

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ПЕРВОМАЙСЬКИЙ ІНДУСТРІАЛЬНО-ПЕДАГОГІЧНИЙ ФАХОВИЙ КОЛЕДЖ  
(ППФК)  
Циклова комісія професійної освіти

ЗАТВЕРДЖЕНО  
Наказ ППФКу  
від 30.08.2024 № 87

**МОВИ І ТЕХНОЛОГІЇ ПРОГРАМУВАННЯ**

**Програма навчальної дисципліни**

підготовки	<i>фахових молодших бакалаврів</i>
галузі знань	<i>01 Освіта/Педагогіка</i>
спеціальності	<i>015 Професійна освіта (Цифрові технології)</i>
спеціалізації	<i>015.39 Цифрові технології</i>
освітньо-професійної програми	<i>Професійна освіта (Цифрові технології)</i>

Програма навчальної дисципліни «*Мови і технології програмування*» освітньо-професійної програми «*Професійна освіта (Цифрові технології)*» (2024) для спеціальності 015 *Професійна освіта (за спеціалізаціями)*, спеціалізація 015.39 *Цифрові технології*, для здобувачів освіти I курсу денної форми навчання.

Розробник: *Максименко Володимир Геннадійович*, викладач комп'ютерних дисциплін, спеціаліст I категорії.

Програма розглянута і схвалена на засіданні циклової комісії професійної освіти 28.08.2024, протокол № 1 та схвалена на засіданні методичної ради 28.08.2024, протокол № 1.

### 1. Опис навчальної дисципліни (освітнього компоненту)

Найменування показників	Характеристика навчальної дисципліни	
<b>Форма навчання</b>	денна	
<b>Рік підготовки</b>	I (перший)	
<b>Семестр</b>	1-й	2-й
<b>Мова навчання</b>	українська	
<b>Статус дисципліни</b>	обов'язкова	
<b>Інформаційний обсяг навчальної дисципліни</b>		
<b>кількість кредитів ECTS</b>	6	
<b>загальна кількість годин, в т.ч.:</b>	180 год.	
– аудиторних	102 год.	
– самостійної роботи здобувачів освіти	48 год.	
– підготовка до екзамену	30 год.	
<b>Модулів</b>	2	
<b>Змістових модулів</b>	4	
<b>Лекції</b>	38 год.	18 год.
<b>Семінарські</b>	—	—
<b>Практичні</b>	—	—
<b>Лабораторні</b>	26 год.	20 год.
<b>Самостійна робота</b>	28 год.	20 год.
<b>Індивідуальні завдання</b>	—	
<b>Вид контролю</b>	екзамен	

## 2. Предмет, мета та завдання навчальної дисципліни

**Предметом** вивчення навчальної дисципліни «Мови і технології програмування» є загальні принципи програмування та технології, пов'язані з розробкою програмного забезпечення; способи та прийоми створення програм мовою C#.

**Метою викладання** навчальної дисципліни «Мови і технології програмування» є засвоєння здобувачами освіти основних концепцій, принципів та понять сучасного програмування, оволодіння методикою розробки програм засобами мови C# із застосуванням технології об'єктно-орієнтованого програмування.

**Основними завданнями** навчальної дисципліни «Мови і технології програмування» є:

- знання та розуміння сутності сучасних парадигм програмування;
- вивчення мови програмування C#;
- засвоєння основних концепцій програмування засобами мови C#;
- формування навичок проєктування, розробки та налагодження програм у середовищі програмування Microsoft Visual Studio.

**Передумови вивчення навчальної дисципліни.** Для успішного освоєння дисципліни «Мови і технології програмування» необхідними є отримані здобувачами освіти знання та вміння з предмету «Основи алгоритмізації та програмування», який вивчався при здобутті робітничої професії «Оператор з обробки інформації та програмного забезпечення».

## 3. Компетентності та заплановані результати навчання

Компетентності, які мають бути сформовані в результаті опанування освітнього компоненту (дисципліни), та очікувані результати навчання:

<b>Компетентності</b>	
<b>Загальні компетентності (ЗК)</b>	ЗК4 Здатність використовувати інформаційні та комунікаційні технології. ЗК5 Знання та розуміння предметної області та розуміння професійної діяльності. ЗК8 Здатність застосовувати знання у практичних ситуаціях.
<b>Спеціальні компетентності (СК)</b>	СК5 Здатність застосовувати інноваційні педагогічні та цифрові технології, інформаційне та програмне забезпечення для вирішення професійних завдань у галузі цифрових технологій. СК9 Здатність розв'язувати типові спеціалізовані задачі, пов'язані із виконанням необхідних розрахунків, конструюванням технічних об'єктів у галузі цифрових технологій. СК15 Здатність використовувати цифрові технології з метою розробки програмних продуктів для професійної діяльності майстра виробничого навчання.
<b>Зміст підготовки здобувачів фахової передвищої освіти, сформульований у термінах результатів навчання (РН)</b>	
РН23 Створювати програмні продукти із застосуванням сучасних технологій програмування.	

*У результаті вивчення навчальної дисципліни здобувачі освіти повинні:*

**знати:**

- основні принципи програмування;
- об'єктно-орієнтовану методика аналізу і розробки програмних систем;
- синтаксис і семантику базових конструкцій мови C#;
- способи та особливості організації структур даних засобами мови C#;
- засоби мови програмування C# для реалізації об'єктно-орієнтованих алгоритмів;
- правила визначення та використання класів у мові C#;
- інтерфейс середовища програмування Microsoft Visual Studio;

**вміти:**

- оперувати термінологією дисципліни;
- застосовувати теоретичні положення курсу до вирішення задач;
- створювати, редагувати, запускати, тестувати, налагоджувати та компілювати програми засобами середовища програмування Microsoft Visual Studio;
- реалізовувати типові алгоритми засобами мови C# у середовищі програмування Microsoft Visual Studio;
- визначати класи, розробляти методи та алгоритми, на основі яких створювати програми розв'язання задач із використанням об'єктів.

## 4. Структура навчальної дисципліни

Назви змістових модулів і тем	Кількість годин			
	Усього	у тому числі		
		лекцій	лабора-торних	самоств. роботи
<b>МОДУЛЬ 1 (1-й семестр)</b>				
<b>Змістовий модуль 1. Основи мови C#</b>				
<i>Тема 1.</i> Програмування. Базові поняття	2	2	–	–
<i>Тема 2.</i> Основи алгоритмізації	2	2	–	–
<i>Тема 3.</i> Вступ до основ програмування на платформі NET	4	2	–	2
<i>Тема 4.</i> Структура програми на C#	4	2	2	–
<i>Тема 5.</i> Основні елементи мови C#	6	2	2	2
<i>Тема 6.</i> Вирази та операції в мові C#	6	2	2	2
<i>Тема 7.</i> Перетворення типів	4	2	–	2
<i>Тема 8.</i> Реалізація алгоритмів із розгалуженням	6	2	2	2
<i>Тема 9.</i> Реалізація циклічних алгоритмів	4	2	2	–
<i>Тема 10.</i> Структурний підхід до розробки алгоритмів. Функції	6	2	2	2
<b>Разом за змістовим модулем 1</b>	<b>44</b>	<b>20</b>	<b>12</b>	<b>12</b>
<i>Тематичне оцінювання</i>				
<b>Змістовий модуль 2. Масиви і методи</b>				
<i>Тема 11.</i> Огляд масивів. Одновимірні масиви.	6	2	2	2
<i>Тема 12.</i> Двовимірні та багатовимірні масиви	6	2	2	2
<i>Тема 13.</i> Ступінчасті (зубчасті) масиви	6	2	2	2
<i>Тема 14.</i> Тривимірні масиви - багатовимірні масиви будь-якої мірності	6	2	2	2
<i>Тема 15.</i> Операції з масивами	6	2	2	2
<b>Разом за змістовим модулем 2</b>	<b>30</b>	<b>10</b>	<b>10</b>	<b>10</b>
<i>Тематичне оцінювання</i>				
<b>Змістовий модуль 3. Використання концепцій ООП при розробці додатків на мові C#</b>				
<i>Тема 16.</i> Вступ до об'єктно-орієнтованого програмування	2	2	–	–
<i>Тема 17.</i> Класи та об'єкти в C#	6	2	2	2
<i>Тема 18.</i> Спадкування та поліморфізм	6	2	2	2
<i>Тема 19.</i> Обробка помилок та виняткові ситуації	4	2	–	2
<b>Разом за змістовим модулем 3</b>	<b>16</b>	<b>8</b>	<b>4</b>	<b>6</b>
<i>Тематичне оцінювання</i>				
<b>Разом за модулем 1</b>	<b>92</b>	<b>38</b>	<b>26</b>	<b>28</b>

<b>МОДУЛЬ 2 (2-й семестр)</b>				
<b>Змістовий модуль 4. Використання концепцій ООП для розробки додатків з графічним інтерфейсом користувача мовою С#. Основи використання технології Windows Forms</b>				
<i>Тема 20.</i> Вступ до технології Windows Forms та розробки інтерфейсу	6	2	–	2
<i>Тема 21.</i> Основи використання елементів управління	6	2	2	2
<i>Тема 22.</i> Робота з подіями та обробка подій	6	2	2	2
<i>Тема 23.</i> Управління формами та вікнами	6	2	2	2
<i>Тема 24.</i> Організація макету і розміщення елементів	6	2	2	2
<i>Тема 25.</i> Робота з текстом та графікою	6	2	2	2
<i>Тема 26.</i> Робота зі списками та таблицями	6	2	2	2
<i>Тема 27.</i> Діалогові вікна та робота з файлами	6	2	2	2
<i>Тема 28.</i> MDI та SDI-додатки	6	2	2	2
<i>Тема 29.</i> Мережеві додатки на С#	6	2	2	2
<b>Разом за змістовим модулем 4</b>	<b>58</b>	<b>20</b>	<b>18</b>	<b>20</b>
<i>Тематичне оцінювання</i>				
<b>Підготовка до екзамену</b>	<b>30</b>			<b>30</b>
<b>Разом за модулем 2</b>	<b>88</b>	<b>20</b>	<b>18</b>	<b>50</b>
<b>Разом</b>	<b>180</b>	<b>58</b>	<b>44</b>	<b>78</b>

## 5. Програма навчальної дисципліни

### МОДУЛЬ 1 (1-й семестр)

#### Змістовий модуль 1. Основи мови С#

##### *Тема 1. Програмування. Базові поняття.*

Поняття програми та алгоритму, мова програмування, базові концепції програмування, змінні та типи даних, оператори, умовні конструкції, цикли, функції (методи), основні етапи розробки програм, аналіз вимог, проектування, кодування, тестування, експлуатація та підтримка, інструменти програмування, компілятори, інтегровані середовища розробки (IDE).

##### *Тема 2. Основи алгоритмізації*

Алгоритм як послідовність кроків для визначення способів вирішення конкретної задачі. Алгоритмічні конструкції як базові елементи для створення алгоритмів. Графічні схеми алгоритмів як спосіб графічного подання алгоритмів. Мови програмування як формальні мови, які використовуються для написання програм. Середовища програмування як програмні засоби для розробки, тестування і відлагоджування програм. Етапи розв'язування задач на комп'ютері.

##### *Тема 3. Вступ до основ програмування на платформі .NET*

CLR як частина платформи .NET. Бібліотеки класів .NET. Поняття про С# як мову програмування з використанням об'єктно-орієнтованого підходу. Призначення основних просторів імен бібліотеки класів платформи Microsoft .Net.

#### **Тема 4. Структура програми на C#**

Основні компоненти програми на C#. Простір імен як контейнер для групи пов'язаних класів. Клас — основний будівельний блок програми, який описує структуру об'єкта, включаючи його властивості та методи. Метод як блок коду, який виконує певну дію або повертає результат. Поля і властивості. Поле Main() як головний метод, з якого починається виконання програми. Відображення результатів: Виведення та введення даних.

#### **Тема 5. Основні елементи мови C#**

Визначення різних видів змінних та об'єктів у програмах за допомогою вбудованих типів даних. Присвоєння значення та використання змінної. Приведення типів використовується, як конвертація одного типу даних в інший. Літерали(константи) як фіксовані значення в коді програми

#### **Тема 6. Вирази та операції в мові C#**

Вирази та операції в мові C# як ключові елементи в програмуванні, обчисленні значень та взаємодії з даними в програмах. Вирази, як комбінації операторів, операндів та інших елементів, які виконують обчислення значень. Арифметичні, логічні, тернарний, порозрядні, порівняння та інші операції, при застосованні до різних типів даних. Присвоювання значення змінним за допомогою операторів присвоювання. Пріоритет операцій.

#### **Тема 7. Перетворення типів**

Перетворення типів як процес зміни типу даних об'єкта або значення з одного типу в інший. Явна та неявна конвертація. Забезпечення правильної обробки даних у різних випадках взаємодії між об'єктами різних типів. Різні методи та властивості для конвертації між рядками та числовими типами даних, як забезпечення коректної обробки введених користувачем даних та їх подальшої обробки в програмі.

#### **Тема 8. Реалізація алгоритмів із розгалуженням у C#**

Розгалуження як можливість виконання різних блоків коду в залежності від умови. Умовний оператор - конструкція, яка дозволяє виконувати певний блок коду, якщо задана умова виявиться істинною. Складений оператор, як об'єднання кількох інструкцій в один блок. Оператор вибору switch - спосіб заміни послідовності умовних операторів, який дозволяє вибирати один з кількох можливих варіантів виконання коду в залежності від значення виразу. Тернарний оператор, як короткий спосіб запису умовного оператора, що дозволяє визначити значення змінної на основі умови.

#### **Тема 9. Реалізація циклічних алгоритмів у C#**

Цикл як конструкція, що дозволяє повторювати виконання певного блоку коду певну кількість разів. Цикл з передумовою (while та do-while), як приклад використання блоку коду, доки визначена умова є істинною. Цикл з післяумовою (for), як приклад виконання блоку коду певну кількість разів, визначену перед початком циклу.

#### **Тема 10. Структурний підхід до розробки алгоритмів. Функції**

Структурний підхід до розробки алгоритмів, як організація програмного коду у вигляді структур, які складаються з блоків коду, що виконують конкретні функції у мові програмування C#. Функції(методи), як основні елементи структурного підходу для групування логічно пов'язаних операцій.

### **Змістовий модуль 2. Масиви у мові C#**

#### **Тема 11. Масиви**

Масив, як набір змінних одного типу. Ініціалізація масиву – присвоєння початкових значень елементам масиву. Доступ до елементів за допомогою індексів. Довжина масиву – кількість елементів. Ітерація через масив - використання циклів. Операції з масивами – сортування, пошук мінімуму або максимуму, зміну значень тощо.

**Тема 12. Двовимірні та багатовимірні масиви**

Двовимірний масив як список одновимірних масивів. Багатовимірний масив, як загальний термін, що включає двовимірні масиви та масиви вищих розмірностей. Застосування двовимірних та багатовимірних масивів в представленні матриць, таблиць, графіків, обробки зображень тощо.

**Тема 13. Ступінчасті (зубчасті) масиви**

Ступінчасті масиви, як потужний інструмент у мові С#. Загальний формат оголошення ступінчастого масиву. Ініціалізація ступінчастого масиву. Доступ до окремих елементів. Застосування ступінчастих масивів

**Тема 14. Тривимірні масиви – багатовимірні масиви будь-якої мірності**

Тривимірні масиви, як потужний інструмент у мові С#, для роботи з даними більш вимірюваного характеру. Загальна форма оголошення тривимірного масиву.

**Тема 15. Операції з масивами**

Методи для копіювання масивів, пошуку елементів, сортування масивів, злиття масивів, обрізання масивів. Застосування лінійних операцій над масивами.

### **Змістовий модуль 3. Використання головних концепцій ООП при розробці додатків мовою С#**

**Тема 16. Вступ до об'єктно-орієнтованого програмування**

Об'єктно-орієнтоване програмування (ООП) як одна з парадигм програмування. Визначення об'єктно-орієнтованого програмування (ООП). Значення ООП у розробці сучасного програмного забезпечення. Оновні поняття, принципи ООП, класи та об'єкти, інкапсуляція, поліморфізм та спадкування

**Тема 17. Класи та об'єкти, співвідношення між ними**

Клас - шаблон, за яким визначається форма об'єкту. Об'єкти, як екземпляри класу. Поля класу, як змінні, які містять дані, що описують клас. Методи - функції, які виконують дії з цими даними. Типи полів та методів класу. Використання елементів класу за межами класу

**Тема 18. Спадкування, інкапсуляція поліморфізм**

Спадкування, інкапсуляція та поліморфізм як основні концепції об'єктно-орієнтованого програмування. Створення нового класу, на основі існуючого, за допомогою спадкування. Інкапсуляція як приховування деталей реалізації класу від користувача. Поліморфізм - використання об'єктів по-різному. Переваги поліморфізму в розробці програмного забезпечення. Використання абстрактних класів та інтерфейсів для досягнення поліморфізму

**Тема 19. Обробка помилок та виняткові ситуації**

Неминучість помилок при написанні програми. Обробка помилок та виняткових ситуацій як важливі концепції у програмуванні. Винятки як тип помилки, який відбувається під час виконання програми. Перевірка помилок, валідація введення та логування помилок, як різні концепції, пов'язані з обробкою помилок.

## **МОДУЛЬ 2 (2-й семестр)**

### **Змістовий модуль 4. Використання концепцій ООП для розробки додатків з графічним інтерфейсом користувача мовою С#. Основи використання технології Windows Forms**

**Тема 20. Вступ до технології Windows Forms та розробки інтерфейсу**

Windows Forms як технологія, що дозволяє розробляти графічний інтерфейс користувача для програм на мові С#. Створення інтерфейсу за допомогою різноманітних

елементів керування. Форма як вікно, на якому можна розміщувати елементи керування. Візуальний редактор для додавання елементів керування на форму.

**Тема 21. Основи використання елементів управління**

Елементи управління як графічні об'єкти, які використовуються для взаємодії з користувачем в програмах на мові С#. Визначення зовнішнього виду елементів управління за допомогою властивостей.

**Тема 22. Робота з подіями та обробка подій**

Подія як взаємодія з елементом управління. Обробники подій як методи, які виконуються, коли відбувається певна подія. Додавання обробника подій до елемента управління.

**Тема 23. Управління формами та вікнами**

Вікно як контейнер, який містить одну або декілька форм. Створення нового проєкту та використання візуального редактора. Групування, переміщення, змінювання розмірів та налаштування властивостей елементів керування та вікон.

**Тема 24. Організація макету і розміщення елементів**

Макет як графічна схема підготовленого до друку матеріалу, що показує організацію елементів, розташування і взаємний зв'язок зображень, тексту і використовуваних шрифтів. Інструменти CSS Grid та Flexbox. CSS Grid як двовимірна система макету, яка дозволяє контролювати розміщення елементів в рядках та стовпцях. Flexbox як одновимірна система макету, яка дозволяє контролювати розміщення елементів в рядку або стовпці.

**Тема 25. Робота з текстом та графікою**

Інтерфейс графічних пристроїв (GDI) як інтерфейс Microsoft Windows для представлення графічних об'єктів і передачі їх на пристрої відображення, таких як монітори і принтери. Використання класу простору імен System.Drawing. Створення графічних об'єктів, малювання лінії, кривої, текст, використання шрифтів, кольорів, пензлів та інших елементів. Використання класу Image.

**Тема 26. Робота зі списками та таблицями**

Робота зі списками та таблицями як важлива концепція в розробці програм на мові С#. Використання маркірованого, нумерованого та багаторівневого списків. Використання класів простору імен System.Data та System.Data.SqlClient для створення таблиць, додавання до них даних, виконання запитів та інших операцій.

**Тема 27. Діалогові вікна та робота з файлами**

Використання діалогових вікон для взаємодії з користувачем в програмах на мові С#. Відображення діалогових вікон для отримання відповідей. Набір елементів керування діалогових вікон.

**Тема 28. MDI та SDI-додатки**

MDI (Multiple Document Interface) та SDI (Single Document Interface) як два різних способи організації графічного інтерфейсу користувача в програмах на мові С#.

**Тема 29. Мережеві додатки на С#**

С# як інструмент для створення мережевих додатків. Огляд платформи .NET для мережевого програмування. Робота з простором імен System.Net. Основи TCP/IP та моделі OSI. Клас System.Net.Sockets.Socket як основний клас для роботи з сокетом. Створення TCP-клієнтів та серверів та реалізація UDP-комунікацій. Клас HttpClient для відправки HTTP-запитів і обробки HTTP-відповідей. Поняття асинхронності в контексті мережевих операцій.

**Основні поняття дисципліни.** Алгоритм. Структура даних. Середовище програмування. Мова програмування. Змінна. Типи даних. Оператор. Вираз. Умовний вираз. Цикл. Функція. Процедура. Масив. Символ. ООП (Об'єктно-Орієнтоване

Програмування). Клас. Об'єкт. Метод. Абстрація. Інкапсуляція. Наслідування. Поліморфізм. Виняток. Помилка. Додаток. Діалогове вікно. Форма. Елемент управління. Подія. Вікно.

## 6. Теми лекцій

№ з/п	Назва теми	Кількість годин
	<b>ЗМ 1. Основи мови С#</b>	<b>20</b>
1.	Програмування. Базові поняття.	2
2.	Основи алгоритмізації	2
3.	Вступ до основ програмування на платформі .NET	2
4.	Структура програми на С#	2
5.	Основні елементи мови С#	2
6.	Вирази та операції в мові С#	2
7.	Перетворення типів	2
8.	Реалізація алгоритмів із розгалуженням	2
9.	Реалізація циклічних алгоритмів	2
10.	Структурний підхід до розробки алгоритмів. Функції	2
	<b>ЗМ 2. Масиви</b>	<b>10</b>
11.	Огляд масивів. Одновимірні масиви.	2
12.	Двовимірні та багатовимірні масиви	2
13.	Ступінчасті (зубчасті) масиви	2
14.	Тривимірні масиви — багатовимірні масиви будь-якої мірності	2
15.	Операції з масивами	2
	<b>ЗМ 3. Використання концепцій ООП при розробці додатків мовою С#</b>	<b>8</b>
16.	Вступ до об'єктно-орієнтованого програмування	2
17.	Класи та об'єкти в С#	2
18.	Спадкування та поліморфізм	2
19.	Обробка помилок та виняткові ситуації	2
	<b>ЗМ 4. Використання концепцій ООП для розробки додатків з графічним інтерфейсом користувача мовою С#. Основи використання технології Windows Forms</b>	<b>20</b>
20.	Вступ до технології Windows Forms та розробки інтерфейсу	2
21.	Основи використання елементів управління	2
22.	Робота з подіями та обробка подій	2
23.	Управління формами та вікнами	2
24.	Організація макету і розміщення елементів	2
25.	Робота з текстом та графікою	2
26.	Робота зі списками та таблицями	2
27.	Діалогові вікна та робота з файлами	2
28.	MDI та SDI-додатки	2
29.	Мережеві додатки на С#	2
	<b>Усього годин</b>	<b>58</b>

## 7. Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
	<b>ЗМ 1. Основи мови С#</b>	<b>12</b>
1.	<i>Лабораторна робота № 1.</i> Знайомство з середовищем розробки програм Microsoft Visual Studio та базовими елементами мови С#	2
2.	<i>Лабораторна робота № 2.</i> Створення простої С#-програми, перетворення типів	2
3.	<i>Лабораторна робота № 3.</i> Оголошення та ініціалізація змінних і констант у С#	2
4.	<i>Лабораторна робота № 4.</i> Розробка розгалужених консольних програм засобами мови С#	2
5.	<i>Лабораторна робота № 5.</i> Розробка циклічних консольних програм засобами мови С#	2
6.	<i>Лабораторна робота № 6.</i> Розробка, виклик та застосування користувацьких функцій засобами мови С#	2
	<b>ЗМ 2. Масиви</b>	<b>10</b>
7.	<i>Лабораторна робота № 7.</i> Виведення та перебір масиву	2
8.	<i>Лабораторна робота № 8.</i> Заповнення масиву з клавіатури	2
9.	<i>Лабораторна робота № 9.</i> Виведення масиву в зворотньому порядку	2
10.	<i>Лабораторна робота № 10.</i> Пошук парних чисел у масиві	2
11.	<i>Лабораторна робота № 11.</i> Пошук найменшого елемента масиву	2
	<b>ЗМ 3. Використання головних концепцій ООП при розробці додатків мовою С#</b>	<b>4</b>
12.	<i>Лабораторна робота № 12.</i> Розробка класу та його структури	2
13.	<i>Лабораторна робота № 13.</i> Розробка класу та його об'єктів	2
	<b>ЗМ 4. Використання концепцій ООП щодо розробки додатків з графічним інтерфейсом користувача мовою С#. Основи використання технології Windows Forms.</b>	<b>16</b>
14.	<i>Лабораторна робота № 14.</i> Розробка простого додатку Window Forms у середовищі MS Visual Studio	2
15.	<i>Лабораторна робота № 15.</i> Розробка додатку Window Forms із застосуванням вікон повідомлень	2
16.	<i>Лабораторна робота № 16.</i> Розробка додатку Window Forms із використанням базових компонентів	2
17.	<i>Лабораторна робота № 17.</i> Розробка додатку Window Forms для виконання розрахунків	2
18.	<i>Лабораторна робота № 18.</i> Розробка додатку Window Forms для конвертування одиниць вимірювання	2
19.	<i>Лабораторна робота № 19.</i> Розробка додатку Window Forms для роботи з двовимірними масивами (матрицями)	2
20.	<i>Лабораторна робота № 20.</i> Розробка додатку Window Forms для роботи з діалоговими вікнами та файлами	2
21.	<i>Лабораторна робота № 21.</i> Розробка MDI та SDI додатків	2
22.	<i>Лабораторна робота № 22.</i> Завантаження із Інтернет	
	<b>Усього годин</b>	<b>44</b>

## 8. Самостійна робота

№ з/п	Назва теми	Кількість годин
	<b>ЗМ 1. Основи мови С#</b>	<b>12</b>
1.	<i>Тема 3. Призначення основних просторів імен бібліотеки класів платформи Microsoft .Net</i>	2
2.	<i>Тема 5. Літерали (константи) мови С#</i>	2
3.	<i>Тема 6. Пріоритет операцій у С#</i>	2
4.	<i>Тема 7. Різні методи та властивості для конвертації між рядками та числовими типами даних, як забезпечення коректної обробки введених користувачем даних та їх подальшої обробки в програмі.</i>	2
5.	<i>Тема 8. Тернарний оператор</i>	2
6.	<i>Тема 10. Функції(методи), як основні елементи структурного підходу для групування логічно пов'язаних операцій.</i>	2
	<b>ЗМ 2. Масиви у мові С#</b>	<b>10</b>
7.	<i>Тема 11. Операції з масивами</i>	2
8.	<i>Тема 12. Застосування двовимірних та багатовимірних масивів</i>	2
9.	<i>Тема 13. Застосування ступінчастих масивів</i>	2
10.	<i>Тема 14. Застосування багатовимірних масивів</i>	2
11.	<i>Тема 15. Застосування лінійних операцій над масивами</i>	2
	<b>ЗМ 3. Використання головних концепцій ООП при розробці додатків на мові С#</b>	<b>6</b>
12.	<i>Тема 17. Використання елементів класу за межами класу</i>	2
13.	<i>Тема 18. Використання абстрактних класів та інтерфейсів для досягнення поліморфізму</i>	2
14.	<i>Тема 19. Перевірка помилок, валідація введення та логування помилок, як різні концепції, пов'язані з обробкою помилок.</i>	2
	<b>ЗМ 4. Використання концепцій ООП щодо розробки додатків з графічним інтерфейсом користувача на мові С#. Основи використання технології Windows Forms.</b>	<b>20</b>
15.	<i>Тема 20. Міжплатформена сумісність</i>	2
16.	<i>Тема 21. Графічний інтерфейс користувача (GUI)</i>	2
17.	<i>Тема 22. Передача додаткових параметрів в обробник подій</i>	2
18.	<i>Тема 23. Управління життєвим циклом форм</i>	2
19.	<i>Тема 24. Менеджери розміщення елементів управління</i>	2
20.	<i>Тема 25. Основні типи простору імен System.Drawing</i>	2
21.	<i>Тема 26. Маніпулювання даними за допомогою DataGridView</i>	2
22.	<i>Тема 27. Класи з простору імен System.IO, що використовуються для роботи з файлами і директоріями</i>	2
23.	<i>Тема 28. Переваги та недоліки SDI в контексті дизайну і взаємодії з користувачем</i>	2
24.	<i>Тема 29. Поняття асинхронності в контексті мережевих операцій.</i>	2
25.	<b>Підготовка до екзамену</b>	<b>30</b>
	<b>Усього годин</b>	<b>78</b>

## 9. Засоби та форми контролю

Оцінювання навчальної діяльності здобувачів освіти здійснюється шляхом поточного контролю, тематичних та семестрових атестацій.

Поточне оцінювання проводиться у формі фронтального та індивідуального опитування, тестування, перевірки конспектів та завдань для самостійної роботи, виконання лабораторних робіт та їх захисту.

Передбачено 4 тематичні атестації відповідно до змістових модулів програми.

Підсумковий контроль — екзамен, який проводиться у формі комп'ютерного тестування (теоретична частина) та розробки програми мовою С# (практична частина).

## 10. Критерії оцінювання результатів навчання

Оцінювання результатів навчальної діяльності здобувачів освіти здійснюється за 4-бальною шкалою оцінювання за наведеними далі критеріями.

**Відмінно** — студент вільно володіє основними концепціями програмування на С# (класами, об'єктами, властивостями, методами, наслідуванням тощо). Може розробляти власні алгоритми, самостійно вирішує складні завдання. Аргументує свої рішення, вміє застосовувати альтернативні підходи.

При виконанні лабораторних робіт завдання виконує в повному обсязі, усвідомлено та без помилок, у встановлені терміни; реалізує оптимальні рішення. Використовує як типові, так і авторські алгоритми для вирішення завдань. Задачі виконує у повному обсязі та у встановлені терміни.

Виконав 90-100 % тестових завдань

Виконав 100 % обсягу самостійної роботи, проявив ініціативу в додаткових завданнях (створення проектів, робота з бібліотеками С#, використання сторонніх API).

**Добре** — розуміє основні поняття мови С#, володіє базовими алгоритмами, хоча допускає несуттєві помилки або неточності. Може аналізувати код і робити висновки, але інколи потребує допомоги у розробці складних алгоритмів.

При виконанні лабораторних робіт з програмування мовою С# допускає несуттєві помилки, які може самостійно виправити, в деяких випадках потребує консультацій викладача; роботи виконуються за типовими алгоритмами дій, у встановлені терміни, у повному обсязі. Звіти оформлені якісно та грамотно, вчасно здані.

При тестуванні правильно виконує 70-89 % завдань.

Виконав 100 % обсягу самостійної роботи, але без глибоких досліджень або додаткових ініціатив.

**Задовільно** — володіє лише основними поняттями, розуміння концепцій С# поверхове. Відповіді на запитання неповні, часто містять грубі помилки.

При виконанні лабораторних робіт допускає помилки, які може виправити частково, потребує консультацій викладача. Роботи виконуються за типовими нескладними алгоритмами дій, у неповному обсязі або з порушенням встановлених термінів.

При тестуванні правильно виконує 50-69 % завдань.

Виконав не менше 70 % обсягу самостійної роботи.

**Незадовільно** — не розуміє основних принципів програмування на С#, відповідає з грубими помилками, не здатен пояснити ключові поняття.

При виконанні лабораторних робіт завдання виконує тільки за найпростішими типовими алгоритмами дій, припускає суттєві помилки, не може працювати без постійних консультацій викладача; суттєво порушує терміни виконання або робота не виконується взагалі. Виконав менше 50 % обсягу самостійної роботи.

При тестуванні правильно виконує 0-49 % завдань.

**Таблиця відповідності результатів контролю знань здобувачів освіти, рівень знань яких оцінюється за 4-бальною шкалою, у системі ECTS**

За 4-бальною шкалою	Оцінка в ECTS	Критерії оцінювання
5 (відмінно)	A	Здобувач освіти виявляє особливі творчі здібності, вміє самостійно здобувати знання, без допомоги викладача знаходить та опрацьовує необхідну інформацію, вміє використовувати набуті знання і вміння для прийняття рішень у нестандартних ситуаціях, переконливо аргументує відповіді, самостійно розкриває власні обдарування і нахили
4 (добре)	B	Здобувач освіти вільно володіє вивченим обсягом матеріалу, застосовує його на практиці, вільно розв'язує вправи і задачі у стандартних ситуаціях, самостійно виправляє допущені помилки, кількість яких незначна
	C	Здобувач освіти вміє зіставляти, узагальнювати, систематизувати інформацію під керівництвом викладача; в цілому самостійно застосовувати її на практиці; контролювати власну діяльність; виправляти помилки, серед яких є суттєві, добирати аргументи для підтвердження думок
3 (задовільно)	D	Здобувач освіти відтворює значну частину теоретичного матеріалу, виявляє знання і розуміння основних положень; з допомогою викладача може аналізувати навчальний матеріал, виправляти помилки, серед яких є значна кількість суттєвих
	E	Здобувач освіти володіє навчальним матеріалом на рівні, вищому за початковий, значну частину його відтворює на репродуктивному рівні
2 (незадовільно)	FX	Здобувач освіти володіє матеріалом на рівні окремих фрагментів, що становлять незначну частину навчального матеріалу
	F	Здобувач освіти володіє матеріалом на рівні елементарного розпізнання і відтворення окремих фактів, елементів, об'єктів

### 11. Методичне забезпечення

1. Програма навчальної дисципліни.
2. Силабус навчальної дисципліни.
3. Конспекти лекцій.
4. Інструктивно-методичні матеріали до проведення лабораторних занять.
5. Тестові завдання.
6. Мультимедійні презентації.
7. Методичні рекомендації до самостійного вивчення.
8. Екзаменаційні матеріали.
9. Програмне забезпечення: операційна система Windows, середовище розробки програмного забезпечення Microsoft Visual Studio.

## 10. Рекомендована література та ресурси інтернет

### Основна

1. Васильев А. Программирование на С# для начинающих. Основные сведения. Москва: Эксмо, 2018. 592 с.
2. Васильев А. Программирование на С# для начинающих. Особенности языка. – Москва : Эксмо, 2018. 528 с.
3. Евдокимов П. В. С# на примерах. СПб. : Наука и Техника, 2019. 320 с.
4. Кравець П. О. Об'єктно-орієнтоване програмування. Львів : Видавництво Львівської політехніки, 2012. 624 с.
5. Культин Н. Б. Основы программирования в Microsoft Visual С# 2010. СПб. : БХВ-Петербург, 2011. 368 с.
6. Павловская Т.А. С#. Программирование на языке высокого уровня. СПб. : Питер, 2014. 432 с.
7. Парфьонов Ю. Е., Федорченко В. М., Лосев М. Ю., Щербаков О. В. Об'єктно-орієнтоване програмування: конспект лекцій для студентів напряму підготовки «Комп'ютерні науки» всіх форм навчання. Харків : Вид. ХНЕУ, 2010. 312 с.
8. Програмування мовою С# 6.0. Навчальний посібник для технічних спеціальностей вищих навчальних закладів. Тернопіль : ТНТУ, 2016. 229 с.
9. Фленов М. Е. Библия С#. СПб. : БХВ-Петербург, 2016. 544 с.
10. Шилдт Г. Полный справочник по С#. Москва : Издательский дом «Вильямс», 2004. 752 с.

### Додаткова

1. Использование С#. Специальное издание. Москва : Издательский дом «Вильямс», 2002. 528 с.
2. Трэй Нэш. С# 2008: ускоренный курс для профессионалов. Москва : Издательский дом «Вильямс», 2008. 576 с.
3. Троелсен Э. С# и платформа NET. Библиотека программиста. СПб. : Питер, 2002. 800 с.
4. Прайс Д., Гандэрлой М. Visual С#.NET. Полное руководство.: Киев : ВЕК+, СПб. : КОРОНА Принт, Киев : НТИ, Москва : Энтроп, 2004. 960 с.
5. Секунов Н. Ю. Разработка приложений на С++ и С#. Библиотека программиста. СПб. : Питер, 2003. 608 с.
6. Майо Д. С#: Искусство программирования. Энциклопедия программиста. СПб. : ООО «ДиаСофтЮП», 2002. 656 с.
7. Байдачный С. С. .NET Framework. Секреты создания Windows-приложений. Москва : СОЛОН-Пресс, 2004. 496 с.
8. Шилдт Г. С#: учебный курс. СПб. : Питер; Киев : Издательская группа ВНУ, 2003. 512 с.
9. Петцольд Ч. Программирование для Microsoft Windows на С#. В 2-х томах / Пер. с англ. Москва : Издательско-торговый дом «Русская Редакция», 2002. 576 + 624 с.
10. Рамбо Дж., Якобсон А., Буч Г. UML: специальный справочник. СПб. : Питер, 2002. 656 с.

## Довідкова

1. ДСТУ 3008-2015. Документація. Звіти у сфері науки і техніки. Структура і правила оформлення. Київ : ДП «УкрНДНЦ», 2016. 26 с.
2. ДСТУ 4163-2003. Уніфікована система організаційно-розпорядчої документації. Вимоги до оформлювання документів. Київ : Держспоживстандарт України, 2003. 22 с.

## 11. Інформаційні ресурси

1. Курс «Основи програмування на С#». URL : [https://courses.prometheus.org.ua/courses/Microsoft/CS201/2016\\_T1/about](https://courses.prometheus.org.ua/courses/Microsoft/CS201/2016_T1/about)
2. Відеокурс «Повний курс С# для початківців». URL : <https://www.youtube.com/playlist?list=PLHK-VBs5Sk3oVTgnGqOWfmRHEYvXv8xAw>
3. Відеоуроки С#. Теорія та практика. URL : <http://surl.li/llyhb>