

**МІНІСТЕРСТВО ВНУТРІШНІХ СПРАВ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ВНУТРІШНІХ СПРАВ**

Кафедра протидії кіберзлочинності, факультет № 4

РОБОЧА ПРОГРАМА

навчальної дисципліни „Алгоритмізація та програмування”
обов’язкових компонент освітньо-професійної програми
першого (бакалаврського) рівня вищої освіти

125 "Кібербезпека та захист інформації" («Безпека інформаційних та
комунікаційних систем»)

Харків 2023

ЗАТВЕРДЖЕНО

Науково-методичною радою
Харківського національного
університету внутрішніх справ
Протокол від 31.10.2023 №9

СХВАЛЕНО

Вченою радою факультету № 4
Протокол від 18.10.2023 №11

ПОГОДЖЕНО

Секцією Науково-методичної ради
ХНУВС з технічних дисциплін

Протокол від 27.10.2023 №9

Розглянуто на засіданні кафедри протидії кіберзлочинності
(протокол від 16.10.2023 №24)

Розробники:

1. *Доцент кафедри, к.т.н., доцент Слободянюк О.В.*

Рецензенти:

1. *Провідний науковий співробітник Науково-дослідної лабораторії з проблем розвитку інформаційних технологій ХНУВС, канд.техн.наук, доцент Мордвинцев М.В.;*

2. *Канд.техн.наук, доцент, доцент кафедри програмної інженерії ХНУРЕ Лановий О.Ф.*

1. Опис навчальної дисципліни

Найменування показників	Шифри та назви галузі знань, код та назва спеціальності, ступень вищої освіти	Характеристика навчальної дисципліни
Кількість кредитів ECTS – 15 Загальна кількість годин – 450 Кількість тем – 19	12 Інформаційні технології; (шифр галузі) (назва галузі знань) 125 – Кібербезпека бакалавр (назва СВО)	Цикл дисциплін професійної та практичної підготовки Навчальний курс – 1,2,3 Семестр – 2,3,4,5 Види контролю: залік, екзамен
Тижневих годин для денної форми навчання: аудиторних – 3,5 самостійної роботи – 3,5		Розподіл навчальної дисципліни за видами занять: (денна форма навчання) Лекції – 88; Практичні заняття – 40; Лабораторні заняття – 84; Самостійна робота – 238; (заочна форма навчання) Лекції – 16; Практичні заняття – 8; Лабораторні заняття – 16; Самостійна робота – 410; Індивідуальні завдання: КР - 4 семестр

2. Мета та завдання навчальної дисципліни

Мета викладання навчальної дисципліни „Алгоритмізація та програмування”: вивчення і освоєння засобів алгоритмізації і програмування, розвиток аналітичних здібностей студентів, їх вміння логічно мислити, будувати логічні ланцюжки і застосовувати ці вміння для розв’язання навчальних та практичних задач.

Завдання вивчення дисципліни „Алгоритмізація та програмування”:

- розвиток у студентів практичних навичок побудови алгоритмів розв’язання практичних задач і їх подальшої програмної реалізації за допомогою сучасних інструментальних систем програмування;
- підготовка студентів до ефективного розв’язання задач, що постають в процесі наступного навчання і в подальшій професійній діяльності.

Міждисциплінарні зв’язки курсу «Алгоритмізація та програмування» обумовлені тим, що він ґрунтується на дисципліні «Вища математика» (1 курс).

Очікувані результати навчання: у результаті вивчення навчальної дисципліни здобувач вищої освіти повинен

знати:

- методику складання алгоритмів,
- алгоритмічну мову C#,
- методику розробки програм у середовищі інструментальної системи

- візуального програмування Visual Studio;
- типові алгоритмічні рішення та їх програмну реалізацію в середовищі візуального програмування Visual Studio;
- методи аналізу програмного коду;

вміти:

- розробляти алгоритми і програми на алгоритмічній мові C# в середовищі Visual Studio,
- відлагоджувати програми розв'язання задач на ПЕОМ;
- розв'язувати задачі практичної спрямованості за допомогою інструментальної системи програмування Visual Studio.
- аналізувати програмний код.

В навчальному плані для вивчення дисципліни передбачені такі організаційні форми занять як лекції, практичні і лабораторні заняття.

Програмні компетентності, які формуються при вивченні навчальної дисципліни:		
Інтегральна компетентність	Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми у галузі забезпечення інформаційної безпеки і\або кібербезпеки, що характеризується комплексністю та неповною визначеністю умов.	
Загальні компетентності (ЗК)	ЗК 1.	Здатність застосовувати знання у практичних ситуаціях.
	ЗК 5.	Здатність до пошуку, оброблення та аналізу інформації.
Фахові компетентності спеціальності (ФК)	ФК 2.	Здатність до використання інформаційно-комунікаційних технологій, сучасних методів і моделей інформаційної безпеки та\або кібербезпеки.
	ФК 3.	Здатність до використання програмних та програмно-апаратних комплексів засобів захисту інформації в інформаційно-телекомунікаційних (автоматизованих) системах.
Програмні результати навчання	ПРН 2.	Організовувати власну професійну діяльність, обирати оптимальні методи та способи розв'язування складних спеціалізованих задач та практичних проблем у професійній діяльності, оцінювати їхню ефективність.

	ПРН 4.	Аналізувати, аргументувати, приймати рішення при розв'язанні складних спеціалізованих задач та практичних проблем у професійній діяльності, які характеризуються комплексністю та неповною визначеністю умов, відповідати за прийняті рішення.
	ПРН 53.	Вирішувати задачі аналізу програмного коду на наявність можливих загроз.

3. Програма навчальної дисципліни

Тема № 1. Основні поняття алгоритмізації.

Місце, роль та задачі дисципліни. Предмет, структура і зміст дисципліни. Етапи розв'язання задач за допомогою ЕОМ. Введення в алгоритмізацію. Поняття алгоритму та його властивості.

Способи уявлення алгоритмів. Класифікація алгоритмічних мов та їх порівняльний аналіз. Транслятори: компілятори та інтерпретатори.

Типи алгоритмічних процесів і структур. Лінійні алгоритмічні процеси і алгоритмічні структури. Розгалужувальні алгоритмічні процеси і алгоритмічні структури. Циклічні алгоритмічні процеси і алгоритмічні структури.

Тема № 2. Алгоритмізація типових задач.

Типові алгоритмічні рішення. Типові лінійні алгоритми. Типові розгалужувальні алгоритми. Ітераційні цикли. Типові алгоритмічні рішення обробки одновимірних масивів.

Обробка двовимірних масивів. Вкладені цикли. Організація вкладених циклів. Типові алгоритмічні рішення обробки матриць.

Сортування та пошук.

Тема № 3. Вступ у C#.

Загальні відомості про алгоритмічну мову C#. Типи даних. Вирази у C#. Арифметичні вирази. Логічні вирази. Вбудовані функції. Консольне введення-виведення даних в C#.

Найпростіші оператори. Перетворення типів. Оператори керування (goto, if, switch). Програмування циклів. Оператори циклу for, while, do while, foreach.

Тема № 4. Робота з масивами..

Масиви. Оголошення масивів. Операції з масивами. Типові програмні рішення. Сума та добуток елементів масиву. Максимальний та мінімальний елементи масиву. Комбіновані задачі. Програмування алгоритмів сортування та пошуку.

Тема № 5. Обробка текстових і комбінованих даних.

Засоби обробки текстових даних у C#. Стандартні функції обробки текстових даних. Типові програмні рішення обробки текстів.

Комбіновані типи даних та засоби їх обробки. Оголошення структур. Поля

і методи структур. Масиви структур і робота з ними.

Тема № 6. Підпрограми(методи).

Класифікація підпрограм(методів). Процедури та функції. Засоби обміну даними між програмою та підпрограмою. Формальні та фактичні параметри. Передача параметрів за посиланням та за значенням. Область дії імен. Локальні та глобальні елементи.

Тема № 7. Основи об'єктно-орієнтованого програмування.

Поняття класу. Інкапсуляція. Спадкування класів.

Модифікатори доступу. Поліморфізм. Перевизначення методів. Конструктори та деструктори. Поняття події. Виключення. Обробка подій. Абстрактні класи. Делегати.

Тема № 8. Узагальнення. Інтерфейси. Колекції.

Узагальнені класи. Використання декількох узагальнених параметрів. Узагальнені методи. Інтерфейси.

Колекції. Загальні відомості про колекції. Узагальнені колекції. Списки. Клас List<T>. Колекція Dictionary<K, V>(словник). Колекція SortedDictionary<K, V>.

Множини: класи HashSet<T> та SortedSet<T>. Клас HashSet<T>. Клас SortedSet<T>.

Кортежі. Клас Tuple.

Тема № 9. Робота з файлами.

Загальна структура файлу. Робота з потоками і файловою системою. Робота з дисками. Робота з каталогами. Робота з файлами. Класи File та FileInfo. Читання і запис файлу. Клас FileStream. Довільний доступ до файлів. Читання і запис текстових файлів. StreamReader и StreamWriter. Робота з бінарними файлами. BinaryWriter и BinaryReader.

Тема № 10. Введення у середовище візуального програмування Windows Forms.

Основні терміни та поняття. Структура середовища. Методика розробки програм у Windows Forms.

Найпростіше введення-виведення даних у Windows Forms. Способи найпростішого введення даних. Компоненти TextBox, MaskedTextBox. Способи найпростішого виведення даних. Компоненти Label, LinkLabel. Вікно виведення MessageBox.

Командні кнопки. Компонент Button. Кнопки-перемикачі. Компоненти Radiobutton и CheckBox.

Тема № 11. Діалогова взаємодія з користувачем.

Організація діалогової взаємодії користувача з програмою. Панель інструментів ToolStrip. Створення головного меню MenuStrip. Створення спливаючого меню ContextMenuStrip.

Тема № 12. Обробка масивів даних у Windows Forms.

Обробка одновимірних і двовимірних масивів. Компоненти ListBox, ComboBox, CheckedListBox, ListView, DataGridView.

Тема № 13. Діалоги. Контейнери.

Тема № 1: Основні поняття алгоритмізації.	10	6				4	к/р
Тема № 2: Алгоритмізація типових задач.	78	14		14	12	38	к/р
Тема № 3: Вступ у C#.	32	8		4	4	16	к/р
Всього за семестр № 2:	120	28		18	16	58	залік
Семестр № 3							
Тема № 3: Вступ у C#.	4	2				2	к/р
Тема № 4: Типові програмні рішення: робота з масивами.	30	4		4	8	14	
Тема № 5: Обробка текстових і комбінованих даних.	34	6		4	8	16	к/р
Тема № 6: Підпрограми(методи).	7	4				3	
Тема № 7: Основи об'єктно-орієнтованого програмування.	8	4				4	
Тема № 8: Узагальнення. Інтерфейси. Колекції.	32	4		4	8	16	к/р
Тема № 9: Робота з файлами.	20	4		2	4	10	
Всього за семестр № 3:	135	28		14	28	65	залік
Семестр № 4							
Тема № 10: Введення у середовище візуального програмування Windows Forms.	24	6		2	4	12	к/р
Тема № 11: Діалогова взаємодія з користувачем.	16	2		2	4	8	
Тема № 12: Обробка масивів даних у Windows Forms.	28	4		2	8	14	
Тема № 13: Діалоги.	6	2				4	к/р
Тема № 14: Індикатори процесів і робота з датами у Windows Forms.	6	2				4	
Тема № 15: Графіка у Windows Forms.	25	6		2	4	13	
Всього за семестр № 4:	105	22		8	20	55	екзамен
Семестр № 5							
Тема № 16: Поняття зворотної розробки програмного забезпечення.	12	2				10	
Тема № 17: Огляд шкідливого програмного забезпечення.	16	2			4	10	
Тема № 18: Статичний аналіз програмного забезпечення.	30	2			8	20	
Тема № 19: Динамічний аналіз програмного забезпечення.	32	4			8	20	
Всього за семестр № 5:	90	10			20	60	залік
Всього по дисципліні	450	88		40	84	238	

4.2. Розподіл часу навчальної дисципліни за темами (заочна форма навчання)

Номер та назва навчальної теми	Кількість годин, відведених на вивчення навчальної дисципліни						Вид контролю
	Всього	з них:					
		Лекції	Семінарські заняття	Практичні заняття	Лабораторні заняття	Самостійна робота	
Семестр № 2							
Тема № 1: Основні поняття алгоритмізації.	31	1				30	к/р
Тема № 2: Алгоритмізація типових задач.	45	2		1	2	40	
Тема № 3: Вступ у C#.	44	1		1	2	40	к/р
Всього за семестр № 2:	120	4		2	4	110	10 залік
Семестр № 3							
Тема № 3: Вступ у C#.	11	1				10	к/р
Тема № 4: Типові програмні рішення: робота з масивами.	25	1		1	1	22	
Тема № 5: Обробка текстових і комбінованих даних.	12			1	1	10	
Тема № 6: Підпрограми(методи).	15	1				14	к/р
Тема № 7: Основи об'єктно-орієнтованого програмування.	22					22	к/р
Тема № 8: Узагальнення. Інтерфейси. Колекції.	26	1			1	24	
Тема № 9: Робота з файлами.	24				1	23	
Всього за семестр № 3:	135	4		2	4	125	10 залік
Семестр № 4							
Тема № 10: Введення у середовище візуального програмування Windows Forms.	14	1			1	12	к/р
Тема № 11: Діалогова взаємодія з користувачем.	18	1			1	16	
Тема № 12: Обробка масивів даних у Windows Forms.	20	1		1	1	17	
Тема № 13: Діалоги.	17			1		16	к/р
Тема № 14: Індикатори процесів і робота з датами у Windows Forms.	11			1		10	
Тема № 15: Графіка у Windows Forms.	25	1		1	1	22	
Всього за семестр № 4:	105	4		4	4	93	12 екзамен
Семестр № 5							
Тема № 16: Поняття зворотної розробки програмного забезпечення.	21	1				20	
Тема № 17: Огляд шкідливого програмного забезпечення.	22	1			1	20	

Тема № 18: Статичний аналіз програмного забезпечення.	22	1			1	20	
Тема № 19: Динамічний аналіз програмного забезпечення.	25	1			2	22	
Всього за семестр № 5:	90	4			4	82	залік
Всього по дисципліні	450	16		8	16	410	

4.3. Питання, що виносяться на самостійне опрацювання

Перелік питань до тем навчальної дисципліни		Література:
Тема № 1: Основні поняття алгоритмізації.		1
	Способи подання алгоритмів. Запис алгоритму природною мовою. Графічний спосіб. Запис алгоритму алгоритмічною мовою.	1
	Типи алгоритмічних процесів і структур.	1
Тема № 2: Алгоритмізація типових задач.		1
	Розробка типових алгоритмів з ітераційними циклами.	1
	Типові алгоритмічні розв'язки задач обробки одновимірних масивів: пошук номера максимального елемента, сортування.	1
	Типові алгоритмічні розв'язки задач обробки двовимірних масивів: пошук номера максимального і мінімального елементів.	1
	Засоби контролю правильності алгоритмів	1
Тема № 3: Вступ у C#.		2, 4
	Типи числових змінних і констант і форми їх подання.	2, 3, 4
	Основи роботи в консольному середовищі.	2, 4
	Засоби відлагодження програм в C#.	4, 6
	Особливості введення виведення даних в C#.	2, 4, 6
Тема № 4: Типові програмні рішення: робота з масивами.		2, 4
	Побудова типових програм з ітераційними циклами.	2, 4
	Оператори циклу While і Foreach.	2, 4
	Розробка типових програм пошуку номера мінімального і максимального елементів одновимірних масивів.	2, 4
	Розробка типових програм пошуку номера мінімального і максимального елементів двовимірних масивів.	2, 4
Тема № 5: Обробка текстових і комбінованих даних.		2, 4
	Стандартні процедури обробки текстових даних.	2, 4
	Розробка типових програм обробки текстових даних.	2, 4
	Розробка типових програм обробки комбінованих даних.	2, 4
Тема № 6: Підпрограми(методи).		2, 4
	Підпрограми-функції і підпрограми-процедури.	2, 4
	Глобальні змінні. Область дії змінних.	2, 4
Тема № 7: Основи об'єктно-орієнтованого програмування.		2, 4
	Директиви.	2, 4
	Конструктори і деструктори.	2, 4
	Спадкування.	2, 4
	Віртуальні методи. Перевизначення імен.	2, 4
	Абстрактні класи.	2, 4
	Делегати.	2, 4

	Тема № 8: Узагальнення. Інтерфейси. Колекції.	2, 4
	Узагальнені класи.	2, 4
	Узагальнені методи.	2, 4
	Списки. Клас List<T>.	2, 4
	Словники. Клас Dictionary<K, V>.	2, 4
	Множини.	2, 4
	Тема № 9: Робота з файлами.	2, 4
	Особливості роботи з текстовими файлами.	2, 4
	Особливості роботи з бінарними файлами.	2, 4
	Тема № 10: Введення в середовище візуального програмування Windows Forms.	4, 6, 7
	Структура середовища.	4, 6, 7
	Особливості роботи з візуальними компонентами.	4, 6, 7
	Тема № 11: Діалогова взаємодія з користувачем.	4, 6, 7
	Панель інструментів ToolStrip. Рядок стану StatusStrip.	4, 6, 7
	Створення спливаючого меню ContextMenuStrip.	4, 6, 7
	Компоненти GroupBox, Panel, FlowLayoutPanel, TableLayoutPanel. Панель вкладок TabControl и SplitContainer.	4, 6, 7
	Тема № 12: Обробка масивів даних у Windows Forms.	4, 6, 7
	Особливості розробки програм обробки одновимірних масивів у Windows Forms.	4, 6, 7
	Особливості розробки програм обробки двовимірних масивів у Windows Forms.	4, 6, 7
	Тема № 13: Діалоги.	4, 6, 7
	Діалогові компоненти OpenFileDialog, SaveFileDialog, FontDialog, FolderBrowserDialog.	4, 6, 7
	Діалогові компоненти ColorDialog, PrintDialog, PageSetupDialog. Компонент WebBrowser.	4, 6, 7
	Тема № 14: Індикатори процесів і робота з датами у Windows Forms.	6,7
	Індикатори процесів - компоненти TrackBar, Timer и ProgressBar.	6,7
	Робота з датами – компоненти DateTimePicker и MonthCalendar.	6,7
	Тема № 15: Графіка у Windows Forms.	6,7
	Об'єкт Graphics, класи Pen, Font, Brush, Color. Побудова графічних примітивів – ліній і фігур.	6,7
	Об'єкт Graphics, класи Pen, Font, Brush, Color. Побудова графічних примітивів – ліній і фігур.	6,7
	Тема № 16: Поняття зворотної розробки програмного забезпечення.	4,8
	Сутність і задачі реверсінжинірингу.	4,8
	Засоби реверсінжинірингу.	4,8
	Тема № 17: Огляд шкідливого програмного забезпечення.	4,8
	Методики огляду шкідливого програмного забезпечення.	4,8
	Антивірусні засоби.	4,8
	Тема № 18: Статичний аналіз програмного забезпечення.	4,8
	Сутність статичного аналізу.	4,8
	Засоби статичного аналізу.	4,8
	Тема № 19: Динамічний аналіз програмного забезпечення.	4,8
	Сутність динамічного аналізу.	4,8
	Засоби динамічного аналізу.	4,8

5. Індивідуальні завдання

5.1. Теми курсових робіт (2 семестр).

Тематика курсової роботи – розробка алгоритмів і програм розв’язання практичних і навчальних задач в сучасних інструментальних середовищах програмування. Орієнтовні теми курсових робіт наведені у Методичних рекомендаціях до виконання курсових робіт.

6. Методи навчання

Лекції із застосуванням електронних матеріалів; мультимедійні лекції; практичні та лабораторні заняття в комп’ютерному класі.

7. Перелік питань та завдань, що виносяться на підсумковий контроль

1. Етапи розв’язання задач за допомогою ЕОМ.
2. Складання математичних моделей задач.
3. Класифікація алгоритмічних мов та їх зрівняльний аналіз.
4. Поняття алгоритму та алгоритмізації. Властивості алгоритму.
5. Способи уявлення алгоритмів.
6. Типи алгоритмічних процесів і алгоритмічних структур.
7. Складання математичної моделі розв’язуваної задачі.
8. Розробка алгоритмів розгалужувальних процесів
9. Типові алгоритмічні структури і процеси.
10. Розробка алгоритмів пошуку суми та добутку елементів одновимірних масивів.
11. Розробка алгоритмів пошуку мінімального та максимального елементів одновимірних масивів.
12. Вкладені цикли. Організація вкладених циклів.
13. Розробка алгоритмів пошуку суми та добутку елементів матриць.
14. Розробка алгоритмів пошуку мінімального та максимального елементів матриць.
15. Загальні відомості про C#. Види об’єктів. Типи даних.
16. Структура програми.
17. Масиви. Оголошення масивів. Операції з масивами.
18. Введення-виведення масивів.
19. Введення даних з клавіатури. Особливості.
20. Виведення даних на дисплей. Особливості.
21. Арифметичні вирази.
22. Логічні вирази.
23. Прості і структуровані оператори.
24. Оператори керування (goto, if, switch). Програмування циклів за допомогою операторів goto, if.
25. Оператор циклу for.

26. Оператор циклу while.
27. Оператор циклу do while.
28. Оператор циклу foreach.
29. Типові програмні рішення. Сума та добуток елементів двовимірного масиву.
30. Типові програмні рішення. Максимальний та мінімальний елементи двовимірного масиву.
31. Програмування алгоритмів пошуку.
32. Рядковий тип. Оголошення і особливості зберігання в пам'яті. Операції з рядками.
33. Методи обробки рядків.
34. Тип structure. Оголошення і звернення до структур і їх полів. Масиви структур.
35. Робота з датами і часом. Структура DateTime.
36. Поняття підпрограми. Типи підпрограм.
37. Оголошення підпрограми, її структура.
38. Процедури в C#. Оголошення процедури, звернення до неї.
39. Функції в C#. Оголошення функції, звернення до неї.
40. Параметри-значення та параметри-змінні.
41. Локальні та глобальні об'єкти. Область дії імен. Стек. Купа.
42. Особливості ООП у C#: інкапсуляція, спадкування, поліморфізм.
43. Класи: поля, методи, властивості.
44. Конструктори і деструктори.
45. Засоби керування спадкуванням.
46. Узагальнені класи. Використання декількох узагальнених параметрів.
47. Узагальнені методи.
48. Інтерфейси.
49. Колекції. Загальні відомості про колекції.
50. Узагальнені колекції.
51. Списки. Клас List<T>.
52. Колекція Dictionary<K, V>(словник).
53. Колекція SortedDictionary<K, V>.
54. Множини: класи HashSet<T> та SortedSet<T>.
55. Кортежі. Клас Tuple.
56. Файли. Загальна структура файлу. Способи формування файлів.
57. Читання і запис файлу. Клас FileStream. Довільний доступ до файлів.
58. Читання і запис текстових файлів. StreamReader и StreamWriter.
59. Робота з бінарними файлами. BinaryWriter и BinaryReader.
60. Загальні відомості про C# та Windows Forms.
61. Методика розробки програм у Windows Forms. Структура робочого екрану. Призначення елементів головного меню. Призначення і використання вікон середовища Windows Forms.
62. Поняття події у Windows Forms. Розробка обробників подій. Структура та методика побудови стартової форми додатку. Методика відлагодження програми та запуску її на виконання.

63. Розміщення компонентів, способи визначення їх властивостей. Техніка використання стандартних компонентів.
64. Засоби введення даних у Windows Forms. Компоненти TextBox, MaskedTextBox, методика їх використання.
65. Засоби виведення даних у Windows Forms. Компоненти Label, LinkLabel. Вікно виведення MessageBox.
66. Командні кнопки. Компонент Button, його властивості та особливості.
67. Кнопки-перемикачі. Компоненти RadioButton, CheckBox, їх властивості та особливості.
68. Побудова головного меню. Компонент MenuStrip, його властивості та особливості застосування.
69. Побудова контекстного меню. Компонент ContextMenuStrip, його властивості та особливості застосування.
70. Введення-виведення одновимірних масивів. Компоненти ListBox, ComboBox, CheckedListBox.
71. Компоненти ListView, TreeView.
72. Робота з двовимірними масивами. Компонент DataGridView.
73. Панель інструментів ToolStrip і рядок стану StatusStrip.
74. Діалогові компоненти OpenFileDialog, SaveDialog.
75. Діалогові компоненти FontDialog, ColorDialog, PrintDialog.
76. Графіка у Windows Forms. Об'єкт Graphics.
77. Графіка у Windows Forms. Клас Pen.
78. Графіка у Windows Forms. Клас Font.
79. Графіка у Windows Forms. Клас Brush.
80. Графіка у Windows Forms. Клас Color.
81. Побудова графічних примітивів – ліній і фігур.
82. Сутність і задачі реверсінжинірингу.
83. Цілі застосування зворотної розробки програмного забезпечення.
84. Області застосування зворотної розробки програмного забезпечення.
85. Основні типи програмного забезпечення.
86. Сучасні комп'ютерні архітектури.
87. Засоби віртуалізації.
88. Мета використання засобів віртуалізації.
89. Типи шкідливого програмного забезпечення.
90. Методології виявлення шкідливого програмного забезпечення.
91. Сучасні антивірусні засоби.
92. Поняття сигнатури.
93. Базовий алгоритм дослідження шкідливого програмного забезпечення.
94. Поняття статичного аналізу.
95. Основні принципи статичного аналізу.
96. Типи захисту програмного забезпечення.
97. Аналіз заголовків файлів.
98. Таблиця імпорту.
99. Інструменти статичного аналізу програмного забезпечення.
100. Поняття динамічного аналізу.

101. Принципи динамічного аналізу.
102. Моніторинг процесів.
103. Сучасні файлові системи.
104. Аналіз реєстрів.
105. Дослідження мережевих з'єднань.
106. Інструменти динамічного аналізу.

8. Критерії та засоби оцінювання результатів навчання здобувачів

Контрольні заходи оцінювання результатів навчання включають в себе поточний та підсумковий контроль.

Засобами оцінювання результатів навчання можуть бути: екзамени (комплексні екзамени); тести; наскрізні проекти; командні проекти; аналітичні звіти, реферати, есе; розрахункові та розрахунково-графічні роботи; презентації результатів виконаних завдань та досліджень; завдання на лабораторному обладнанні, тренажерах, реальних об'єктах тощо; інші види індивідуальних та групових завдань.

Поточний контроль.

До форм поточного контролю належить оцінювання:

- рівня знань під час семінарських, практичних, лабораторних занять;
- якості виконання індивідуальної та самостійної роботи.

Поточний контроль здійснюється під час проведення семінарських, практичних та лабораторних занять і має на меті перевірку набутих здобувачем вищої освіти (далі – здобувач) знань, умінь та інших компетентностей з навчальної дисципліни.

У ході поточного контролю проводиться систематичний вимір приросту знань, їх корекція. Результати поточного контролю заносяться викладачем до журналів обліку роботи академічної групи за національною системою оцінювання («відмінно», «добре», «задовільно», «незадовільно»).

Оцінки за самостійну та індивідуальну роботу виставляються в журналі обліку роботи академічної групи окремою графою за національною системою оцінювання («відмінно», «добре», «задовільно», «незадовільно»). Результати цієї роботи враховуються під час виставлення підсумкових оцінок.

При розрахунку успішності здобувачів в Університеті враховуються такі види робіт: навчальні заняття (семінарські, практичні, лабораторні тощо); самостійна та індивідуальна роботи (виконання домашніх завдань, ведення конспектів першоджерел та робочих зошитів, виконання розрахункових завдань, підготовка рефератів, наукових робіт, публікацій, розроблення спеціальних технічних пристроїв і приладів, моделей, комп'ютерних програм, виступи на наукових конференціях, семінарах та інше); контрольні роботи (виконання тестів, контрольних робіт у формі, передбаченій в робочою програмою навчальної дисципліни). Вони оцінюються за національною системою оцінювання («відмінно», «добре», «задовільно», «незадовільно»).

Здобувач, який отримав оцінку «незадовільно» за навчальні заняття

або самостійну роботу, зобов'язаний перескласти її.

Загальна кількість балів (оцінка), отримана здобувачем за семестр перед підсумковим контролем, розраховується як середньоарифметичне значення з оцінок за навчальні заняття та самостійну роботу, та для переводу до 100-бальної системи помножується на коефіцієнт **10**.

$$\begin{array}{l} \text{Загальна кількість} \\ \text{балів (перед} \\ \text{підсумковим} \\ \text{контролем)} \end{array} = \left(\begin{array}{l} \text{Результат} \\ \text{навчальних занять} \\ \text{за семестр} \end{array} + \begin{array}{l} \text{Результат} \\ \text{самостійної} \\ \text{роботи за семестр} \end{array} \right) / 2 * 10$$

Підсумковий контроль.

Підсумковий контроль проводиться з метою оцінки результатів навчання на певному ступені вищої освіти або на окремих його завершених етапах.

Для обліку результатів підсумкового контролю використовується поточно-накопичувальна інформація, яка реєструється в журналах обліку роботи академічної групи. Результати підсумкового контролю з дисциплін відображаються у відомостях обліку успішності, навчальних картках здобувачів, залікових книжках. **Присутність здобувачів на проведенні підсумкового контролю (заліку, екзамену) обов'язкова.** Якщо здобувач вищої освіти не з'явився на підсумковий контроль (залік, екзамен), то науково-педагогічний працівник ставить у відомість обліку успішності відмітку «не з'явився».

Підсумковий контроль (екзамен, залік) оцінюється за національною шкалою. Для переводу результатів, набраних на підсумковому контролі, з національної системи оцінювання в 100-бальну вводиться коефіцієнт **10**, таким чином максимальна кількість балів на підсумковому контролі (екзамені, заліку), які використовуються при розрахунку успішності здобувачів, становить - **50**

Підсумкові бали з навчальної дисципліни визначаються як сума балів, отриманих здобувачем протягом семестру та балів, набраних на підсумковому контролі (екзамені, заліку).

$$\begin{array}{l} \text{Підсумкові бали} \\ \text{навчальної дисципліни} \end{array} = \begin{array}{l} \text{Загальна кількість балів (перед} \\ \text{підсумковим контролем)} \end{array} + \begin{array}{l} \text{Кількість балів за} \\ \text{підсумковим контролем} \end{array}$$

Здобувач вищої освіти, який під час складання підсумкового контролю (екзамен, залік) отримав незадовільну оцінку, складає його повторно. Повторне складання підсумкового екзамену чи заліку допускається не більше двох разів з кожної навчальної дисципліни: один раз – викладачеві, а другий – комісії, до складу якої входить керівник відповідної кафедри та 2-3 науково-педагогічних працівника.

Якщо дисципліна вивчається протягом двох і більше семестрів з семестровим контролем у формі екзамену чи заліку, то результат вивчення дисципліни в поточному семестрі визначається як середньоарифметичне значення балів, набраних у поточному та попередньому семестрах.

$$\begin{array}{l} \text{Підсумкові бали} \\ \text{навчальної дисципліни} \end{array} = \begin{array}{l} \text{Підсумкові бали за} \\ \text{поточний семестр} \end{array} + \begin{array}{l} \text{Підсумкові бали за} \\ \text{попередній семестр} \end{array} : 2$$

Вимоги до здобувачів стосовно засвоєння змісту навчальної дисципліни:

Робота під час навчальних занять	Самостійна та індивідуальна робота	Підсумковий контроль
Отримати позитивні оцінки за кожну лабораторну роботу	Підготувати звіт до кожної лабораторної роботи, вирішити індивідуальне завдання до кожної лабораторної роботи, вирішити домашні завдання до практичних занять, вивчити теоретичні питання з підготовки до практичних занять.	Отримати за підсумковий контроль не менше 30 балів

9. Шкала оцінювання: національна та ECTS

Оцінка в балах	Оцінка за національною шкалою	Оцінка	
		Оцінка	Пояснення
97-100	Відмінно («зараховано»)	A	«Відмінно" – теоретичний зміст курсу засвоєний цілком , необхідні практичні навички роботи з освоєним матеріалом сформовані, усі навчальні завдання, які передбачені програмою навчання виконані в повному обсязі, відмінна робота без помилок або з однією незначною помилкою.
94-96			
90-93			
85- 89	Добре («зараховано»)	B	«Дуже добре" – теоретичний зміст курсу засвоєний цілком , потрібні практичні навички роботи з освоєним матеріалом в основному сформовані, усі навчальні завдання, які передбачені програмою навчання, виконані , якість виконання більшості з них оцінена числом балів, близьким до максимального , робота з двома-трьома незначними помилками.
80-84			
75-79		C	«Добре" – теоретичний зміст курсу засвоєний цілком , практичні навички роботи з освоєним матеріалом в основному сформовані, усі навчальні завдання, які передбачені програмою навчання, виконані , якість виконання жодного з них не оцінена мінімальним числом балів, деякі види завдань виконані з помилками , робота з декількома незначними помилками, або з однією-двома значними помилками.
70 -74	Задовільно («зараховано»)	D	«Задовільно" – теоретичний зміст курсу засвоєний частково , але прогалини не несуть істотний характер, потрібні практичні навички роботи з освоєним матеріалом в основному сформовані, більшість передбачених програмою навчання навчальних завдань виконана , деякі з виконаних завдань містять помилки , робота з трьома значними помилками.
65-69			

60-64		E	«Достатньо» – теоретичний зміст курсу освоєний частково , деякі практичні навички роботи не сформовані , частина передбачених програмою навчання навчальних завдань не виконана , або якість виконання деяких з них оцінена числом балів, близьким до мінімального , робота, що задовольняє мінімуму критеріїв оцінки.
41-59	Незадовільно («не зараховано»)	F X	«Умовно незадовільно» – теоретичний зміст курсу засвоєний частково , потрібні практичні навички роботи несформовані , більшість передбачених програмою навчання, навчальних завдань не виконано , або якість їхнього виконання оцінено числом балів, близьким до мінімального ; при додатковій самостійній роботі над матеріалом курсу можливе підвищення якості виконання навчальних завдань (з можливістю повторного складання), робота, що потребує доробки.
21-40			
1-20		F	« Безумовно незадовільно » – теоретичний зміст курсу неосвоєний , потрібні практичні навички роботи несформовані , всі виконані навчальні завдання містять грубі помилки , додаткова самостійна робота над матеріалом курсу не приведе до значного підвищення якості виконання навчальних завдань, робота, що потребує повної переробки.

10. Рекомендована література (основна, допоміжна), інформаційні ресурси в Інтернеті

Основна література.

1. Струков В.М., Струкова В.Є. Основи алгоритмізації. Навчальний посібник. Харків: видавництво «Східно-регіональний центр гуманітарно-освітніх ініціатив», 2012р. 136 с.

2. Основи програмування на мові C#. Консольне програмування : навч. посіб. / МВС України, Харків. нац. ун-т внутр. справ ; за заг. ред. канд. техн. наук, доц. В. М. Струкова ; [В. М. Струков, Ю.В. Гнусов та ін.]. Харків :ХНУВС, 2023. 236 с.

3. Кублій Л. І. Алгоритмізація та програмування. Практикум. Електронне мережне навчальне видання. Київ: КПІ ім. Ігоря Сікорського, 2019. 209 с. URL: <https://ela.kpi.ua/handle/123456789/28216>

4. Мова програмування C#. Теорія і практика URL: https://www.bestprog.net/uk/sitemap_ua/c-3/

5. Романов В. В., Просянкін-Жарова Т. І., Безносик О. Ю. Алгоритмізація та програмування. Частина 1. Базові концепції програмування. Лабораторний практикум. Навчальний посібник. Електронне мережне навчальне видання. Київ: КПІ ім. Ігоря Сікорського, 2022. 151 с. URL: <https://ela.kpi.ua/handle/123456789/56436>

6. Рудий Т. В., Паранчук Я. С., Сенік В. В. Алгоритмізація та програмування. Частина 1. Структурне програмування : навчальний посібник. Львів : Львівський державний університет внутрішніх справ, 2023. 240 с.

7. Трофименко О.Г., Прокоп Ю.В., Логінова Н.І., Задерейко О.В. C++. Алгоритмізація та програмування : підручник. 2-ге вид. перероб. і доповн. Одеса: Фенікс, 2019. 477 с.

8. Dennis Yurichev. Reverse Engineering for Beginners. URL: <https://beginners.re/>

Допоміжна література.

1. Коваль В.С., Струбицький П.Р. Алгоритми і структури даних. Навчальний посібник. Тернопіль: ФОП Шпак В. Б., 2017 р. – 74 с.
2. Joseph Albahari, Ben Albahari. C# 5.0 in a NUTSHELL. Fifth Edition. – O'Reilly Media, Inc., 2012. – 1066 p.
3. Ron Fridman. Decoding Greatness: How the Best in the World Reverse. - Simon & Schuster, 2021. – 288 p.

Інформаційні ресурси в інтернеті.

4. <https://docs.microsoft.com/en-us/dotnet/csharp/>.
5. <https://metanit.com/sharp/tutorial/1.1.php>
6. <https://termin.in.ua/zvorotnyy-inzhynirynh-revers-inzhynirynh/>