні прийоми структурного, процедурного, модульного програмування; технологія розробки алгоритмів вирішення прикладних задач, кодування обраною мовою програмування, налагодження програми.

Завдання дисципліни. Надати студентам знання технології розробки алгоритмів вирішення прикладних задач у галузі комп'ютерних наук, мов програмування та їх елементів, алгоритмічних структур; та навички застосування мови програмування C#, типів даних, констант, операторів, виразів, алгоритмічних структур при розробці програмного забезпечення у галузі комп'ютерних наук, що характеризуються комплексністю та невизначеністю умов.

Результати навчання. Студент, який успішно завершив вивчення дисципліни, повинен: знати – основні обчислювальні та пошукові алгоритми, програмні засоби та способи відображення даних, основні інструменти програмування та їх застосування; уміти – читати та пояснювати алгоритми; обговорювати алгоритмічну коректність; використовувати та адаптувати стандартні алгоритми; розробляти алгоритми; читати та пояснювати значення програмних об'єктів; створювати, використовувати та змінювати програми, читати та пояснювати результати операцій; реалізовувати та описувати операції; створювати програми для реалізації стандартних алгоритмів; виявляти

та виправляти логічні та синтаксичні помилки в програмному коді; описувати процес трансляції програми; визначати зображення різних типів даних за допомогою різних форматів; успішно використовувати інструменти програмування під час розробки програмного забезпечення; ефективно застосовувати стандартні комп'ютерні застосування.

4. Структура залікових кредитів дисципліни

Тема	Кількість годин відведених на		
	Денна форма навчання		
	Лекції	Лабораторні роботи	СРС
1-й семестр			
Тема 1. Вступ до фаху. Алгоритмізація та проектування алгоритмів.	2	8	8
Тема 2. Вступ до програмування	2		8
Тема 3. Мови програмування та їх елементи. Мова програмування C#.	2	8	8
Тема 4. Константи, змінні. Типи даних. Оператори. Вирази. Методи.	2		8
Тема 5. Алгоритмічна структура розгалуження мовою програмування.	2	8	8
Тема 6. Алгоритмічна структура повторення мовою програмування.	2		8
Тема 7. Парадигми програмування.	2	8	12
Тема 8. Масиви. Одновимірні масиви.	2		8
Тема 9. Багатовимірні масиви.	2	8	12
Тема 10. Символьний та рядковий типи даних.	2		12
Тема 11. Стандартні класи для роботи з масивами та рядками.	2	8	12
Тема 12. Користувальчік інтерфейси .NET Framework.	2		12
Тема 13. Windows Forms.	2	8	12
Тема 14. Обробка виняткових ситуацій.	2		14
Тема 15. Робота з файловими структурами.	2	10	20
Тема 16. Робота з спеціалізованими файлами.	2		20
	32	66	202
2-й семестр			
Тема 17. Робота з інформаційними джерелами при вирішенні практичних задач в галузі комп'ютерних наук.	2	2	4
Тема 18. Робота з інформаційними джерелами для вибору засобів розробки програмного забезпечення.	2	2	4
Тема 19. Етапи розроблення та впровадження програмного забезпечення.	2	4	6
Тема 20. Основи тестування програмного забезпечення.	2	2	8
Тема 21. Використання структур даних при розробці програмного забезпечення.	2	2	6
Тема 22. Основи багатопоточності.	2	2	6
Тема 23. Основи Windows Presentation Foundation.	2	2	8
Тема 24. Засоби Windows Presentation Foundation.	2	2	14
	16	18	56
Разом	48	84	258

5. Програма навчальної дисципліни

5.1 Зміст лекційного курсу

Номер лекції	Перелік тем лекцій, їх анотації	Кількість годин
	1-ий семестр	
	Тема 1. Вступ до фаху. Алгоритмізація та проєктування алгоритмів.	2
1	Комп'ютерні науки – вступ до спеціальності. Чинники успішного професійного розвитку та діяльності. Історія розвитку предметної області комп'ютерних наук. Алгоритм як основне поняття програмування. Властивості алгоритмів. Типи алгоритмічних структур: лінійна структура, алгоритмічна структура розгалуження, алгоритмічна конструкція повторення. Ефективність алгоритмів. Способи подання алгоритмів. Блок-схеми. Засоби побудови блок-схем та текстові редактори. Літ.: [1] с. 10-19, [3] с. 1-9	2
2	Тема 2. Вступ до програмування	2
	Архітектура комп'ютера. Програмне забезпечення комп'ютера. Етапи розробки програмного забезпечення. Огляд парадигм програмування. Комп'ютерні платформи. Платформа .NET Framework. Огляд сучасних інтегрованих систем програмування. Середовище програмування Visual Studio. Транслятори, редактори, компонувальники та налагоджувачі. Процедурне програмування. Основи синтаксису C# (типи даних, константи, змінні, ініціалізація даних, введення й виведення даних). Літ.: [1] с. 4-9, [2] с. 7-27	2
	Тема 3. Мови програмування та їх елементи. Мова програмування C#.	2
3	Поняття про технології та мови програмування. Класифікація мов програмування. Елементи мови програмування. Алфавіт, синтаксис та семантика мови програмування C#. Літ.: [1] с. 20-25, [2] с. 28-39	2
	Тема 4. Константи, змінні. Типи даних. Оператори. Вирази. Методи.	2
4	Константи та змінні. Типи даних C#. Ініціалізація даних. Вирази (унарні, бінарні та присвоєння). Операції послідовного присвоєння, умовного та явного перетворення типів. Пріоритет та асоціативність операцій. Методи та стандартні бібліотеки. Організація введення-виведення. Клас Math як основа роботи з математичними виразами. Літ.: [1] с. 26-31, [2] с. 40-57, [3] с. 24-29	2
	Тема 5. Алгоритмічна структура розгалуження мовою програмування.	2
5	Умовні конструкції розгалуження «if» та «if-else». Оператор вибору «switch». Переривання виконання алгоритму. Вкладені умови. Літ.: [1] с. 32-39, [2] с. 58-63, [3] с. 48-52	2
	Тема 6. Алгоритмічна структура повторення мовою програмування.	2
6	Оператор циклу з умовою «for», поняття лічильника. Оператор «foreach». Оператори циклу «while», «do-while». Переривання виконання циклу. Вкладені цикли. Літ.: [1] с. 40-47, [2] с. 64-70, [3] с. 53-67	2
	Тема 7. Парадигми програмування.	2
7	Проєктування та розробка програмного забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного. Методи (функції). Приклади програм із застосуванням різних парадигм програмування. Введення в об'єктно-орієнтоване програмування. Літ.: [1] с.80-84, [2] с. 71-141	2
	Тема 8. Масиви. Одновимірні масиви.	2
8	Одновимірні масиви. Способи заповнення масивів. Алгоритми пошуку в масивах (послідовностях). Літ.: [1] с.48-55, [2] с. 165-167, [3] с. 77-96	2
	Тема 9. Багатовимірні масиви.	2
9	Двовимірні масиви. Багатовимірні масиви. Ступінчаті масиви, масиви масивів. Алгоритми пошуку в масивах (послідовностях). Приклади програм, що використовують багатовимірні масиви. Літ.: [1] с. 56-62, [2] с. 168-177, [3] с. 105-116	2

	Тема 10. Символьний та рядковий типи даних.	2
10	Символьний тип даних, символна константа. Рядки, константа-рядок. Введення-виведення рядків та масивів символів. Розташування у пам'яті. Обробка рядків: вставлення та видалення символів, аналіз слів, копіювання та поєднання рядків. Алгоритми пошуку підрядків у рядках. Алгоритми та демонстраційні програми роботи з текстами. Літ.: [1] с.63-68, [2] с. 42-44, [3] с. 39-47	2
	Тема 11. Стандартні класи для роботи з масивами та рядками.	2
11	Методи класу Array для роботи з масивами. Методи класу StringBuilder для роботи з масивами та рядками. Форматування рядків. Літ.: [1] с. 69-74, [2] с. 176-177, [3] с. 87-92	2
	Тема 12. Користувацькі інтерфейси .NET Framework.	2
12	Поняття користувацького інтерфейсу. Типи користувацького інтерфейсу. Загальні відомості про Windows Forms, Windows Presentation Foundation, Universal Windows Platform. Літ.: [2] с. 208-222	2
	Тема 13. Windows Forms.	2
13	Введення у Windows Forms, Структура програми Windows Forms. Спадкування класу Form. Робота з формами та елементами управління. Програмне створення елементів керування вікна. Елементи керування з панелі Toolbox Літ.: [1] с. 101-105, [2] с. 223-236	2
	Тема 14. Обробка виняткових ситуацій.	2
14	Відлагодження програмного забезпечення. Поняття виняткової ситуації. Обробка винятків у мові C#. Принципи обробки винятків. Послідовність обробки винятків. Блоки «try» та «catch». Оператор «throw». Створення власних винятків. Літ.: [2] с. 143-156	2
	Тема 15. Робота з файловими структурами.	2
15	Файл. Формати файлів. Створення файлів за допомогою середовища програмування. Алгоритми читання та запису інформації в текстовий файл. Літ.: [1] с.75-79, [2] с. 281-284	2
	Тема 16. Робота з спеціалізованими файлами.	2
16	Запис та обробка інформації на прикладі програм Microsoft Word, Excel та створення XML-документів Літ.: [4], [5], [6]	2
	Разом за семестр:	34

Номер лекції	Перелік тем лекцій, їх анотації	Кількість годин
	2-ий семестр	
	Тема 1. Робота з інформаційними джерелами при вирішенні практичних задач в галузі комп'ютерних наук	2
1	Підходи до пошуку, оброблення та аналізу інформації з різних джерел для прийняття обґрунтованих рішень при розв'язанні задач у галузі комп'ютерних наук. Вимоги до формулювання постановки завдання. Оформлення результатів вишукувань у текстових редакторах. Приклади оформлення аналізу інформаційних джерел та постановки завдання. Літ.: [8] с.45-150, [9] с. 86-90	2
	Тема 2. Робота з інформаційними джерелами для вибору засобів розробки програмного забезпечення	2
2	Особливості пошуку, оброблення та аналізу інформації з різних джерел для прийняття обґрунтованих рішень щодо вибору засобів розробки програмного забезпечення. Критерії вибору зручної парадигми програмування за поставленого завдання. Документування процесу та результатів проєктування програмного забезпечення. Приклади документування результатів аналізу засобів розробки програмного забезпечення. Літ.: [9] с. 15-23	2
	Тема 3. Етапи розроблення та впровадження програмного забезпечення	2

3	Основні етапи розроблення програмного забезпечення. Етапи проєктування та розробки. Використання ескізів для проєктування інтерфейсу програмного забезпечення. Розроблення алгоритму методом покрокового уточнення. Вимоги до програмного коду. Поняття рефакторингу коду. Інструменти для рефакторингу коду. Літ.: [9] с. 24-50, [10] с. 1-65	2
	Тема 4. Основи тестування програмного забезпечення.	2
4	Проблема оцінювання якості створеного програмного забезпечення. Поняття тестування програмного забезпечення. Роль тестування в процесі розробки програмного забезпечення. Види тестування. Функціональне тестування. Визначення вимог до використання програмного забезпечення. Оцінювання відповідності програмного забезпечення вимогам та самокритичність. Приклади документування результатів функціонального тестування програмного забезпечення. Літ.: [11] с. 24-59, [12] с. 88-108	2
	Тема 5. Використання структур даних при розробці програмного забезпечення.	2
5	Колекції. Список (клас List). Масиви (клас ArrayList). Стек (клас Stack). Черга (клас Queue). Хештаблиці (клас Hashtable). Літ.: [1] с. 91-95, [2] с. 192-207	2
	Тема 6. Основи багатопоточності.	2
6	Однопоточність та багатопоточність. Багатозадачність з орієнтацією на процеси і потоки. Процес. Поток. Синхронізація. Клас Thread. Використання основного потоку. Стан потоку. Створення декількох потоків. Пріоритети потоків. Блокування. Взаємне блокування. Клас Monitor. Приклади реалізації багатопоточності. Літ.: [1] с. 96-100, [2] с. 288-308	2
	Тема 7. Основи Windows Presentation Foundation.	2
7	Огляд технології WPF. Структура WPF-застосунків. Мова розмітки XAML. Компоновка та побудова WPF-застосунків. Найпростіший WPF-застосунок. Літ.: [2] с. 237-246	2
	Тема 8. Засоби Windows Presentation Foundation.	2
8	Клас Application. Базові елементи управління. Макет вікна. Панелі. Події. Стили і шаблони. Прив'язка даних. Шаблон MVVM (Model-View-ViewModel). Літ.: [2] с. 247-257	2
	Разом за семестр:	18

5.2 Зміст лабораторних занять

№ п/п	Тема лабораторного заняття	Кількість годин
1-й семестр		
1	Проєктування алгоритмів вирішення задач. Ознайомлення з програмними засобами побудови блок-схем та текстовими редакторами. Розробка, проєктування алгоритмів вирішення задач відповідно до завдання. Подання алгоритмів у вигляді опису, псевдокоду, блок-схеми. Оформлення звіту з виконання завдання. Літ.: [1] с. 10-19, [1], с. 20-25, [2] с. 28-39, [3] с. 1-9	8
2	Створення простої програми на мові C#. Ознайомлення з середовищем програмування Visual Studio та мовою програмування C#. Створення програми на мові C#, компіляція програмного коду. Оформлення звіту з виконання поставленого завдання. Літ.: [1] с. 4-9, [2] с. 7-27	8
3	Робота з типами даних, виразами, операціями та математичними функціями на мові C#. Використання виразів і операцій при написанні програмного коду. Робота з типами даних C#. Використання методів бібліотек стандартних математичних функцій. Створення програм з використанням стандартних математичних функцій відповідно до завдання. Оформлення звіту з виконання поставленого завдання. Літ.: [1] с. 26-31, [2] с. 40-57	8
4	Робота з операторами розгалуження, вибору та циклу в програмних реалізаціях на мові C#. Проєктування алгоритмів із використанням розгалуження й вибору, циклів. Використання операторів розгалуження й вибору, циклів при написанні програмного коду. Створення програм відповідно до завдання. Оформлення звіту з виконання поставленого завдання. Літ.: [1] с. 32-47, [2] с. 60-71, [3] с. 48-67	8

5	Робота з одновимірними масивами. Програмна реалізація одновимірних масивів, алгоритми пошуку мінімальних та максимальних значень, сортування в одновимірних масивах. Оформлення звіту з виконання поставленого завдання. Літ.: [1] с.48-55, [2] с. 65-168, [3] с. 77-96	8
6	Робота з багатовимірними масивами. Програмна реалізація двовимірних масивів, алгоритми пошуку мінімальних та максимальних значень, сортування в двовимірних масивах. Оформлення звіту з виконання поставленого завдання.	8
7	Робота з символьним та рядковим типом даних Програмна реалізація алгоритмів пошуку та сортування символьних величин і рядків. Програмна реалізація методів сортування та пошуку в масивах та рядках за допомогою вбудованих класів Array та StringBuilder. Оформлення звіту з виконання поставленого завдання. Літ.: [1] с.63-64, [2] с. 42-44, [2] с. 176-178	8
8	Розробка програм з використанням Windows Forms. Створення Windows Forms додатка відповідно до завдання. Зчитування інформації з текстових полів, програмування кнопок, робота з компонентом Chart. Оформлення звіту з виконання поставленого завдання. Літ.: [1] с. 101-105, [2] с. 208-236	10
Разом за семестр:		66

№ п/п	Тема лабораторного заняття	Кількість годин
2-й семестр		
1	Робота з текстовими та спеціалізованими файлами Використання Visual Studio C# для читання та запису інформації в текстовий файл. Програмне створення спеціалізованих файлів та наповнення їх інформацією. Робота з компонентом dataGridView у Windows Forms. Оформлення звіту з виконання поставленого завдання. Літ.: [1] с.75-79, [3] с. 253-259, [5], [6], [7]	2
2	Пошук та аналіз інформаційних джерел для вирішення практичної задачі в галузі комп'ютерних наук Проведення дослідження предметної області згідно отриманої від викладача теми. Пошук, оброблення та аналіз інформації з різних джерел для прийняття обґрунтованих рішень щодо розв'язання задачі у галузі комп'ютерних наук. Оформлення результатів вишукувань і постановка завдання. Літ.: [8] с.45-150, [9] с. 86-90	2
3	Проектування програмного продукту для вирішення практичної задачі. Розробка, проектування алгоритмів для вирішення прикладної задачі відповідно до поставленого завдання. Вибір парадигми програмування та засобів розробки програмного продукту. Оформлення результатів проектування програмного продукту. Літ.: [2] с. 20-28, [9] с. 15-23	4
4	Створення програмного продукту для вирішення практичної задачі згідно проекту Проектування інтерфейсу програмного продукту згідно проекту. Створення компонентів інтерфейсу. Програмування подій для взаємодії користувача з компонентами інтерфейсу. Літ.: [9] с. 24-50, [10] с. 1-65	2
5	Командна робота з тестування. Документування результатів розроблення програмного продукту Візуальне подання програмної архітектури. Опис програмних складових. Функціональне тестування програмного продукту та визначення вимог до використання. Підсумок результатів розроблення програмного продукту. Літ.: [2] с. 11-21, [11] с. 24-59, [12] с. 88-108	2
6	Робота зі структурами даних. Програмна реалізація простого списку List, стеку та хеш-таблиці відповідно до завдання. Оформлення звіту з виконання поставленого завдання. Літ.: [1] с. 91-95, [3] с. 192-207	2
7	Робота з потоками. Програмна реалізація читання та запису текстової інформації в файл за допомогою багатопоточності відповідно до завдання. Оформлення звіту з виконання поставленого завдання. Літ.: [1] с. 96-100, [3] с. 266-287	2
8	Створення WPF-застосунка. Створення WPF-застосунка. Створення програми відповідно до завдання. Оформлення звіту з виконання поставленого завдання. Літ.: [3] с. 237-257	2
Разом за семестр:		18

5.3 Зміст самостійної (у т.ч. індивідуальної) роботи здобувача вищої освіти

Самостійна робота студентів усіх форм здобуття освіти полягає у систематичному опрацюванні програмного матеріалу з відповідних джерел інформації, підготовці до лабораторних занять, контрольних робіт, виконанні індивідуальних завдань. Крім цього до послуг студентів сторінка навчальної дисципліни у Модульному середовищі для навчання, де розміщені Робоча програма дисципліни та необхідні документи з її навчально-методичного забезпечення.

Номер тижня	Вид самостійної роботи <i>1-й семестр</i>	Кількість годин
1	Опрацювання теоретичного матеріалу з лекції 1, підготовка до виконання лабораторної роботи №1	8
2	Опрацювання теоретичного матеріалу з лекції 2, підготовка до захисту лабораторної роботи №1.	8
3	Опрацювання теоретичного матеріалу з лекції 3, підготовка до виконання лабораторної роботи №2	8
4	Опрацювання теоретичного матеріалу з лекції 4, підготовка до захисту лабораторної роботи №2	8
5	Опрацювання теоретичного матеріалу з лекції 5, підготовка до виконання лабораторної роботи №3	8
6	Опрацювання теоретичного матеріалу з лекції 6, підготовка до захисту лабораторної роботи №3	8
7	Опрацювання теоретичного матеріалу з лекції 7, підготовка до лабораторної роботи №4, Підготовка до КР №1	12
8	Опрацювання теоретичного матеріалу з лекції 8, підготовка до захисту лабораторної роботи №3	8
9	Опрацювання теоретичного матеріалу з лекції 9, підготовка до виконання лабораторної роботи №4, початок виконання ІЗ (завдання 1)	12
10	Опрацювання теоретичного матеріалу з лекції 10, підготовка до захисту лабораторної роботи №4, виконання ІЗ (завдання 1)	12
11	Опрацювання теоретичного матеріалу з лекції 11, підготовка до виконання лабораторної роботи №5, виконання ІЗ (завдання 2)	12
12	Опрацювання теоретичного матеріалу з лекції 12, підготовка до захисту лабораторної роботи №5, початок виконання ІЗ (завдання 2)	12
13	Опрацювання теоретичного матеріалу з лекції 13, підготовка до виконання лабораторної роботи №6, виконання ІЗ (завдання 2)	12
14	Опрацювання теоретичного матеріалу з лекції 14, підготовка до захисту лабораторної роботи №6, виконання ІЗ (завдання 2). Підготовка до КР №2	14
15	Опрацювання теоретичного матеріалу з лекції 15, підготовка до виконання лабораторної роботи №7 та №8, виконання ІЗ (завдання 2). Підготовка до іспиту.	20
16	Опрацювання теоретичного матеріалу з лекції 16, підготовка до захисту лабораторних робіт №7 та №8, оформлення звіту по ІЗ (завдання 1-2). Підготовка до іспиту.	20
17	Опрацювання теоретичного матеріалу з лекції 16, підготовка до здачі ІЗ (завдання 1-2). Підготовка до іспиту.	20
	Разом:	202

Номер тижня	Вид самостійної роботи <i>2-й семестр</i>	Кількість годин
1-2	Опрацювання теоретичного матеріалу з лекції 1, підготовка до виконання та захисту лабораторної роботи №1, початок виконання ІЗ (завдання 1)	4
3-4	Опрацювання теоретичного матеріалу з лекції 2, підготовка до виконання та захисту лабораторної роботи №2 виконання ІЗ (завдання 1)	4
5-6	Опрацювання теоретичного матеріалу з лекції 3, підготовка до виконання та захисту лабораторної роботи №3, Підготовка до КР №1	6
7-8	Опрацювання теоретичного матеріалу з лекції 4. Підготовка до лабораторної роботи. Підготовка до захисту лабораторної роботи. Виконання курсового проєкту. Підготовка до проміжного контрольного заходу.	8
9-10	Опрацювання теоретичного матеріалу з лекції 5, підготовка до захисту лабораторної роботи №6, виконання ІЗ (завдання 2). Виконання курсового проєкту. Підготовка до КР №2	6
11-12	Опрацювання теоретичного матеріалу з лекції 6, підготовка до виконання та захисту лабораторної роботи №6, виконання ІЗ (завдання 1). Виконання курсового проєкту. Підготовка до КР №2	6
13-14	Опрацювання лекційного матеріалу з лекції 7, Підготовка до лабораторної роботи. Підготовка до виконання та захисту лабораторної роботи, виконання ІЗ (завдання 1).	8

15-16	Опрацювання теоретичного матеріалу з лекції 8, підготовка до виконання та захисту лабораторної роботи №6, виконання ІЗ (завдання 2). Підготовка звіту з ІЗ. Підготовка до КР №2. Підготовка до захисту курсового проєкту.	10
17	Захист курсового проєкту. Захист ІЗ.	4
	Разом годин:	56

Примітки: КР – контрольна робота, ІЗ – індивідуальне завдання.

На самостійне опрацювання студентів виносяться визначені у методичних рекомендаціях до лабораторних занять та самостійної роботи питання з відповідних тем. Керівництво самостійною роботою та контроль за виконанням індивідуального завдання здійснюється викладачем згідно з розкладом консультацій у позаурочний час.

Вимоги до виконання індивідуального домашнього завдання викладені в Модульному середовищі для навчання на сторінці навчальної дисципліни.

5.4 Курсове проєктування

Індивідуальна робота студентів у другому семестрі відповідно до навчального плану передбачає виконання курсового проєкту.

Метою курсового проєктування з дисципліни є закріплення, поглиблення та узагальнення здобувачами одержаних під час навчання знань і практичних навичок з розроблення алгоритмів та використання мов програмування, та їх застосування до комплексного вирішення конкретних спеціалізованих завдань та практичних проблем у галузі комп'ютерних наук і в процесі навчання.

На початку семестру керівник курсового проєкту проводить загальну консультацію, на якій висвітлюються мета та завдання курсового проєктування, етапи і терміни його виконання, календарний план, напрямки тематики, форми та критерії оцінювання. Розглядаються загальні вимоги щодо написання й оформлення курсового проєкту, вимоги до розробки програмного застосунку, розклад індивідуальних консультацій, список рекомендованої навчальної й довідкової літератури для виконання завдань курсового проєкту. Оголошується перелік запропонованих тем курсових проєктів.

Після чого, згідно з розкладом індивідуальних консультацій, із здобувачами проводиться обговорення, коригування, уточнення або заміна тем курсових проєктів, формується титульний аркуш. Завдання на курсовий проєкт відповідним чином коригується викладачем, й не пізніше четвертого тижня семестру видається здобувачу, підписується здобувачем та керівником. Завдання на курсовий проєкт містить Календарний план виконання у вигляді таблиці, де зазначені назви розділів (етапів) та терміни їх виконання.

Після одержання завдання і ознайомлення зі змістом та обсягом роботи із курсового проєкту, здобувач за погодженням із керівником проєкту формулює орієнтовні назви розділів та пунктів пояснювальної записки курсового проєкту. У подальшому, здобувач згідно з календарним планом поетапно розробляє та оформляє пояснювальну записку і презентаційний матеріал, створює програмний застосунок і своєчасно звітує перед керівником курсового проєкту про виконання кожного етапу відповідно до графіка. Звітність здобувачів керівникові курсового проєкту стосовно ходу його виконання є обов'язковою.

Пояснювальна записка є текстовим документом, що містить аналіз, обґрунтування й опис прийнятих у курсовому проєкті рішень щодо розробки програмного застосунку, а також документування процесу програмного проєктування та реалізації програмного застосунку. Всі розділи пояснювальної записки повинні бути логічно взаємопов'язані і переконливо аргументовані. Матеріал її має бути цілком присвячений темі роботи, досягненню мети, вирішенню поставлених завдань. Неприпустимі будь-які відступи, що не мають відношення до завдання. Рекомендована кількість сторінок пояснювальної записки курсового проєкту з дисципліни «Алгоритмізація та програмування» (без завдання, анотації та додатків) 30–35 с. Максимальна сукупна кількість сторінок (із додатками) – не більше 100 с.

Консультації з курсового проєкту здійснює керівник проєкту – викладач кафедри відповідно до графіку, затвердженого деканом факультету. Захист курсового проєкту здійснюється після його перевірки керівником перед комісією, яка складається з 2–3 викладачів кафедри. До складу комісії обов'язково входить керівник проєкту.

Приклади тем індивідуального завдання для Курсового проєкту

- Застосунок для вивчення синтаксису мови C# для початківців з використанням інтервального повторення матеріалу на платформі .NET.
- Застосунок-менеджер для планування розпорядку дня мовою C#.
- Програмний застосунок на мові C# для визначення енергетичної цінності раціонів здорового харчування.
- Застосунок для перевірки рівня знань з української мови на платформі .NET.
- Програмне забезпечення на платформі .NET для оцінювання ігрового потенціалу сформованих футбольних команд за характеристиками гравців.
- Застосунок для визначення якості сну за опитуванням користувача на платформі .NET.
- Застосунок для обліку витрачених калорій за виконаними видами спортивної діяльності з метою індивідуального регулювання навантаження засобами .NET.
- Застосунок-тренажер для покращення набору тексту методом сліпого друку засобами мови програмування C#.
- Застосунок для планування виду та обсягу тренувань в спортзалі згідно цілей користувача мовою C#.
- Програмний застосунок для підбору комплектуючих персональних комп'ютерів за параметрами користувача на платформі .NET.

- Програмний продукт для визначення рівня емпатії шляхом тестування психологічних властивостей та уподобань мовою C#.
- Програмний застосунок для відтворення аудіофайлів з елементами візуалізації та персоналізованих налаштувань на платформі .NET.
- Програмний застосунок-щоденник для планування особистого навчального процесу студентами вищих навчальних закладів мовою C#.
- Комп'ютерна програма для формування білетів з підсумкових заходів дисциплін на платформі .NET.
- Графічний редактор для створення двовимірних зображень засобами .NET.
- Програма на платформі .NET для автоматизованого обрахунку потреб замовлень продуктового магазину.
- Застосунок для читання електронних книг з можливістю персоналізованих налаштувань засобами .NET.
- Програма-калькулятор на мові C# для розрахунку вартості комунальних послуг у багатоквартирних будинках міста Хмельницький.

Критерії оцінювання курсового проєкту

Оцінювання курсового проєкту здійснюється за інституційною чотирибальною шкалою «відмінно», «добре», «задовільно», «незадовільно» і шкалою ЄКТС.

Критерії оцінювання курсового проєкту:

- відмінно – розроблений програмний застосунок виконує всі необхідні функції, що були поставлені у завданні, використано сучасні засоби розробки, пояснювальна записка повністю відповідає вимогам оформлення та розкриває всі положення курсового проєкту, на всі запитання на захисті дано вичерпні й коректні відповіді;
- добре – розроблений програмний застосунок виконує основні необхідні функції, що були поставлені у завданні, використано сучасні засоби розробки, пояснювальна записка у достатній мірі відповідає вимогам оформлення і розкриває ключові положення проєкту, на всі запитання на захисті дано коректні відповіді;
- задовільно – розроблений програмний застосунок виконує більшість необхідних функцій, що були поставлені у завданні, пояснювальна записка переважно відповідає вимогам оформлення й розкриває більшість положень курсового проєкту, на більшість запитань на захисті дано коректні відповіді;
- незадовільно – розроблений програмний застосунок виконує недостатню кількість функцій, що були поставлені у завданні, або не в повній мірі відповідає темі курсового проєкту, пояснювальна записка не відповідає вимогам оформлення або недостатньо розкриває положення курсового проєкту, на більшість запитань на захисті дано некоректні відповіді.

При оцінюванні комісія враховує фахову підготовку й теоретичні знання здобувача в межах дисципліни, вільне володіння фаховими термінами та поняттями, правильність і точність відповідей на питання, оригінальність та правильність вирішення поставлених завдань, дотримання графіку виконання та звітності, ступінь самостійності роботи здобувача, правильність вибору методів вирішення завдань та якість документування одержаних результатів у пояснювальній записці. Оцінюванню підлягає виключно особистий внесок розробника, тому здобувачу слід звернути особливу увагу та чітко вказати, де особисті наробки, а де використано загальнодоступні елементи та компоненти сторонніх розробників, як то програмні бібліотеки, автоматизовані засоби генерації програмних компонентів, існуючі математичні моделі та алгоритми тощо.

Структурування оцінювання курсового проєкту за компонентами та ваговими коефіцієнтами

№	Компоненти оцінювання	Коеф.
1	Документування аналізу предметної області, існуючих технологій та аналогів. Постановка задачі, визначення вимог до системи.	0,1
2	Документування розробки блок-схем функцій програмного застосунку та UML-моделювання	0,2
3	Документування розробки комбінації засобів розробки, програмної реалізації застосунку та аналіз його функціональності	0,2
4	Якість оформлення пояснювальної записки (нормоконтроль)	0,1
5	Програмний застосунок – функціонал, коректність результатів, складність реалізації, використані технології.	0,2
6	Захист курсового проєкту (доповідь, відповіді на питання)	0,2

6. Технології та методи навчання

Процес навчання з дисципліни ґрунтується на використанні традиційних та сучасних технологій, зокрема: лекції (з використанням методів візуалізації); лабораторні заняття (з використанням тренінгів, майстер-класів), самостійна робота (індивідуальні завдання).

Необхідні інструменти, обладнання, програмне забезпечення: комп'ютер (надається для використання в лабораторіях кафедри КН), Visual Studio (ліцензія ХНУ), Dia, Flying Logic, Edraw MAX, Algorithm Flowcharts Editor (виробниками дозволяється безкоштовне некомерційне використання), текстовий редактор (пропонується використання безкоштовних онлайн-сервісів).

7. Методи контролю

Поточний контроль здійснюється під час аудиторних лабораторних занять, а також у дні проведення

контрольних заходів, встановлених робочою програмою і графіком освітнього процесу, в т.ч. з використанням Модульного середовища для навчання. При цьому використовуються такі методи поточного контролю:

- оцінювання результатів виконання індивідуального завдання;
- оцінювання контрольних робіт;
- оцінювання результатів захисту лабораторних робіт.

При виведенні підсумкової семестрової оцінки враховуються результати як поточного контролю, так і підсумкового контролю, який проводиться з усього матеріалу дисципліни за білетами, попередньо розробленими і затвердженими на засіданні кафедри. Здобувач вищої освіти, який набрав з будь-якого виду навчальної роботи, суму балів нижчу за 60 відсотків від максимального балу, **не допускається** до семестрового контролю, поки не виконає обсяг роботи, передбачений Робочою програмою. Здобувач вищої освіти, який набрав позитивний середньозважений бал (60 відсотків і більше від максимального балу) з усіх видів поточного контролю і не склав іспит, вважається таким, який **має** академічну заборгованість. Ліквідація академічної заборгованості із семестрового контролю здійснюється у період екзаменаційної сесії або за графіком, встановленим деканатом відповідно до «Положення про контроль і оцінювання результатів навчання здобувачів вищої освіти у ХНУ».

8. Політика дисципліни

Політика навчальної дисципліни загалом визначається системою вимог до здобувача вищої освіти, що передбачені чинними положеннями Університету про організацію і навчально-методичне забезпечення освітнього процесу. Зокрема, проходження інструктажу з техніки безпеки; відвідування занять з дисципліни є обов'язковим. За об'єктивних причин (підтверджених документально) теоретичне навчання за погодженням із лектором може відбуватись в он-лайн режимі. Успішне опанування дисципліни і формування фахових компетентностей і програмних результатів навчання передбачає необхідність підготовки до лабораторного заняття (вивчення теоретичного матеріалу з теми роботи, попередню підготовку протоколу роботи (наведені у Методичних рекомендаціях до лабораторних занять)), активно працювати на занятті, якісно підготувати звіт, захистити результати виконаної роботи, брати участь у дискусіях щодо прийнятих конструктивних рішень при виконанні здобувачами лабораторних робіт тощо.

Здобувачі вищої освіти мають дотримуватися встановлених термінів виконання всіх видів навчальної роботи відповідно до робочої програми навчальної дисципліни. Термін захисту лабораторної роботи вважається своєчасним, якщо студент захистив її на наступному після виконання роботи занятті. Пропущене лабораторне заняття студент зобов'язаний відпрацювати у встановлений викладачем термін, але не пізніше, ніж за два тижні до кінця теоретичних занять у семестрі.

Засвоєння студентом теоретичного матеріалу з дисципліни оцінюється за результатами опитування під час лабораторних занять, виконання індивідуального завдання та контрольних робіт. Виконання індивідуального завдання завершується його здачею на перевірку у терміни, встановлені графіком самостійної роботи.

Здобувач вищої освіти, виконуючи самостійну роботу або індивідуальну роботу з дисципліни, має дотримуватися політики доброчесності (заборонені списування, плагіат (в т.ч. із використанням мобільних девайсів)). У разі виявлення порушення політики академічної доброчесності в будь-яких видах навчальної роботи здобувач вищої освіти отримує незадовільну оцінку і має повторно виконати завдання з відповідної теми (виду роботи), що передбачені робочою програмою. Будь-які форми порушення академічної доброчесності **не допускаються**.

У межах вивчення навчальної дисципліни допускається визнання та зарахування результатів навчання, здобутих шляхом неформальної освіти на відкритих освітніх платформах. Такі результати можуть враховуватися за умови їх відповідності компетентностям і програмним результатам навчання, визначеним робочою програмою дисципліни, або у разі забезпечення вивчення окремих тем і видів робіт, що передбачені програмою. (детальніше у Положенні про порядок визнання та зарахування результатів навчання здобувачів вищої освіти у ХНУ).

Окремі результати вивчення курсу можуть бути зараховані у випадку отримання студентом результатів навчання у неформальній освіті, що підтверджені відповідним документом (сертифікат, свідоцтво, освітня програма тощо):

- Як результат виконання ЛР №1, 3, 4, зараховується онлайн курс <https://www.coursera.org/learn/c-sharp-for-dot-net?action=enroll#testimonials>
- Як результат виконання ЛР №1, 3, 4, 7, зараховується онлайн курс <https://www.freecodecamp.org/learn/foundational-c-sharp-with-microsoft/foundational-c-sharp-with-microsoft-certification-exam/foundational-c-sharp-with-microsoft-certification-exam>

9. Оцінювання результатів навчання студентів у семестрі

Оцінювання академічних досягнень здобувача вищої освіти здійснюється відповідно до «Положення про контроль і оцінювання результатів навчання здобувачів вищої освіти у ХНУ». При поточному оцінюванні виконаної здобувачем роботи з кожної структурної одиниці і отриманих ним результатів викладач виставляє йому певну кількість балів із встановлених Робочою програмою для цього виду роботи. При цьому кожна структурна одиниця навчальної роботи може бути зарахована, якщо здобувач набрав не менше 60 відсотків (мінімальний рівень для позитивної оцінки) від максимально можливої суми балів, призначеної структурній одиниці.

При оцінюванні результатів навчання здобувачів вищої освіти з будь-якого виду навчальної роботи (структурної одиниці) рекомендується використовувати наведені нижче узагальнені критерії:

Таблиця – Критерії оцінювання навчальних досягнень здобувача вищої освіти

Оцінка та рівень досягнення здобувачем запланованих ПРН та сформованих компетентностей	Узагальнений зміст критерія оцінювання
Відмінно (високий)	Здобувач вищої освіти глибоко і у повному обсязі опанував зміст навчального матеріалу, легко в ньому орієнтується і вміло використовує понятійний апарат; уміє пов'язувати теорію з практикою, вирішувати практичні завдання, впевнено висловлювати і обґрунтовувати свої судження. Відмінна оцінка передбачає логічний виклад відповіді мовою викладання (в усній або у письмовій формі), демонструє якісне оформлення роботи і володіння спеціальними приладами та інструментами, прикладними програмами. Здобувач не вагається при видозміні запитання, вміє робити детальні та узагальнюючі висновки, демонструє практичні навички з вирішення фахових завдань. При відповіді допустив дві–три несуттєві <i>похибки</i> .
Добре (середній)	Здобувач вищої освіти виявив повне засвоєння навчального матеріалу, володіє понятійним апаратом, орієнтується у вивченому матеріалі; свідомо використовує теоретичні знання для вирішення практичних задач; виклад відповіді грамотний, але у змісті і формі відповіді можуть мати місце окремі неточності, нечіткі формулювання правил, закономірностей тощо. Відповідь здобувача вищої освіти будується на основі самостійного мислення. Здобувач вищої освіти у відповіді допустив дві–три <i>несуттєві помилки</i> .
Задовільно (достатній)	Здобувач вищої освіти виявив знання основного програмного матеріалу в обсязі, необхідному для подальшого навчання та практичної діяльності за професією, справляється з виконанням практичних завдань, передбачених програмою. Як правило, відповідь здобувача вищої освіти будується на рівні репродуктивного мислення, здобувач вищої освіти має слабкі знання структури навчальної дисципліни, допускає неточності і <i>суттєві помилки</i> у відповіді, вагається при відповіді на видозмінене запитання. Разом з тим, набув навичок, необхідних для виконання нескладних практичних завдань, які відповідають мінімальним критеріям оцінювання і володіє знаннями, що дозволяють йому під керівництвом викладача усунути неточності у відповіді.
Незадовільно (недостатній)	Здобувач вищої освіти виявив розрізнені, безсистемні знання, не вміє виділяти головне і другорядне, допускається помилок у визначенні понять, перекручує їх зміст, хаотично і невпевнено викладає матеріал, не може використовувати знання при вирішенні практичних завдань. Як правило, оцінка «незадовільно» виставляється здобувачеві вищої освіти, який не може продовжити навчання без додаткової роботи з вивчення навчальної дисципліни.

Структурування дисципліни за видами навчальної роботи і оцінювання результатів навчання студентів денної форми здобуття освіти у 1 семестрі

Аудиторна робота								Контрольні заходи		Самостійна робота		Семестровий контроль	Разом
Лабораторні заняття №:								Контрольна робота (КР)		Індивідуальне завдання (ІЗ)		Іспит	Сума балів
1	2	3	4	5	6	7	8	КР1	КР2	ІЗ1	ІЗ2		
Кількість балів за кожний вид навчальної роботи (мінімум-максимум)													
3-5	3-5	3-5	3-5	3-5	3-5	3-5	3-5	3-5	3-5	3-5	3-5	24-40	
24-40								6-10		6-10		24-40	60-100*

Структурування дисципліни за видами навчальної роботи і оцінювання результатів навчання студентів денної форми здобуття освіти у 2 семестрі

Аудиторна робота								Контрольні заходи		Самостійна робота		Семестровий контроль
Лабораторні заняття №:								Контрольна робота (КР)		Індивідуальне завдання (ІЗ)		Залік
1	2	3	4	5	6	7	8	КР1	КР2	ІЗ1	ІЗ2	
Кількість балів за кожний вид навчальної роботи (мінімум-максимум)												
6-10	6-10	6-10	6-10	6-10	6-10	6-10	6-10	3-5	3-5	3-5	3-5	За рейтингом
48-80								6-10		6-10		60-100*

Примітки: *За набрану з будь-якого виду навчальної роботи з дисципліни кількість балів, нижче встановленого мінімуму, здобувач отримує незадовільну оцінку і має її перездати у встановлений викладачем (деканом) термін. Інституційна оцінка встановлюється відповідно до таблиці «Співвідношення інституційної шкали оцінювання і шкали оцінювання ЄКТС».

Оцінювання результатів захисту лабораторної роботи

Лабораторна робота, що виконана й оформлена студентом, відповідно до вимог, які визначені викладачем на початку навчального семестру, комплексно оцінюється викладачем під час її захисту з урахуванням таких критеріїв: самостійність та правильність програмної реалізації завдань; повнота відповіді та знання теоретичних основ (зокрема роботи з типами даних, операторами розгалуження та циклами, використання масивів і рядкових типів даних, а також базових понять методів і функцій); якість програмного коду, його структура, дотримання принципів слабкої зв'язності та високої згуртованості.

Результат виконання і захисту здобувачем вищої освіти кожної лабораторної роботи оцінюється, відповідно до таблиці «**Критерії оцінювання навчальних досягнень здобувача вищої освіти**» (мінімальний позитивний бал – 3 бали, максимальний – 5 балів для першого семестру та мінімальний позитивний бал – 6 бали, максимальний – 10 балів для другого семестру).

У випадку виявлення здобувачем рівня знань, нижчого ніж 60 відсотків від максимального балу, становленого Робочою програмою для кожної структурної одиниці, лабораторна робота йому не зараховується і для її захисту він має детальніше опрацювати матеріал з теми роботи, методика її виконання, виправити грубі помилки та повторно вийти на її захист у призначений для цього викладачем час.

Оцінювання контрольної роботи

Контрольна робота передбачає виконання п'яти завдань. При оцінюванні контрольної роботи враховуються: повнота відповіді та якість виконання. Кожне завдання оцінюється 1 балом, загальна сума балів на позитивну оцінку становить від 3 до 5.

Розподіл балів при оцінюванні завдань контрольної роботи

Кількість правильних відповідей	1	2	3	4	5
Відсоток правильних відповідей	0-59		60	80	100
Кількість отриманих балів	0		3	4	5

При отриманні негативної оцінки контрольну роботу слід перездати до терміну *наступного* контролю.

Оцінювання результатів виконання індивідуального завдання

Виконане та оформлене відповідно до вимог, визначених Методичними рекомендаціями, індивідуальне завдання (ІЗ) комплексно оцінюється викладачем з урахуванням таких критеріїв: самостійність виконання; правильність розв'язання поставлених задач; обґрунтованість вибору методів розв'язання; повнота пояснень та аргументованість відповідей; якість оформлення та дотримання вимог до структури і змісту роботи.

Результат виконання здобувачем вищої освіти кожного ІЗ оцінюється відповідно до таблиці **Критеріїв оцінювання навчальних досягнень здобувача вищої освіти** з урахуванням рівня досягнення запланованих програмних результатів навчання та сформованих компетентностей. За підсумками захисту присвоюється відповідна сума балів (мінімальний позитивний бал – 3 бали, максимальний – 5 балів).

У разі, якщо здобувач вищої освіти виявив рівень знань і виконання ІЗ, що нижчий ніж 60 відсотків від максимальної кількості балів, встановленої Робочою програмою для цієї структурної одиниці, завдання не зараховується. У такому випадку студент має повторно опрацювати зміст завдання, усунути помилки та здати на перевірку доопрацьоване ІЗ у терміни, погоджені з викладачем.

Оцінювання результатів підсумкового семестрового контролю (іспит)

Освітня програма передбачає підсумковий семестровий контроль з дисципліни у формі іспиту, завданням якого є системне й об'єктивне оцінювання як теоретичної, так і практичної підготовки здобувача з навчальної дисципліни. Складання іспиту відбувається за попередньо розробленими і затвердженими на засіданні кафедри білетами. Відповідно до цього в екзаменаційному білеті пропонується поєднання питань як теоретичного, так і практичного характеру.

Таблиця – Оцінювання результатів підсумкового семестрового контролю здобувачів денної форми навчання (40 балів для підсумкового контролю)

Види завдань	Для кожного окремого виду завдань		
	Мінімальний (достатній) бал (задовільно)	Потенційні позитивні бали* (середній бал) (добре)	Максимальний (високий) бал (відмінно)
Завдання № 1	6	8	10
Завдання № 2	6	8	10
Завдання № 3	6	8	10
Завдання № 4	6	8	10
Разом:	24		40

Примітка. *Позитивний бал за іспит, відмінний від мінімального (24 балів) та максимального (40 балів), знаходиться в межах 25-39 балів та розраховується як сума балів за усі структурні елементи (завдання) іспиту.

Для кожного окремого виду завдань підсумкового семестрового контролю застосовуються критерії оцінювання навчальних досягнень здобувача вищої освіти, наведені вище (**Таблиця – Критерії оцінювання навчальних досягнень здобувача вищої освіти**).

Підсумкова семестрова оцінка за інституційною шкалою і шкалою ЄКТС визначається в автоматизованому

режимі після внесення викладачем результатів оцінювання у балах з усіх видів навчальної роботи до електронного журналу. Співвідношення інституційної шкали оцінювання і шкали оцінювання ЄКТС наведені нижче у таблиці «Співвідношення».

Семестровий іспит виставляється, якщо загальна сума балів, яку набрав студент з дисципліни за результатами поточного контролю, знаходиться у межах від 60 до 100 балів. При цьому за інституційною шкалою ставиться оцінка «відмінно/добре/задовільно», а за шкалою ЄКТС – буквене позначення оцінки, що відповідає набраній студентом сумі балів відповідно до таблиці Співвідношення.

Таблиця – Співвідношення інституційної шкали оцінювання і шкали оцінювання ЄКТС

Оцінка ЄКТС	Рейтингова шкала балів	Інституційна оцінка (рівень досягнення здобувачем вищої освіти запланованих результатів навчання з навчальної дисципліни)	
		Залік	Іспит/диференційований залік
A	90-100	Зараховано	Відмінно/Excellent – високий рівень досягнення запланованих результатів навчання з навчальної дисципліни, що свідчить про безумовну готовність здобувача до подальшого навчання та/або професійної діяльності за фахом
B	83-89		Добре/Good – середній (максимально достатній) рівень досягнення запланованих результатів навчання з навчальної дисципліни та готовності до подальшого навчання та/або професійної діяльності за фахом
C	73-82		
D	66-72		Задовільно/Satisfactory – Наявні мінімально достатні для подальшого навчання та/або професійної діяльності за фахом результати навчання з навчальної дисципліни
E	60-65		
FX	40-59	Незараховано	Незадовільно/Fail – Низка запланованих результатів навчання з навчальної дисципліни відсутня. Рівень набутих результатів навчання є недостатнім для подальшого навчання та/або професійної діяльності за фахом
F	0-39		Незадовільно/Fail – Результати навчання відсутні

10. Питання для самоконтролю результатів навчання

1. Яким чином можна створити консольний додаток на мові C#?
2. Які парадигми програмування ви знаєте? Розкрийте коротко їх суть.
3. Що таке .NET Framework?
4. Які компоненти містить .NET Framework, що визначають його структуру?
5. Що таке транслятор, компілятор та інтерпретатор?
6. Що таке алгоритм?
7. Які властивості алгоритмів ви знаєте?
8. Які типи алгоритмів ви знаєте? Дайте їм визначення.
9. Що таке складність алгоритму? Які складності ви знаєте?
10. Які існують способи подання алгоритмів? Наведіть приклади.
11. Як позначаються основні блоки і що вони значать в блок-схемах?
12. Що таке мова програмування? Класифікації мов програмування.
13. З яких складових складається будь-яка мова програмування?
14. Що таке ключове слово? Наведіть приклад декількох ключових слів та що вони значать.
15. Що таке синтаксис мови програмування? Наведіть кілька правил синтаксису C#.
16. Що таке коментар? Які види коментарів ви знаєте?
17. Який метод в C# використовується для зчитування рядка символів, введеного з клавіатури в консольному вікні?
18. Яка різниця між методами Console.Write(), та Console.WriteLine ()?
19. Що таке змінна? Які види змінних існують у C#?
20. Які правила створення ідентифікатора змінної?
21. Що таке оголошення та ініціалізація змінної?
22. Що таке стек та купа? Яка між ними різниця?
23. Що таке константа? Характерні ознаки константи. Приклад константи.
24. Якими типами даних оперує мова C#?
25. В чому полягає різниця в оголошенні типу в C# і в CLR?
26. Які типи даних визначає користувач?
27. Як розміщуються дані у пам'яті під час роботи програми?
28. Яким чином здійснюється приведення типів змінних?
29. Основні логічні оператори.
30. Що таке унарний, бінарний та тернарний оператор? Наведіть приклади кожного з них.
31. Що таке інкремент та декремент?

32. Які методи класу Math ви знаєте? Для чого вони використовуються?
33. Що таке умовна конструкція? Які типи умовних конструкцій ви знаєте?
34. Наведіть синтаксис оператора вибору – if.
35. Наведіть синтаксис оператора вибору – if...else.
36. Наведіть синтаксис оператора вибору switch.
37. Які існують оператори для створення циклів?
38. Оператор циклу for. Призначення, синтаксис.
39. Основна відмінність між циклами While і do While.
40. Що таке лічильник? З яким кроком може змінюватись лічильник?
41. Перерахуйте оператори переходу.
42. Дайте означення масиву.
43. Які види масивів ви знаєте?
44. Скільки вимірів може мати масив?
45. Як оголошуються одновимірні масиви?
46. Що таке коваріантність масивів?
47. Яким чином можна заповнювати масиви?
48. Яким чином описуються багатовимірні масиви?
49. Що таке зубчатий масив?
50. Як створити та ініціалізувати двовимірний зубчатий масив?
51. Що таке головна діагональ в двовимірному масиві? В якому масиві вона існує?
52. Для чого використовують масиви масивів?
53. Що таке символна інформація? Який тип даних в .NET Framework представляє символи?
54. Які види символів може представляти тип даних char?
55. Наведіть декілька прикладів оголошення змінної типу char.
56. Що таке символ? Який тип даних використовується для представлення символів?
57. Які методи класу Char ви знаєте, що вони роблять?
58. Що таке символна константа? Наведіть приклади символних констант.
59. Що таке рядок? Який тип даних в .NET Framework представляє рядок?
60. Наведіть декілька прикладів оголошення змінної типу string.
61. Як розташовуються у пам'яті рядки? Який максимальний розмір об'єкта string?
62. Які методи класу String ви знаєте, що вони роблять?
63. За допомогою якого методу можна порівняти два рядки? Запишіть у вигляді коду.
64. За допомогою якого методу можна об'єднати два рядки? Запишіть у вигляді коду.
65. За допомогою якого методу можна скопіювати один рядок в інший? Запишіть у вигляді коду.
66. За допомогою якого методу можна вставити підрядок з заданого індексу в інший рядок? Запишіть у вигляді коду.
67. За допомогою якого методу можна видалити задану кількість символів з рядка? Запишіть у вигляді коду.
68. За допомогою якого методу можна замінити символи в рядку? Запишіть у вигляді коду.
69. За допомогою якого методу можна виділити у рядку підрядок? Запишіть у вигляді коду.
70. За допомогою якого методу можна розділити рядок на масив підрядків? Запишіть у вигляді коду.
71. Які інтерфейси успадковує клас System.Array?
72. Перерахуйте та опишіть декілька методів класу System.Array.
73. Який недолік класу String компенсує клас StringBuilder?
74. Якими методами володіє клас StringBuilder?
75. Що таке буфер класу StringBuilder? Що таке ємність буфера? Чи можна її збільшувати та зменшувати?
76. Дайте визначення, що таке файлова система?
77. Що таке файл?
78. Які операції можна проводити з файлами?
79. Що таке формат файлу?
80. Що таке розширення файлу? Як воно записується?
81. Що таке каталог (папка)?
82. Які класи для роботи з файловою системою ви знаєте?
83. Які класи для роботи з потоками даних ви знаєте?
84. Які класи для роботи з каталогами (папками) ви знаєте?
85. Які класи для роботи з файлами ви знаєте?
86. Опишіть основні методи класу Directory.
87. Які операції з каталогами (папками) дозволяє робити клас DirectoryInfo?
88. Опишіть основні методи класу File.
89. Які операції з файлами дозволяє робити клас FileInfo?
90. Що буде, якщо не закрити потік при роботі з фалом за допомогою методу Close класу FileInfo?
91. Для чого призначений клас Path? Які методи цього класу ви знаєте?
92. Для чого призначені таблиці розміщення файлів FAT?
93. Яку інформацію може надавати клас Environment через свої статичні члени?
94. Які каталоги (папки) називаються спеціальними? Перерахуйте їх.

95. Які можливості надає клас FileStream?
96. Що таке парадигма програмування?
97. Які основні парадигми програмування ви знаєте?
98. Що таке імперативне програмування?
99. Що таке декларативне програмування?
100. Яка мова називається мультипарадигмальною? Наведіть приклади таких мов.
101. Що таке узагальнене програмування?
102. Що таке об'єктно-орієнтоване програмування?
103. Які переваги об'єктно-орієнтованого програмування?
104. Що таке функціональне програмування?
105. Що таке логічне програмування?
106. Що таке інкапсуляція?
107. Що таке поліморфізм?
108. Що таке наслідування?
109. Що таке клас?
110. Що таке екземпляр класу?
111. Наведіть синтаксис створення класу та екземпляру класу.
112. Дайте визначення поля та методу класу?
113. Для чого в методах використовується інструкція return? Чи у всіх випадках вона використовується?
114. Що таке перевантаження метода? Наведіть приклад.
115. Які модифікатори доступу до полів та методів класів ви знаєте?
116. Що таке конструктор класу? Чи завжди клас має хоча б один конструктор класу?
117. Що таке деструктор класу?
118. Який клас називається базовим, а який класом-нащадком?
119. Що таке заміщення метода батьківського класу?
120. Що таке колекція у C#?
121. Головна перевага використання колекцій у C#.
122. Які існують типи колекцій?
123. До якого типу колекцій належить клас List? Чому?
124. Чим відрізняються узагальнені колекції від неузагальнених?
125. Які інтерфейси використовуються для реалізації колекцій у C#?
126. Які основні класи колекцій ви знаєте?
127. Чим відрізняється клас Array від класу ArrayList?
128. Що таке стек, його принцип роботи?
129. Що таке черга, її принцип роботи?
130. Яка суть процесу хешування?

11. Навчально-методичне забезпечення

Навчальний процес з дисципліни «Алгоритмізація та програмування» повністю і в достатній кількості забезпечений необхідною навчально-методичною літературою. Зокрема, викладачами кафедри підготовлено і видано роботи:

1. Собко О. В., Молчанова М. О., Бармак О. В., Мазурець О. В. Алгоритмізація та програмування: методичні вказівки до лабораторних робіт для здобувачів першого (бакалаврського) рівня вищої освіти спеціальності 122 «Комп'ютерні науки». Хмельницький, ХНУ. 2022. 108с.
2. Собко О. В., Молчанова М. О., Мазурець О. В., Бармак О. В. Алгоритмізація та програмування : методичні рекомендації до курсового проектування для здобувачів першого (бакалаврського) рівня вищої освіти спеціальності 122 «Комп'ютерні науки» / О. В. Собко, М. О. Молчанова, О. В. Мазурець, О. В. Бармак. Хмельницький : ХНУ, 2024. 36 с..

12. Матеріально-технічне та програмне забезпечення дисципліни

Інформаційна та комп'ютерна підтримка: ПК, планшет, смартфон або інший мобільний пристрій, проектор. Програмне забезпечення: програми Microsoft Office або аналогічні, доступ до мережі Інтернет, робота з презентаціями.

Вивчення навчальної дисципліни потребує використання спеціального програмного прикладного забезпечення, а саме безкоштовної версії Microsoft Visual Studio Community.

13. Рекомендована література:

Основна

1. Собко О. В., Молчанова М. О., Бармак О. В., Мазурець О. В. Алгоритмізація та програмування: методичні вказівки до лабораторних робіт для здобувачів першого (бакалаврського) рівня вищої освіти спеціальності 122 «Комп'ютерні науки». Хмельницький, ХНУ. 2022. 108с.

2. Собко О. В., Молчанова М. О., Мазурець О. В., Бармак О. В. Алгоритмізація та програмування : методичні рекомендації до курсового проєктування для здобувачів першого (бакалаврського) рівня вищої освіти спеціальності 122 «Комп'ютерні науки». Хмельницький : ХНУ, 2024. 36 с.

3. Коноваленко І. В. Платформа .NET та мова програмування C# 8.0: навчальний посібник. Тернопіль, 2020. 320 с. URL: <https://elartu.tntu.edu.ua/bitstream/lib/32825/1/Konovalenko%20I.%20.NET-C%23.pdf>

4. Труніна Г.О., Настенко Д.В., Нестерко А.Б. Обчислювальна техніка та програмування: Конспект лекцій. Київ, КПІ ім. І.Сікорського. 2020. 117 с. URL: https://ela.kpi.ua/bitstream/123456789/39004/1/OTtaPr_KonspLek_1.pdf

Додаткова

5. How to automate Microsoft Excel from Microsoft Visual C# .NET. URL: <https://learn.microsoft.com/uk-ua/previous-versions/office/troubleshoot/office-developer/automate-excel-from-visual-c>

6. How to automate Microsoft Word to create a new document by using Visual C#. URL: <https://learn.microsoft.com/en-us/previous-versions/office/troubleshoot/office-developer/automate-word-create-file-using-visual-c>

7. Edit Word Documents using OpenXML and C#. URL: <https://www.codingame.com/playgrounds/11047/edit-word-documents-using-openxml-and-c-without-automationinterop>

8. Палеха Ю. І., Палеха О.Ю., Горбань Ю.І. Інформаційна культура: навч. посіб. . Київ : Видавництво Ліра-К, 2020 – 400 с. URL: <https://vstup.htek.com.ua/wp-content/uploads/2024/10/27.2-Palekha.pdf>

9. Цибульник, С. О. Технології розроблення програмного забезпечення. Частина 1. Життєвий цикл програмного. Київ, КПІ ім. Ігоря Сікорського, 2022. – 270 с. URL: <https://ela.kpi.ua/items/6291bbd6-9a89-40f5-9dac-5cbb772ebe43>

10. Eland M. Refactoring with C#: Safely Improve .NET Applications and Pay Down Technical Debt with Visual Studio, .NET 8, and C# 12. Packt Publishing, 2023 – 434 pp. URL: https://books.google.com.ua/books?id=IVPiEAAAQBAJ&printsec=frontcover&hl=uk&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false

11. Трофименко О. Г., Дика А. І. Тестування та забезпечення якості програмних систем : навч. посіб. Нац. ун-т «Одес. юрид. академія», Фенікс, 2024. – 195 с. URL: <https://dspace.onua.edu.ua/items/02ec927c-1484-4596-98bb-6999051f34e4>

12. Бармак О. В., Кліменко В. І., Мазурець О. В., Молчанова М. О. Собко О. В.. Основи програмної інженерії та тестування програмного забезпечення. Теоретичний курс та лабораторний практикум: навчальний посібник – Хмельницький. ХНУ, 2023 – 236 с.

13. Бармак О.В., Радюк П.М., Молчанова М.О., Собко О.В. Підходи до практичного аналізу обчислювальних алгоритмів. Вісник Хмельницького національного університету, № 6, 2021 (303), 2021. С.102-105.

14. Молчанова М.О., Мазурець О.В., Собко О.В., Віт. Р.В., Назаров В.В. Алгоритм виявлення аб'юзивного вмісту в україномовному аудіоконтенті для імплементації в об'єктно-орієнтовану інформаційну систему. Науковий журнал «Вісник Хмельницького національного університету» серія: Технічні науки. Хмельницький, 2024. №1 (331). С. 101-106.

14. Інформаційні ресурси

1. Модульне середовище для навчання. URL: <https://msn.khmnu.edu.ua/course/view.php?id=1798>

2. Електронна бібліотека ХНУ, URL: <http://library.khmnu.edu.ua/>

3. Інституційний репозитарій ХНУ, URL : <https://elar.khmnu.edu.ua/home>

АЛГОРИТМІЗАЦІЯ ТА ПРОГРАМУВАННЯ

Тип (статус) дисципліни	Обов'язкова загальної підготовки
Рівень вищої освіти	Перший (бакалаврський)
Мова викладання	Українська
Семестр	Перший, другий
Кількість призначених кредитів ЄКТС	13,0
Форми здобуття освіти, для яких викладається дисципліна	Очна (денна)

Результати навчання. Студент, який успішно завершив вивчення дисципліни, повинен: *знати* – основні обчислювальні та пошукові алгоритми, програмні засоби та способи відображення даних, основні інструменти програмування та їх застосування; *уміти* – читати та пояснювати алгоритми; обговорювати алгоритмічну коректність; використовувати та адаптувати стандартні алгоритми; розробляти алгоритми; читати та пояснювати значення програмних об'єктів; створювати, використовувати та змінювати програми, читати та пояснювати результати операцій; реалізовувати та описувати операції; створювати програми для реалізації стандартних алгоритмів; виявляти та виправляти логічні та синтаксичні помилки в програмному коді; описувати процес трансляції програми; визначати зображення різних типів даних за допомогою різних форматів; успішно використовувати інструменти програмування під час розробки програмного забезпечення; ефективно застосовувати стандартні комп'ютерні застосування.

Зміст навчальної дисципліни. Вступ до фаху. Алгоритмізація та проектування алгоритмів. Вступ до програмування. Мови програмування та їх елементи. Мова програмування C#. Константи, змінні. Типи даних. Оператори. Вирази. Методи. Алгоритмічна структура розгалуження мовою програмування. Алгоритмічна структура повторення мовою програмування. Парадигми програмування. Масиви. Одновимірні масиви. Багатовимірні масиви. Символьний та рядковий типи даних. Стандартні класи для роботи з масивами та рядками. Користувачі інтерфейси .NET Framework. Windows Forms. Обробка виняткових ситуацій. Робота з файловими структурами. Робота з спеціалізованими файлами. Сучасні та перспективні задачі у галузі комп'ютерних наук. Курсове проектування з дисципліни. Робота з інформаційними джерелами при вирішенні практичних задач в галузі комп'ютерних наук. Робота з інформаційними джерелами для вибору засобів розробки програмного забезпечення. Етапи розроблення та впровадження програмного забезпечення. Основи тестування програмного забезпечення. Використання структур даних при розробці програмного забезпечення. Основи багатопоточності. Основи Windows Presentation Foundation. Засоби Windows Presentation Foundation. Сучасні та перспективні засоби та середовища програмування.

Пререквізити: вихідна

Кореквізити: Алгоритмізація та програмування (курсний проєкт) (ОЗП.06) Статистична обробка даних (ОЗП.08), Методи та системи штучного інтелекту (ОФП.01), Вебтехнології (ОФП.03), Інтелектуальний аналіз даних (ОФП.05), Об'єктно-орієнтоване проектування (ОФП.08), Технології створення програмних продуктів (ОФП.09), Основи програмної інженерії та тестування програмного забезпечення (ОФП.10), Технології паралельних обчислень (ОФП.11), Системне та мережеве програмне забезпечення (ОФП.14), Кваліфікаційна робота (ОФП.18), КП Методи та системи штучного інтелекту (ОФП.02), КП Вебтехнології (ОФП.04), КП Інтелектуальний аналіз даних (ОФП.06), Професійна практика (ОФП.17). (це взято з окремої комірки АіП КП)

Запланована навчальна діяльність: Мінімальний обсяг навчальних занять в одному кредиті ЄКТС навчальної дисципліни для *першого* (бакалаврського) рівня вищої освіти за денною формою здобуття освіти становить 10 годин на 1 кредит ЄКТС.

Форми (методи) навчання: Процес навчання з дисципліни ґрунтується на використанні традиційних та сучасних технологій та методів навчання, зокрема: лекції (з використанням методів візуалізації, проблемного й інтерактивного навчання, мотиваційних прийомів, інформаційно-комунікаційних технологій); лабораторні заняття (з використанням інструктування, демонстрування, розв'язування типових і прикладних задач, аналізу кейсів, ситуаційних завдань, елементів дискусії тощо); самостійна робота (опрацювання теоретичного матеріалу, підготовка до виконання лабораторних робіт, поточного та підсумкового контролю, виконання індивідуальних завдань), з використанням інформаційно-комп'ютерних технологій та технологій дистанційного навчання.

Форми оцінювання результатів навчання: оцінювання лабораторних та контрольних робіт; виконання індивідуального завдання; підсумковий контрольний захід; курсовий проєкт.

Вид семестрового контролю: іспит – 1 семестр, залік – 2 семестр, курсовий проєкт – 2 семестр.

Навчальні ресурси:

- 1.Собко О. В., Молчанова М. О., Бармак О. В., Мазурець О. В. Алгоритмізація та програмування: методичні вказівки до лабораторних робіт для здобувачів першого (бакалаврського) рівня вищої освіти спеціальності 122 «Комп'ютерні науки». Хмельницький, ХНУ. 2022. 108с.
2. Собко О. В., Молчанова М. О., Мазурець О. В., Бармак О. В. Алгоритмізація та програмування : методичні рекомендації до курсового проектування для здобувачів першого (бакалаврського) рівня вищої освіти спеціальності 122 «Комп'ютерні науки». Хмельницький : ХНУ, 2024. 36 с.
- 3.Коноваленко І. В. Платформа .NET та мова програмування C# 8.0: навчальний посібник. Тернопіль, 2020. 320 с. URL: <https://elartu.tntu.edu.ua/bitstream/lib/32825/1/Konovalenko%20I.%20.NET-C%23.pdf>
- 4.Труніна Г.О., Настенко Д.В., Нестерко А.Б. Обчислювальна техніка та програмування: Конспект лекцій. Київ, КПІ ім. І.Сікорського. 2020. 117 с. URL: https://ela.kpi.ua/bitstream/123456789/39004/1/OTaPr_KonspLek_1.pdf
- 5.How to automate Microsoft Excel from Microsoft Visual C# .NET. URL: <https://learn.microsoft.com/uk-ua/previous-versions/office/troubleshoot/office-developer/automate-excel-from-visual-c>
- 6.How to automate Microsoft Word to create a new document by using Visual C#. URL: <https://learn.microsoft.com/en-us/previous-versions/office/troubleshoot/office-developer/automate-word-create-file-using-visual-c>
- 7.Edit Word Documents using OpenXML and C#. URL: <https://www.codingame.com/playgrounds/11047/edit-word-documents-using-openxml-and-c-without-automationinterop>
8. Палеха Ю. І., Палеха О.Ю., Горбань Ю.І. Інформаційна культура: навч. посіб. . Київ : Видавництво Ліра-К, 2020 – 400 с. URL: <https://vstup.htek.com.ua/wp-content/uploads/2024/10/27.2-Palekha.pdf>
9. Цибульник, С. О. Технології розроблення програмного забезпечення. Частина 1. Життєвий цикл програмного. Київ, КПІ ім. Ігоря Сікорського, 2022. – 270 с. URL: <https://ela.kpi.ua/items/6291bbd6-9a89-40f5-9dac-5cbb772ebe43>
- 10.Eland M. Refactoring with C#: Safely Improve .NET Applications and Pay Down Technical Debt with Visual Studio, .NET 8, and C# 12. Packt Publishing, 2023 – 434 pp. URL: https://books.google.com.ua/books?id=IVPiEAAAQBAJ&printsec=frontcover&hl=uk&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false
- 11.Трофименко О. Г., Дика А. І. Тестування та забезпечення якості програмних систем : навч. посіб. Нац. ун-т «Одес. юрид.академія», Фенікс, 2024. – 195 с. URL: <https://dspace.onua.edu.ua/items/02ec927c-1484-4596-98bb-6999051f34e4>
- 12.Бармак О. В., Кліменко В. І., Мазурець О. В., Молчанова М. О. Собко О. В.. Основи програмної інженерії та тестування програмного забезпечення. Теоретичний курс та лабораторний практикум: навчальний посібник – Хмельницький. ХНУ, 2023 – 236 с.
13. Бармак О.В., Радюк П.М., Молчанова М.О., Собко О.В. Підходи до практичного аналізу обчислювальних алгоритмів. Вісник Хмельницького національного університету, № 6, 2021 (303), 2021. С.102-105.
14. Молчанова М.О., Мазурець О.В., Собко О.В., Віт. Р.В., Назаров В.В. Алгоритм виявлення аб'юзивного вмісту в україномовному аудіоконтенті для імплементації в об'єктно-орієнтовану інформаційну систему. Науковий журнал «Вісник Хмельницького національного університету» серія: Технічні науки. Хмельницький, 2024. №1 (331). С. 101-106.
15. Модульне середовище для навчання. URL: <https://msn.khmnua.edu.ua/course/view.php?id=1798>
16. Електронна бібліотека ХНУ, URL: <http://library.khmnua.edu.ua/>
17. Інституційний репозитарій ХНУ, URL : <https://elar.khmnua.edu.ua/home>

Викладачі: Ph.D, ст.викл., Собко О.В.
к.т.н., доцент, Мазурець О.В..