

Київський університет імені Бориса Грінченка  
Факультет інформаційних технологій та математики  
Кафедра інформаційної та кібернетичної безпеки  
імені професора Володимира Бурячка

«ЗАТВЕРДЖУЮ»  
Проректор з науково-методичної  
та навчальної роботи

«05»

Олексій ЖИЛЬЦОВ  
2022 р.



РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ  
«ТЕХНОЛОГІЇ БЕЗПЕЧНОГО ПРОГРАМУВАННЯ»

для студентів

спеціальності	125 Кібербезпека
освітнього рівня	першого (бакалаврського)
освітньої програми	125.00.01 Безпека інформаційних і комунікаційних систем

КИЇВСЬКИЙ УНІВЕРСИТЕТ  
ІМЕНІ БОРИСА ГРІНЧЕНКА  
Ідентифікаційний код 02136554  
Начальник відділу  
моніторингу якості освіти

Програма № 0138/22  
Лещенко  
(підпис) (прізвище, ім'я)

«    » 2022

2022 – 2023 навчальний рік

**Розробники:**

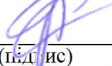
Спасітелєва Світлана Олексіївна, кандидат фізико-математичних наук, доцент, доцент кафедри інформаційної та кібернетичної безпеки імені професора Володимира Бурячка Факультету інформаційних технологій та математики Київського університету імені Бориса Грінченка.

**Викладач:**

Спасітелєва Світлана Олексіївна, кандидат фізико-математичних наук, доцент, доцент кафедри інформаційної та кібернетичної безпеки імені професора Володимира Бурячка Факультету інформаційних технологій та математики Київського університету імені Бориса Грінченка.


Робочу програму розглянуто і затверджено на засіданні кафедри інформаційної та кібернетичної безпеки імені професора Володимира Бурячка

Протокол від 01.09.2022 р. № 12

Завідувач кафедри \_\_\_\_\_  \_\_\_\_\_ Павло СКЛАДАННИЙ  
(підпис)


Робочу програму погоджено з гарантом освітньої програми (керівником освітньої програми 125.00.01 Безпека інформаційних і комунікаційних систем)

\_\_\_\_\_.\_\_\_\_. 2022 р.

Керівник освітньої програми \_\_\_\_\_  \_\_\_\_\_ Артем ПЛАТОНЕНКО  
(підпис)

Робочу програму перевірено

\_\_\_\_\_.\_\_\_\_. 2022 р.

Заступник декана \_\_\_\_\_  \_\_\_\_\_ Євген ІВАНІЧЕНКО  
(підпис)

**Пролонговано:**

на 20\_\_/20\_\_ н.р. \_\_\_\_\_ (\_\_\_\_\_) \_\_\_\_\_, «\_\_\_\_» \_\_\_\_\_ 20\_\_ р., протокол № \_\_\_\_  
(підпис) (ПІБ)

на 20\_\_/20\_\_ н.р. \_\_\_\_\_ (\_\_\_\_\_) \_\_\_\_\_, «\_\_\_\_» \_\_\_\_\_ 20\_\_ р., протокол № \_\_\_\_  
(підпис) (ПІБ)

на 20\_\_/20\_\_ н.р. \_\_\_\_\_ (\_\_\_\_\_) \_\_\_\_\_, «\_\_\_\_» \_\_\_\_\_ 20\_\_ р., протокол № \_\_\_\_  
(підпис) (ПІБ)

на 20\_\_/20\_\_ н.р. \_\_\_\_\_ (\_\_\_\_\_) \_\_\_\_\_, «\_\_\_\_» \_\_\_\_\_ 20\_\_ р., протокол № \_\_\_\_  
(підпис) (ПІБ)

## 1. Опис навчальної дисципліни

Найменування показників	Характеристика дисципліни за формами навчання		
	денна	заочна	
«Технології безпечного програмування»			
Вид дисципліни	обов'язкова		
Мова викладання, навчання та оцінювання	українська		
Загальний обсяг кредитів / годин	9 / 270		
Курс	1	2	
Семестр	2	3	
Кількість змістових модулів з розподілом:	7		
Обсяг кредитів	3	6	
Обсяг годин, в тому числі:	90	180	
Аудиторні	42	56	
Модульний контроль	6	8	
Семестровий контроль	-	60	
Самостійна робота	42	56	
Форма семестрового контролю	залік	екзамен	

## 2. Мета та завдання навчальної дисципліни

Робоча навчальна програма дисципліни «Технології безпечного програмування» є нормативним документом Київського університету імені Бориса Грінченка, який розроблено кафедрою інформаційної та кібернетичної безпеки імені професора Володимира Бурячка на основі освітньо-професійної програми підготовки здобувачів першого (бакалаврського) рівня відповідно до навчального плану спеціальності 125 Кібербезпека, освітньої програми 125.00.01 Безпека інформаційних і комунікаційних систем.

Робочу навчальну програму укладено згідно з вимогами Європейської кредитної трансферно-накопичувальної системи (ЄКТС) організації навчання.

Програма визначає: обсяги знань, який повинен опанувати здобувач першого (бакалаврського) рівня відповідно до вимог освітньо-кваліфікаційної характеристики; алгоритм вивчення навчального матеріалу дисципліни «Технології безпечного програмування»; необхідне методичне забезпечення; складові та технологію оцінювання навчальних досягнень студентів.

**Метою** викладання навчальної дисципліни «Технології безпечного програмування» є засвоєння студентами фундаментальних знань в області теорії і практики безпечного програмування із застосуванням процедурної та об'єктно-орієнтовної технологій програмування.

**Завдання** навчальної дисципліни полягає у формуванні теоретичних знань та практичних умінь у сфері розробки безпечного програмного забезпечення, побудови та аналізу алгоритмів з використанням методів створення ефективних програм на базі відповідних структур даних та алгоритмів їх обробки для систем інформаційної та кібернетичної безпеки та набуття таких компетентностей:

*Фахові компетентності спеціальності*

КФ-3: Здатність до використання програмних та програмно-апаратних комплексів засобів захисту інформації в інформаційно-телекомунікаційних (автоматизованих) системах.

## 3. Результати навчання за дисципліною

У результаті вивчення навчальної дисципліни «Технології безпечного програмування» студент повинен

**знати:**

- теорію та методи безпечного програмування для розв'язування прикладних задач та створення програмного забезпечення систем ІБ;
- етапи життєвого циклу розробки безпечного програмного забезпечення;
- методи розробки алгоритмів, лінійні та ієрархічні структури даних та алгоритмів їх обробки;
- основи процедурного та об'єктно-орієнтованого програмування із застосуванням принципів безпечного програмування;
- методи проектування та реалізації систем ІКБ;

#### **уміти:**

- застосовувати сучасні технології безпечного програмування в системах інформаційної та кібербезпеки;
- обирати відповідну технологію програмування, виконувати аналіз специфікації задач;
- виконувати аналіз програмного забезпечення з метою пошуку, ідентифікації, виявлення та усунення помилок програмування та вразливостей;
- використовувати фундаментальні знання з основ побудови та аналізу алгоритмів для розв'язку практичних задач;
- обирати методи зберігання та ефективні алгоритми обробки для відповідних структур даних для створення захищених програм;
- виконувати декомпозицію (структурування) алгоритму, розробляти структурні схеми програм із відображенням різноманітних зв'язків між компонентами та визначенням їх функцій;
- визначати класи та створювати об'єкти при застосуванні об'єктно-орієнтованих технологій програмування;
- виконувати модифікацію програм, додаючи, змінюючи або усуваючи окремі їх компоненти з метою зміни їх функцій або спроможності їх функціонування в нових технічних умовах;
- застосовувати прийоми проектування та створення власних захищених програм за допомогою IDE Visual C#;
- забезпечувати працездатність всієї програми, виконуючи індивідуальне та інтегральне тестування, а також застосовуючи засоби безпечного програмування;
- виконувати аналіз програмного коду.

Студенти мають досягти наступних **програмних результатів навчання:**

**ПР3-3:** забезпечувати процеси захисту інформаційно-телекомунікаційних (автоматизованих) систем шляхом встановлення та коректної експлуатації програмних та програмно-апаратних комплексів засобів захисту; забезпечувати функціонування спеціального програмного забезпечення, щодо захисту даних від руйнуючих програмних впливів, руйнуючих кодів в інформаційних, інформаційно-телекомунікаційних (автоматизованих) системах; виконувати розробку експлуатаційної документації на КЗЗ.

**ПР3-4:** вирішувати задачі супроводу (в.т. числі: огляд, тестування, підзвітність) системи управління доступом згідно принципів, критеріїв доступу та встановленої політики безпеки в інформаційних та інформаційно-телекомунікаційних (автоматизованих) системах; забезпечувати введення підзвітності системи управління доступом інформаційних ресурсів і процесів в ІТС.

**ПР3-5:** проектувати та реалізувати комплексні системи захисту інформації в АС організації (підприємства) відповідно до вимог нормативних документів системи технічного захисту інформації; - вирішувати задачі захисту потоків даних в інформаційних, інформаційно-телекомунікаційних (автоматизованих) системах.

**ПР3-6:** вирішувати задачі управління процесами забезпечення неперервності бізнесу з використанням процедур резервування програмного забезпечення та безпосередньо

інформаційних ресурсів; виконувати аналіз налаштувань елементів інформаційних систем та комунікаційного обладнання.

**ПРз-7:** вирішувати задачі супроводу та впровадження комплексних систем захисту інформації, а також протидії несанкціонованому доступу до ресурсів і процесів в інформаційних та інформаційно-телекомунікаційних (автоматизованих) системах.

**ПРз-8:** вирішувати задачі попередження та виявлення, ідентифікації, аналізу та реагування на інциденти в інформаційних, інформаційно-телекомунікаційних (автоматизованих) системах.

**ПРз-9:** забезпечувати функціонування системи управління інформаційною та/або кібербезпекою організації на основі керування інформаційними ризиками, здійснення процедур їх кількісного і якісного оцінки.

**ПРз-10:** аналізувати та визначати можливість застосування технологій, методів та засобів технічного захисту інформації; аналізувати та визначати можливість застосування технологій, методів та засобів криптографічного захисту інформації.

**ПРз-11:** забезпечувати процеси моніторингу доступу до ресурсів і процесів ІТС; забезпечувати конфігурування та функціонування систем моніторингу ресурсів та процесів в ІТС.

**ПРз-12:** виконувати впровадження та підтримку систем виявлення вторгнень та використовувати комплекси захисту для забезпечення необхідного рівня захищеності інформації в інформаційних, інформаційно-телекомунікаційних та автоматизованих системах; аналізувати та впроваджувати системи захисту від зловмисних програмних кодів.

#### 4. Структура навчальної дисципліни

Тематичний план для денної форми навчання

Назви змістових модулів і тем	Розподіл годин між видами робіт					
	Усього	Аудиторні:				Самостійна.
		Лекції	Практичні	Лабораторні	Індивідуальні	
<b>СЕМЕСТР 2</b>						
<b>Змістовий модуль 1. Основні принципи безпечного програмування</b>						
Тема 1. Основні поняття програмування	2	2				
Тема 2. Основні принципи безпечного програмування. Характеристика технології .Net Framework	4	2		2		
Тема 3. Данні програми. Безпека даних	8	2		2		4
Тема 4. Управління потоком виконання програми	20	2		8		10
Модульний контроль	2					
<b>Разом</b>	<b>36</b>	<b>8</b>		<b>12</b>		<b>14</b>
<b>Змістовий модуль 2. Структуровані типи даних</b>						
Тема 5. Лінійні структури збереження даних та основні алгоритми роботи з даними	24	2		8		14
Модульний контроль	2					
<b>Разом</b>	<b>26</b>	<b>2</b>		<b>8</b>		<b>14</b>

<b>Змістовий модуль 3. Технологія структурного процедурного програмування</b>						
Тема 6. Розбиття програми на підпрограми. Методи. Статичні методи	13	2		4		7
Тема 7. Способи передачі параметрів у методи	13	2		4		7
Модульний контроль	2					
<b>Разом</b>	<b>28</b>	<b>4</b>		<b>8</b>		<b>14</b>
<b>Семестровий контроль</b>	<b>-</b>					
<b>Усього годин за 2 семестр</b>	<b>90</b>	<b>14</b>		<b>28</b>		<b>42</b>
<b>СЕМЕСТР 3</b>						
<b>Змістовий модуль 4. Основи об'єктно-орієнтованого програмування</b>						
Тема 1. Принципи об'єктно-орієнтованого програмування. Класи та об'єкти.	10	4		6		
Тема 2. Успадкування. Похідні класи.	12	2		4		6
Тема 3. Технологія розробки безпечного об'єктно-орієнтованого програмного забезпечення	10	2				8
Модульний контроль	2					
<b>Разом</b>	<b>34</b>	<b>8</b>		<b>10</b>		<b>14</b>
<b>Змістовий модуль 5. Класи збереження даних</b>						
Тема 4. Класи для роботи з файлами	12	2		4		6
Тема 5. Класи колекцій	14	2		4		8
Модульний контроль	2					
<b>Разом</b>	<b>28</b>	<b>4</b>		<b>8</b>		<b>14</b>
<b>Змістовий модуль 6. Класи базових елементів графічного інтерфейсу користувача</b>						
Тема 6. Проектування та розробка основних компонент графічного інтерфейсу користувача. Бібліотека класів .Net Framework.	14	2		6		6
Тема 7. Класи стандартних елементів управління.	14	2		4		8
Модульний контроль	2					
<b>Разом</b>	<b>30</b>	<b>4</b>		<b>10</b>		<b>14</b>
<b>Змістовий модуль 7. Розробка додатків в інтегрованих середовищах. Побудова керованого та захищеного коду за допомогою Windows Forms, ADO.NET, WPF, ASP.NET</b>						
Тема 8. Засоби збереження даних. DataGridView, DB	16	4		4		8
Тема 9. Засоби для створення веб-додатків - ASP.NET	10	2		2		6
Модульний контроль	2					
<b>Разом</b>	<b>28</b>	<b>6</b>		<b>6</b>		<b>14</b>
<b>Семестровий контроль</b>	<b>30</b>					
<b>Курсова робота</b>	<b>30</b>					
<b>Усього годин за 3 семестр</b>	<b>180</b>	<b>22</b>		<b>34</b>		<b>56</b>

## 5. Програма навчальної дисципліни

### Змістовий модуль 1. Основні принципи безпечного програмування

Вступ. Предмет програмування. Огляд основних понять програмування. Програма та програмний продукт. Модулі. Структурний підхід в програмуванні. Процес розробки програм. Життєвий цикл програмного забезпечення (ПЗ). Методології програмування. Означення алгоритму. Способи опису алгоритму. Властивості алгоритму. Форми представлення алгоритму. Поняття безпечного програмування. Вразливості програмного коду. Властивості безпечного програмного забезпечення. Головні принципи розробки безпечного та надійного коду. Життєвий цикл розробки безпечного програмного забезпечення (Security Development Lifecycle, SDL) . Особливості SDL. Специфікація ПЗ (вимоги безпеки). Проектування ПЗ (моделювання загроз, аналіз атак, перевірка вхідних даних, границі довіри). Методи

проекування ПЗ: структурні методи, об'єктно-орієнтовані методи. Реалізація ПЗ (глибинний захист коду). Тестування та налагодження ПЗ (аудит коду, огляд області атак). Експлуатація та супроводження ПЗ. Методи аналізу коду. Правила написання безпечного коду .Net Framework.

Базові елементи мови на прикладі мови C#. Дані. Базові типи даних. Ідентифікатори. Змінні. Константи. Вирази. Основні операції. Пріоритет операцій. Консольне введення/виведення. Типи структур управління: послідовна структура, структура вибору, структура повторення. Оператори переходу – return, continue, break, goto. Оператори вибору – if, if/else, switch та оператори циклів (повторення) – for, while, do/while.

### **Змістовий модуль 2. Структуровані типи даних**

Похідні типи даних. Одновимірні масиви та символні рядки. Оголошення та ініціалізація масиву. Виконання операцій над елементами масиву. Багатовимірні масиви. Основні алгоритми роботи з числовими масивами: знаходження сум, добутків, максимального та мінімального елементів масиву. Лінійні структури даних: список, стек, черга. Основні алгоритми для роботи з даними: вставка, вилучення елементів, сортування, пошук, шифрування.

### **Змістовий модуль 3. Технологія структурного процедурного програмування**

Розбиття програми на підпрограми. Методи. Типи методів. Визначення методів. Область дії методу. Параметри методів. Формальні та фактичні параметри методів. Передача параметрів методу за значенням та за посиланням. Повернення значень із методу. Перевантаження імен методів. Простір імен. Створення та використання бібліотек методів.

### **Змістовий модуль 4. Основи об'єктно-орієнтованого програмування**

Абстрактні типи даних. Класи та об'єкти. Захист внутрішньої структури об'єкту. Конструктори та деструктори. Основні операції з об'єктом. Властивості. Перевантаження операторів та методів класу. Типи відношень між класами. Відношення успадкування. Правила створення похідних класів. Віртуальні методи. Абстрактні методи. Індикатори. Інтерфейси. Делегати. Базові принципи об'єктно-орієнтованого програмування: інкапсуляція, успадкування, поліморфізм. Моделі процесу створення ПЗ: каскадна модель, еволюційна модель, модель формальної розробки систем, модель на основі раніше створених компонентів, ітераційні моделі, компонентно-орієнтована модель. Життєвий цикл розробки безпечного програмного забезпечення (Security Development Lifecycle, SDL). Особливості SDL. Огляд стандартів процесу створення ПЗ. Організація розробки ПЗ. Стадії та етапи створення ПЗ: формування вимог, розробка концепції, технічне завдання, ескізний проект, технічний проект, робоча документація, введення в дію, супровід. Створення програмної документації: документи управління розробкою та документи що входять до складу ПЗ. Якість ПЗ. Характеристики якості ПЗ. Моделі якості процесу розробки. Статичні та динамічні засоби аналізу програмного коду на наявність вразливостей часу проектування та реалізації.

### **Змістовий модуль 5. Класи збереження даних**

Використання класів для реалізації файлового введення/виведення. Створення, відкриття та закриття файлу. Коректний доступ до файлів. Визначення режимів роботи з файлами. Загальна характеристика структур збереження даних: лінійні та ієрархічні структури даних. Лінійні структури даних: список, стек, черга. Класи колекцій для роботи з структурованими даними.

### **Змістовий модуль 6. Класи базових елементів графічного інтерфейсу користувача**

Загальна характеристика бібліотеки класів Windows Forms.. Загальна характеристика Windows та основи програмування для ОС Windows. Інтерфейс прикладного програмування Win32API. Динамічні бібліотеки Win32API. Структура Windows-програми. Проектування та розробка основних компонент графічного інтерфейсу користувача (GUI). Обробка повідомлень. Засоби обробки подій прикладної програми. Стандартні ресурси прикладної програми: меню, шаблони діалогових блоків, бітові масиви, курсори, піктограми, акселератори, шаблони

панелей інструментів. Класи стандартних елементів управління. Класи графічного інтерфейсу. Графічні об'єкти. Графічні методи. Типи меню. Клас для роботи з меню.

### **Змістовий модуль 7. Розробка додатків в інтегрованих середовищах. Побудова керованого та захищеного коду за допомогою Windows Forms, ADO.NET, WPF, ASP.NET**

Можливості системи безпеки Code Access Security (CAS). Типи проектів. Бібліотека BCL для написання безпечних програм, орієнтованих на загальнономовне середовище виконання Common Language Runtime (CLR). Бібліотека класів System.Security.Cryptography. Засоби збереження даних. Стандарти доступу до баз даних. ODBC, OLE DB постачальники. Класи для роботи з базами даних бібліотеки ADO.NET (ActiveX Data Objects .NET). Способи роботи з файлами. Класи побудови графічних додатків з багатим інтерфейсом бібліотеки WPF (Windows Presentation Foundation). Засоби для створення веб-додатків та веб-сервісів - ASP.NET.

## **6. Контроль навчальних досягнень**

Навчальні досягнення студентів з дисципліни оцінюються за модульно-рейтинговою системою, в основу якої покладено принцип поопераційної звітності, обов'язковості модульного контролю, накопичувальної системи оцінювання рівня знань, умінь та навичок, розширення кількості підсумкових балів до 100.

Оцінка за кожний змістовий модуль включає бали за поточну роботу студента на лабораторних, практичних заняттях, за виконання індивідуальних завдань, за модульну контрольну роботу. Виконання модульних контрольних робіт здійснюється в електронному вигляді. Модульний контроль знань студентів здійснюється після завершення вивчення навчального матеріалу змістового модуля.

У процесі оцінювання навчальних досягнень студентів застосовуються такі методи:

- *Методи усного контролю:* індивідуальне опитування, фронтальне опитування, співбесіда.
- *Комп'ютерного контролю:* тестові програми, програмні проекти.
- *Методи самоконтролю:* уміння самостійно оцінювати свої знання, самоаналіз.

Кількість балів за роботу з теоретичним матеріалом, на лабораторних заняттях, під час виконання самостійної роботи залежить від дотримання таких вимог:

- систематичність відвідування занять;
- своєчасність виконання навчальних і індивідуальних завдань;
- повний обсяг їх виконання;
- якість виконання навчальних і індивідуальних завдань;
- самостійність виконання;
- творчий підхід у виконанні завдань;
- ініціативність у навчальній діяльності;
- виконання тестових завдань.

Контроль успішності студентів з урахуванням поточного і підсумкового оцінювання здійснюється відповідно до навчально-методичної карти дисципліни, де зазначено види контролю і кількість балів за видами. Систему рейтингових балів для різних видів контролю та порядок їх переведення у національну (4-бальну) та європейську (ECTS) шкалу подано нижче у таблицях.



## Розрахунок рейтингових балів за видами поточного (модульного) контролю

### 2 семестр

№ з/п	Вид діяльності студента	Макс. кількість балів за одиницю	Модуль 1		Модуль 2		Модуль 3	
			Кількість. одиниць до розрахунку	Макс. кількість балів за вид	Кількість. одиниць до розрахунку	Макс. кількість балів за вид	Кількість. одиниць до розрахунку	Макс. кількість балів за вид
1	Відвідування лекцій	1	4	4	1	1	2	2
2	Відвідування лабораторних занять	1	6	6	4	4	4	4
3	Виконання завдань для самостійної роботи	5	1	5	1	5	1	5
5	Лабораторне заняття (допуск, виконання, захист)	10	6	60	4	40	4	40
6	Виконання модульної контрольної роботи	25	1	25	1	25	1	25
7	Макс. кількість балів за видами поточного контролю (МВ)	-	-	100	-	75	-	76
Максимальна кількість балів: 251								
Розрахунок коефіцієнта: $251/100=2,51$								

### 3 семестр

№ з/п	Вид діяльності студента	Макс. кількість балів за одиницю	Модуль 4		Модуль 5		Модуль 6		Модуль 7	
			Кількість. одиниць до розрахунку	Макс. кількість балів за вид	Кількість. одиниць до розрахунку	Макс. кількість балів за вид	Кількість. одиниць до розрахунку	Макс. кількість балів за вид	Кількість. одиниць до розрахунку	Макс. кількість балів за вид
1	Відвідування лекцій	1	4	4	2	2	2	2	3	3
2	Відвідування лабораторних занять	1	5	5	4	4	5	5	3	3
3	Виконання завдань для самостійної роботи	5	2	10	2	10	2	10	2	10
4	Лабораторне заняття (допуск, виконання, захист)	10	5	50	4	40	5	50	3	30
5	Виконання модульної контрольної роботи	25	1	25	1	25	1	25	1	25
6	Макс. кількість балів за видами поточного контролю (МВ)	-	-	94	-	81	-	92		71
Максимальна кількість балів: 338										
Розрахунок коефіцієнта: $338/60=5,63$										

### Завдання для самостійної роботи та критерії її оцінювання.

Самостійна робота є видом поза аудиторної індивідуальної діяльності студента, результати якої використовуються у процесі вивчення програмового матеріалу навчальної дисципліни та містить результати дослідницького пошуку, відображає певний рівень його

навчальної компетентності.

### Перелік тем та оцінювання самостійної роботи студента - 2 семестр

№ з/п	Назва теми	Кількість годин	Бали
<b>Змістовий модуль 1. . Основні принципи безпечного програмування</b>		<b>14</b>	<b>5</b>
1.	Виконання програмного проекту за темою «Управління потоком виконання програми»	14	5
<b>Змістовий модуль 2. Структуровані типи даних</b>		<b>14</b>	<b>5</b>
2.	Виконання програмного проекту за темою «Лінійні структури збереження даних та основні алгоритми роботи з даними»	14	5
<b>Змістовий модуль 3. Технологія структурного процедурного програмування</b>		<b>14</b>	<b>5</b>
7.	Виконання програмного проекту за темою «Методи»	14	5
<b>Разом</b>		<b>42</b>	<b>15</b>

### Критерії оцінювання самостійної роботи студента – 2 семестр

№ п/п	Критерії оцінювання роботи	Максимальна кількість балів за кожним критерієм
1	Повний обсяг їх виконання. Якість виконання, відповідність змісту завдання.	2 бали
2	Самостійність виконання. Графічна форма представлення алгоритму розв'язку задачі. Структурований код, дружній та безпечний інтерфейс. Доказовість результатів, визначення шляхів вдосконалення програми.	2 бали
3	Дотримання вимог щодо технічного оформлення.	1 бал
Разом		5 балів

### Перелік тем та оцінювання самостійної роботи студента - 3 семестр

№ з/п	Назва теми	Кількість годин	Бали
<b>Змістовий модуль 4. Основи об'єктно-орієнтованого програмування</b>		<b>14</b>	<b>10</b>
1.	Виконання програмного проекту за темою «Успадкування. Використання інтерфейсів»	14	10
<b>Змістовий модуль 5. Класи збереження даних</b>		<b>14</b>	<b>10</b>
2.	Виконання програмного проекту за темою «Файлове введення/виведення об'єктів»	14	10
<b>Змістовий модуль 6. Класи базових елементів графічного інтерфейсу користувача</b>		<b>14</b>	<b>10</b>
3.	Виконання завдання за темою «Технічне завдання на розробку прикладної програми»	14	10
<b>Змістовий модуль 7. Розробка додатків в інтегрованих середовищах. Побудова керованого та захищеного коду за допомогою Windows Forms, ADO.NET, WPF, ASP.NET</b>		<b>14</b>	<b>10</b>
4.	Виконання завдання за темою «Проектування графічного інтерфейсу прикладної програми»	14	10
<b>Разом</b>		<b>56</b>	<b>40</b>

### Критерії оцінювання самостійної роботи студента – 3 семестр

№ п/п	Критерії оцінювання роботи	Максимальна кількість балів за кожним критерієм
1	Повний обсяг їх виконання. Якість виконання,	2 бали

№ п/п	Критерії оцінювання роботи	Максимальна кількість балів за кожним критерієм
	відповідність змісту завдання.	
2	Самостійність виконання. Розробка та використання класів. Дружній та безпечний інтерфейс. Доказовість результатів. Творчий підхід у виконанні завдань.	2 бали
3	Дотримання вимог щодо технічного оформлення.	1 бал
	Разом	5 балів

### Індивідуальні завдання

Вид індивідуальних завдань	Тематика індивідуальних завдань	Бали
Курсова робота	<p>Проектування та розробка об'єктно-орієнтованої прикладної програми з графічним інтерфейсом користувача (область застосування може пропонуватися студентом) згідно до методичних порад з виконання курсової роботи пропонуються такі напрямки створення прикладних програм:</p> <ol style="list-style-type: none"> <li>1. Проектування та реалізація прикладної програми з графічним інтерфейсом «Шифрування тексту».</li> <li>2. Проектування та реалізація прикладної програми з графічним інтерфейсом для роботи з дисками та каталогами (простий файловий поведир).</li> <li>3. Проектування та реалізація прикладної програми з графічним інтерфейсом «Інтернет послуги».</li> <li>4. Проектування та реалізація прикладної програми з графічним інтерфейсом з генерування пароллю.</li> <li>5. Проектування та реалізація прикладної програми з графічним інтерфейсом «Успішність студентів».</li> <li>6. Проектування та реалізація прикладної програми з графічним інтерфейсом «Операції з многочленами».</li> <li>7. Проектування та реалізація прикладної програми з графічним інтерфейсом «Довідник з перетворення одиниць виміру».</li> <li>8. Проектування та реалізація прикладної програми «Операції з матрицями».</li> <li>9. Проектування та реалізація прикладної програми презентаційної графіки для побудови діаграм.</li> <li>10. Проектування та реалізація ігрової програми (назва гри).</li> <li>11. Проектування та реалізація прикладної програми зміни системних установок.</li> <li>12. Проектування та реалізація довідкової системи з мови програмування C# (C++).</li> <li>13. Проектування та реалізація навчальної програми створення та використання елементів управління.</li> <li>14. Проектування та реалізація прикладної програми з використанням бази даних (назва бази даних).</li> <li>15. Проектування та реалізація веб-додатка (назва).</li> <li>16. Проектування та реалізація прикладної програми з графічним інтерфейсом «Довідник з систем числення»</li> <li>17. Проектування та реалізація прикладної програми для розв'язку системи лінійних рівнянь методом Гауса.</li> </ol>	100

	<ol style="list-style-type: none"> <li>18. Проектування та реалізація прикладної програми “Відділ кадрів”.</li> <li>19. Проектування та реалізація прикладної програми “Розклад занять”.</li> <li>20. Проектування та реалізація прикладної програми з використанням бази даних “Телефонний довідник”.</li> <li>21. Проектування та реалізація прикладної програми з графічним інтерфейсом “Банкомат”.</li> <li>22. Проектування та реалізація прикладної програми з графічним інтерфейсом “Замовлення білетів”.</li> <li>23. Проектування та реалізація прикладної програми з графічним інтерфейсом “Веб-браузер”.</li> <li>24. Проектування та реалізація прикладної програми “Деканат”.</li> <li>25. Проектування та реалізація довідкової програми з створення різних типів меню.</li> <li>26. Проектування та реалізація прикладної програми з графічним інтерфейсом «Заробітна плата».</li> <li>27. Проектування та реалізація прикладної програми з графічним інтерфейсом «Довідник з обрахунку значень тригонометричних функцій»</li> <li>28. Проектування та реалізація прикладної програми “Простий графічний редактор”.</li> <li>29. Проектування та реалізація прикладної програми з графічним інтерфейсом “Туристична агенція”.</li> <li>30. Проектування та реалізація прикладної програми з графічним інтерфейсом “Довідник країн світу”.</li> <li>31. Проектування та реалізація прикладної програми з графічним інтерфейсом “Абітурієнт”.</li> <li>32. Проектування та реалізація прикладної програми з графічним інтерфейсом “Бібліотека”.</li> <li>33. Проектування та реалізація прикладної програми з графічним інтерфейсом “Курси іноземних мов”.</li> <li>34. Проектування та реалізація прикладної програми з графічним інтерфейсом “Довідник програмного забезпечення”.</li> <li>35. Проектування та реалізація прикладної програми з графічним інтерфейсом “Обмін валют”.</li> <li>36. Проектування та реалізація прикладної програми з графічним інтерфейсом “Банківський рахунок”.</li> <li>37. Проектування та реалізація прикладної програми з графічним інтерфейсом «Знаходження найбільшого спільного дільника многочленів».</li> <li>38. Проектування та реалізація прикладної програми з графічним інтерфейсом “Соціологічне опитування”.</li> <li>39. Проектування та реалізація програми сервера ActiveX.</li> <li>40. Проектування та реалізація довідкової прикладної програми з графічним інтерфейсом “Мови програмування”.</li> <li>41. Проектування та реалізація прикладної програми з графічним інтерфейсом “Довідник з видів вразливостей програмного коду”.</li> <li>42. Проектування та реалізація прикладної програми з</li> </ol>	
--	---	--

	графічним інтерфейсом «Вікторина з програмування».	
	43. Проектування та реалізація мобільного додатку (назва).	

**Розподіл балів з оцінювання КР ( 100 балів):**

- Письмова частина КР -35 балів (10 балів -оформлення документа, 25 балів - вміст)
- Програмний проект КР - 45 балів.
- Захист КР - 20 балів. ( 10 б. - презентація роботи, 10 б. - оцінювання/самооцінювання всіх робіт студентів групи)

**Розрахунок рейтингових балів за виконання курсової роботи**

Форми контролю та критерії оцінювання	Кількість балів
Курсова робота виконана вчасно згідно із завданням.	10
Робота структурована, матеріал роботи викладено логічно.	15
Якість оформлення протоколу виконання курсової роботи. (відсутність стилістичних та граматичних помилок).	10
Програмний проект має дружній та безпечний інтерфейс.	15
Проведено статичний аналіз коду та оцінка якості коду	10
Данні програми зберігаються на зовнішніх носіях.	10
Наявність засобів розмежування доступу користувачів програми.	10
Презентація роботи. Доповідь логічна, стисла, структурована, проголошена вільно.	10
Відповіді на запитання (повні та обґрунтовані).	10
<b>Разом</b>	<b>100</b>

**Форми проведення модульного контролю та критерії оцінювання.**

Модульний контроль здійснюється відповідно до навчально-методичної карти дисципліни та перевіряє рівень досягнення результатів навчання студентів. Форма проведення комбінована. Модульна контрольна робота оцінюється у 25 балів. Для 1,2, 4-7 змістових модулів - комп'ютерний тест, що складається з 15-25 запитань закритої та відкритої форм (25 балів), для 3 змістового модуля - комп'ютерний тест, що складається з 15 запитань закритої форми (15 балів) та практико-орієнтованого завдання за темами, визначеними в тематичному плані (10 балів).

**Розрахунок рейтингових балів за модульний контроль(1,2, 4-7 змістові модулі)**

Форма проведення	Кількість завдань	Кількість балів за одиницю	Максимальна підсумкова кількість балів
Комп'ютерний тест	15-25	2-1	25 балів
Разом			25 балів

**Розрахунок рейтингових балів за модульний контроль(3 змістовий модуль)**

Форма проведення	Кількість завдань	Кількість балів за одиницю	Максимальна підсумкова кількість балів
Комп'ютерний тест	15	1	15 балів
Тематичне практичне завдання	1	10	10 балів
Разом			25 балів

### Критерії оцінювання виконання тематичного практичного завдання

№ п/п	Критерії оцінювання роботи	Максимальна кількість балів за кожним критерієм
1	Програмний проект виконано вчасно і в повному обсязі	4 бали
2	Відповідність змісту завдання, структурований код, дружній та безпечний інтерфейс.	3 бали
3	Повні та обґрунтовані відповіді на запитання. Самостійність виконання	3 бали
Разом		10 балів

### Форми проведення семестрового контролю та критерії оцінювання.

**2 семестр.** Форма семестрового контролю – залік. Оцінюється як сума отриманих балів протягом роботи у семестрі відповідно до навчально-методичної карти дисципліни та таблиці розрахунку рейтингових балів за видами поточного (модульного) контролю з коефіцієнтом 2,51.

**3 семестр.** Семестрове (підсумкове) оцінювання здійснюється у формі екзамену, умовою допуску до якого є отриманням студентом 35 балів (з врахуванням коефіцієнту) за результатами поточного контролю.

Форма проведення екзамену – комбінована. Екзамен оцінюється у 40 балів за розподілом: 20 балів – комплексний тест з дисципліни; 20 балів – виконання практико-орієнтованого завдання (програмний проект). Засоби підсумкового контролю – тест (комп'ютерний) для підсумкового контролю, комплект практичних завдань.

Виконання практичного завдання передбачає перевірку рівня оволодіння студентом теоретичними знаннями та практичними вміннями з проектування та реалізації прикладних програм.

Оцінювання практичного завдання відбувається в межах від 0 до 20 балів, згідно критеріїв оцінювання, й здійснюється з урахуванням: рівнів сформованості аналітико-синтетичних, творчих та методичних умінь необхідних для написання безпечного коду.

Бали за виконання тесту та бали за виконання практичного завдання додаються. Оцінювання результатів засвоєння теоретичних знань та оцінювання сформованості практичних навичок володіння технологіями безпечного програмування, продемонстровані студентами на екзамені, представлене у таблицях.

### Розрахунок рейтингових балів за семестровий контроль (екзамен)

Форма проведення	Кількість завдань	Кількість балів за одиницю	Максимальна підсумкова кількість балів
Комплексний тест	20	1	20 балів
Практичне завдання	1	20	20 балів
Разом			40 балів

### Критерії оцінювання виконання практичного завдання

№ п/п	Критерії оцінювання роботи	Максимальна кількість балів за кожним критерієм
1	Повний обсяг виконання	5 бали
2	Якість виконання, відповідність змісту завданню.	5 бали
3	Творчий підхід у виконанні завдання. Структурований код, дружній та безпечний інтерфейс, наявність структур збереження даних, засобів розмежування доступу.	5 бали
4	Повні та обґрунтовані відповіді на запитання. Самостійність виконання	5 бали
Разом		20 балів

## Орієнтовний перелік питань для семестрового контролю.

1. Основні поняття безпечного програмування.
2. Життєвий цикл розробки безпечного програмного забезпечення (ПЗ).
3. Створення керованого та захищеного об'єктно-орієнтованого коду з використанням технології .NET.
4. Обробка виняткових ситуацій.
5. Базові типи даних.
6. Данні програми. Безпека даних.
7. Основні операції. Пріоритет операцій.
8. Типи структур управління: послідовна структура, структура вибору, структура повторення.
9. Одновимірні масиви та символічні рядки.
10. Багатовимірні масиви.
11. Загальна характеристика структур збереження даних: лінійні та ієрархічні структури даних.
12. Лінійні структури даних: список, стек, черга.
13. Методи.
14. Формальні та фактичні параметри методів.
15. Реалізація файлового введення/виведення.
16. Коректний доступ до файлів.
17. Базові концепції класів та об'єктів.
18. Визначення класу та методів класу.
19. Модифікатори доступу до елементів класу.
20. Конструктори та ініціалізація.
21. Перевантаження операторів та методів класу.
22. Статичні члени класу.
23. Просте успадкування класів. Правила створення похідних класів.
24. Віртуальні методи. Заміщені методи. Приховані (нові) методи.
25. Абстрактні класи. Закриті класи.
26. Базові принципи об'єктно-орієнтованого програмування: інкапсуляція, успадкування, поліморфізм.
27. Бібліотека Base Class Library (BCL) для написання безпечних програм, орієнтованих на загальнономовне середовище виконання Common Language Runtime (CLR).
28. Характеристика класів бібліотеки BCL (Base Class Library).
29. Проектування та розробка основних компонент графічного інтерфейсу користувача (GUI).
30. Засоби обробки подій прикладної програми.
31. Стандартні елементи прикладної програми: меню, шаблони діалогових блоків, бітові масиви, курсори, піктограми, шаблони панелей інструментів.
32. Класи форми та стандартних елементів управління.
33. Класи графічного інтерфейсу.
34. Типи меню.
35. Класи для роботи з меню, панеллю швидкого вибору елементів, рядком стану.
36. Стандарти доступу до баз даних.
37. Моделі процесу створення ПЗ. Оцінка якості коду.
38. Тестування, налагодження та статичний аналіз коду.

### Шкала відповідності оцінок

Рейтингова оцінка	Сума балів за всі види навчальної діяльності	Значення оцінки
<b>A</b>	90-100	Відмінно — відмінний рівень знань (умінь) в межах обов'язкового матеріалу з, можливими, незначними недоліками
<b>B</b>	82-89	Дуже добре - достатньо високий рівень знань (умінь) в межах обов'язкового матеріалу без суттєвих (грубих) помилок
<b>C</b>	75-81	Добре - в цілому добрий рівень знань (умінь) з незначною кількістю помилок
<b>D</b>	69-74	Задовільно - посередній рівень знань (умінь) із значною кількістю недоліків, достатній для подальшого навчання або професійної діяльності
<b>E</b>	60-68	Достатньо - мінімально можливий допустимий рівень знань (умінь)
<b>FX</b>	35-59	Незадовільно з можливістю повторного складання - незадовільний рівень знань, з можливістю повторного перескладання за умови належного самостійного доопрацювання
<b>F</b>	1-34	Незадовільно з обов'язковим повторним вивченням курсу - досить низький рівень знань (умінь), що вимагає повторного вивчення дисципліни



## 1. Навчально-методична картка дисципліни

Разом за 2 семестр: 90 год.: лекції – 14 год., лабораторні роботи – 28 год., МК – 6 год., самостійна робота – 42 год.

Модулі (назви, бали)	Змістовий модуль 1. Основні принципи безпечного програмування (100 балів)			Змістовий модуль 2. Структуровані типи даних (75 балів)			Змістовий модуль 3. Технологія структурного процедурного програмування (76 балів)			
Лекції (теми, бали)	Основні поняття програмування. (1 бал)	Поняття даних. Стандартні типи даних (1 бал)	Оператори управління (2 бали)	Лінійні структури збереження даних та основні алгоритми роботи з даними. (1 бал)			Розбиття програми на підпрограми. Методи (2 бали)			
Лабораторні роботи (теми, бали)	Розробка консольних програм лінійної структури (11 балів)	Проектування, реалізація та налагодження обчислювальних прикладних програм (11 балів)	Використання операторів розгалуження (11 балів)	Проектування, реалізація та налагодження консольних програм з розгалуженням (11 балів)	Використання операторів повторення (11 балів)	Розробка та налагодження консольних програм з повторенням (11 балів)	Реалізація одновимірних масивів (11 балів)	Проектування, реалізація та налагодження ПП з використанням одновимірних масивів (11 балів)	Реалізація двовимірних масивів (11 балів)	Проектування, реалізація та налагодження ПП для виконання операцій з матрицями (11 балів)
Самостійна робота	Самостійна робота 1 (5 балів)			Самостійна робота 2 (5 балів)			Самостійна робота 3 (5 балів)			
Модульний контроль	Модульна контрольна робота 1 (25 балів)			Модульна контрольна робота 2 (25 балів)			Модульна контрольна робота 3 (25 балів)			
Підсумковий контроль (вид, бали)	Залік									

Разом за 3 семестр: 180 год.: лекції – 22 год., лабораторні роботи – 34 год., МК – 8 год., самостійна робота – 56 год.

Модулі (назви, бали)	Змістовий модуль 4. Основи об'єктно-орієнтованого програмування (94 бали)			Змістовий модуль 5. Класи збереження даних (81 бал)			Змістовий модуль 6. Класи базових елементів графічного інтерфейсу користувача (92 бали)			Змістовий модуль 7. Розробка додатків в інтегрованих середовищах. Побудова керованого та захищеного коду за допомогою Windows Forms, ADO.NET, WPF, ASP.NET (71 бал)		
Лекції (теми, бали)	Класи та об'єкти. Принципи об'єктно-орієнтованого програмування.(1 бал)			Класи для роботи з файлами. (1 бал)			Класи колекцій. (1 бал)			Проектування та розробка основних компонент графічного інтерфейсу користувача. Бібліотека класів .Net Framework. (1 бал)		
	Базові концепції класів та об'єктів (1 бал)									. Класи стандартних елементів управління (1 бал)		
	Відношення між класами. Успадкування (1 бал)									Класи для роботи з графікою (1 бал)		
	Технологія розробки безпечного об'єктно-орієнтованого ПЗ (1 бал)									Засоби для створення веб-додатків - ASP.NET(1 бал)		
Лабораторні роботи (теми, бали)	Опис класів. Виконання операцій з об'єктами класів. (11 балів)			Організація файлового введення/виведення даних (11 балів)			Розбиття коду на підпрограми. Розробка методів. (11 балів)			Класи для роботи з графікою (1 бал)		
	Проектування, реалізація та налагодження консольних ПП з використанням класів та об'єктів (22 бали)			Проектування, реалізація та налагодження ПП з використанням файлового введення/виведення даних (11 балів)			Проектування, реалізація та налагодження об'єктно-орієнтованих ПП на базі форми (22 бали)			Засоби для створення веб-додатків - ASP.NET(1 бал)		
	Створення похідних класів. Робота з об'єктами похідних класів (11 балів)			Реалізація списків, черг, стеків (11 балів)			Обмін даними між діалоговими вікнами, збереження даних (11 балів)			Засоби збереження даних. DataGridView, DB (1 бал)		
	Проектування, реалізація та налагодження консольних ПП з розгалуженням (11 балів)			Проектування, реалізація та налагодження ПП з використанням класів колекцій (11 балів)			Створення меню різних типів, рядка стану та панелі швидкого вибору пунктів меню (11 балів)			Засоби збереження даних. DataGridView, DB (1 бал)		
Самостійна робота	Самостійна робота 1 (10 балів)			Самостійна робота 2 (10 балів)			Самостійна робота 3 (10 балів)			Самостійна робота 4 (10 балів)		
Модульний контроль	Модульна контрольна робота 1 (25 балів)			Модульна контрольна робота 2 (25 балів)			Модульна контрольна робота 3 (25 балів)			Модульна контрольна робота 4 (25 балів)		
Підсумковий контроль (вид, бали)	Іспит (40 балів)											

## 2. Рекомендовані джерела

### Основна (базова):

1. Andrew Stellman, Jennifer Greene Head First C#. — O'Reilly Media, Inc., 4th Edition, 2020. - 560 p.
2. Mark J. Price. C# 10 and .NET 6 – Modern Cross-Platform Development: Build apps, websites, and services with ASP.NET Core 6, Blazor, and EF Core 6 using Visual Studio 2022 and Visual Studio Code, 6th Edition . — Packt Publishing, 2021. - 824 p.
3. Албахари Дж. С# 9.0. Довідник. — К.: Діалектика, 8-е видання, том 2, 2021. - 1056 с.
4. John Sharp. Microsoft Visual C# Step by Step (Developer Reference). — Packt Publishing, 9th Edition, 2020. - 832 p.
5. McConnell S. Code Complete: A Practical Handbook of Software Construction, Second Edition. — Microsoft Press, 2016. - 916 с.
6. Коноваленко І.В. Програмування мовою С# 6.0. Навчальний посібник для технічних спеціальностей вищих навчальних закладів. - Тернопіль.: ТНТУ, 2016. – 229 с.
7. Роберт С. Мартін Чистий код: створення, аналіз і рефакторинг. — К.: Український книжковий дім "Фабула", 2019. - 368 с.

### Додаткова:

1. Thomas H. Cormen Introduction to Algorithms — MIT Press, 9th Edition, 2020. - 1293 p.
2. Martin Fowler, Kent Beck, John Brant, William Opdyke, Don Roberts. Refactoring: Improving the Design of Existing Code . — Addison-Wesley Professional, 2nd Edition, 2018. – 432 p.
3. Jason Alls. Clean Code in C#: Refactor your legacy C# code base and improve application performance by applying best practices, — Packt Publishing, 2020. - 500 p.

### Додаткові ресурси:

1. Visual Studio 2019, 2022 (англ.) [Електронний ресурс]. – Режим доступу: <https://docs.microsoft.com/en-us/visualstudio/windows/?view=vs-2022>
2. Документація з С# (англ.) [Електронний ресурс]. – Режим доступу: <https://docs.microsoft.com/en-us/dotnet/csharp/>
3. С# .Net> Посібник (укр.) [Електронний ресурс]. – Режим доступу: <https://programm.top/uk/c-sharp/tutorial/>
4. ВікіПідручник C Sharp 2017 (укр.) [Електронний ресурс]. – Режим доступу: [https://uk.wikibooks.org/wiki/C\\_Sharp](https://uk.wikibooks.org/wiki/C_Sharp)
5. Портал знань. Основи С# (укр.) [Електронний ресурс]. – Режим доступу: <http://www.znannya.org/?view=csharp>
6. Аналіз коду (англ.) [Електронний ресурс]. – Режим доступу: <https://docs.microsoft.com/en-us/visualstudio/code-quality/?view=vs-2019>
7. Оцінка якості коду (метрики коду) (англ.) [Електронний ресурс]. – Режим доступу: <https://docs.microsoft.com/en-us/visualstudio/code-quality/code-metrics-values?view=vs-2019>
8. Создание приложения Windows Forms (англ.) [Електронний ресурс]. – Режим доступу: <https://docs.microsoft.com/en-us/visualstudio/ide/create-csharp-winform-visual-studio?view=vs-2022>
9. Windows Forms (укр.) [Електронний ресурс]. – Режим доступу: <https://metanit.com/sharp/windowsforms/>
10. ASP.NET Core 2.0 (укр.) [Електронний ресурс]. – Режим доступу: <https://metanit.com/sharp/aspnet5/>