

Міністерство освіти і науки України  
Національний аерокосмічний університет  
«Харківський авіаційний інститут»

Кафедра комп'ютерних систем, мереж і кібербезпеки (№ 503 )

**ЗАТВЕРДЖУЮ**

Гарант освітньої програми

  
Анатолій Шостак  
(підпис) (ініціали та прізвище)

« 29 » серпня 2025 р.

**СИЛАБУС *ОБОВ'ЯЗКОВОЇ*  
НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

Системне програмування

(назва навчальної дисципліни)

Галузь знань: 12 "Інформаційні технології"  
(шифр і найменування галузі знань)

Спеціальність: 123 "Комп'ютерна інженерія"  
(код та найменування спеціальності)

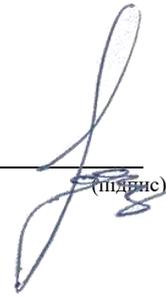
Освітня програма: Системне програмування

Рівень вищої освіти: перший (бакалаврський)

**Силабус введено в дію з 01.09.2025 року**

**Харків 2025 рік**

Розробник: Годунов О.С., ст. викладач  
(прізвище та ініціали, посада, науковий ступінь та вчене звання)



(підпис)

Силабус навчальної дисципліни розглянуто на засіданні кафедри \_\_\_\_\_  
«Комп'ютерних систем, мереж і кібербезпеки»  
(назва кафедри)

Протокол № 1 від « 29 » \_\_\_\_\_ серпня \_\_\_\_\_ 2025 року

Завідувач кафедри д.т.н., професор \_\_\_\_\_ В. С. Харченко  
(науковий ступінь та вчене звання) (підпис) (ініціали та прізвище)

Представник здобувачів освіти

\_\_\_\_\_



(підпис)

П. Огарко

(ініціали та прізвище)

## 1. Загальна інформація про викладача



*ПІБ: Годунов Олександр Сергійович*

---

*Посада: старший викладач кафедри комп'ютерних систем, мереж і кібербезпеки*

---

*Науковий ступінь:*

---

*Вчене звання:*

---

*E-mail: a.godunov@csn.khai.edu*

---

*Перелік дисциплін, які викладає: Веб-технології, Системне програмування, Веб-програмування.*

---

*Напрями наукових досліджень: веб-технології, веб-сервіси, хмарні технології, операційні системи*

---

## 2. Опис навчальної дисципліни

Форма здобуття освіти	<i>Денна, заочна</i>
Семестр	<i>7</i>
Мова викладання	<i>Українська</i>
Тип дисципліни	<i>Обов'язкова</i>
Обсяг дисципліни: кредити ЄКТС / кількість годин	<i>Денна: 4 кредиту / 120 годин (48 аудиторних, з яких: лекції – 32, лабораторні – 16, СРЗ – 72) Заочна: 4 кредиту / 120 годин (8 аудиторних, з яких: лекції – 4, лабораторні – 4, СРЗ – 112)</i>
Види навчальної діяльності	<i>Лекції, лабораторні заняття, самостійна робота здобувача</i>
Види контролю	<i>Поточний контроль, модульний контроль, семестровий контроль – іспит</i>
Пререквізити	<i>"Основи Функціонування комп'ютерів", "Технології програмування", "Операційні системи", "Технології програмування (КП) ", "Комп'ютерні мережі"</i>

### **3. Мета та завдання навчальної дисципліни, переліки компетентностей та очікуваних результатів навчання**

**Мета:** Метою курсу є ознайомлення студентів з системними платформами розробки додатків під операційну систему Windows. Розгляд архітектури створених додатків на базі платформ Win32, .NET Framework і .NET Core, UWP, WinUI 3, а так само вивчення системних об'єктів.

**Завдання:** вивчення системних об'єктів: Файли, Файлові проєкції, Процеси, Потоки, Об'єкти синхронізації (mutex, semaphore, event, critical sections) і Сокети;

Вивчення принципів побудови додатків на Win32 / C++;

Вивчення системних об'єктів .NET framework/.NET Core для роботи з процесами, потоками, об'єктами синхронізації у рамках формату WPF. Tasks та await/async;

Вивчення архітектури UWP додатків, та основ створення інтерфейсів для UWP додатків;

Вивчення архітектури WinUI 3 додатків, та основ створення інтерфейсів у даному форматі.

#### **Компетентності, які набуваються:**

**Інтегральна компетентність:** Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми під час професійної діяльності в комп'ютерній галузі або навчання, що передбачає застосування теорій та методів комп'ютерної інженерії і характеризується комплексністю та невизначеністю умов.

#### **Загальні компетентності (ЗК):**

- ЗК2. Здатність вчитися і оволодівати сучасними знаннями.
- ЗК3. Здатність застосовувати знання у практичних ситуаціях.

#### **Фахові компетентності (ФК):**

– ФК2. Здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення.

– ФК3. Здатність створювати системне та прикладне програмне забезпечення комп'ютерних систем та мереж.

– ФК13. Здатність вирішувати проблеми у галузі комп'ютерних та інформаційних технологій, визначати обмеження цих технологій.

– ФК16. Вміти оцінювати отримані результати та аргументовано захищати прийняті рішення.

– ФК17. Спілкуватись усно та письмово з професійних питань українською мовою та однією з іноземних мов (англійською, німецькою, італійською, французькою, іспанською).

– ФК18. Використовувати інформаційні технології та для ефективного спілкування на професійному та соціальному рівнях.

***Програмні результати навчання (ПРН):***

- ПРН1. Знати і розуміти наукові положення, що лежать в основі функціонування комп'ютерних засобів, систем та мереж.
- ПРН3. Знати новітні технології в галузі комп'ютерної інженерії.
- ПРН4. Знати та розуміти вплив технічних рішень в суспільному, економічному, соціальному і екологічному контексті.
- ПРН6. Вміти застосовувати знання для ідентифікації, формулювання і розв'язування технічних задач спеціальності, використовуючи методи, що є найбільш придатними для досягнення поставлених цілей.
- ПРН7. Вміти розв'язувати задачі аналізу та синтезу засобів, характерних для спеціальності.
- ПРН8. Вміти системно мислити та застосовувати творчі здібності до формування нових ідей.
- ПРН10. Вміти розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем, розраховувати, експлуатувати, типове для спеціальності обладнання.
- ПРН13. Вміти ідентифікувати, класифікувати та описувати роботу комп'ютерних систем та їх компонентів.
- ПРН16. Вміти оцінювати отримані результати та аргументовано захищати прийняті рішення.
- ПРН19. Здатність адаптуватись до нових ситуацій, обґрунтовувати, приймати та реалізовувати у межах компетенції рішення.
- ПРН20. Усвідомлювати необхідність навчання впродовж усього життя з метою поглиблення набутих та здобуття нових фахових знань, удосконалення креативного мислення.
- ПРН24. Вміти аналізувати, оцінювати та забезпечувати надійність системного програмного забезпечення впродовж розроблення, тестування та використання.

## 4. Зміст навчальної дисципліни

### МОДУЛЬ 1

#### Змістовий модуль 1 – Вступ до системного програмування. Платформа Win32.

##### **Тема 1. Огляд мов, технологій і принципів побудови додатків під ОС Windows.**

*Анотація:* Розгляд хронології появи технологій створення додатків на основі: процедурно-орієнтованого підходу, об'єктно-орієнтованого, компонентного і розподіленого. Огляд існуючих бібліотек і мов, які дозволяють створювати додатки з ОС Windows.

*Тема лекції 1:* Типові підходи до побудови коду в еволюції розвитку системних і прикладних додатків. Доступні платформи розробки системних додатків під Windows. Огляд платформи Win32, формат проекту в Visual Studio, попередня компіляція заголовних файлів, заголовок Windows.h.

*Самостійна робота здобувача:* Аналіз структури та перевірка доступу до ресурсів курсу в системі онлайн навчання. Встановлення Інсталяція необхідних розширень і типів проекту для Visual Studio.

##### **Тема 2. Основи програмування на Win32.**

*Анотація:* Програмування виконується за допомогою мови C++ і середовища розробки Visual Studio. Створення проекту на Visual Studio. Базові типи даних Win32. Налаштування проекту. Обробка помилок від системних викликів на Win32. Базові типи символів і рядків у win32.

*Тема лекції 2:* Типи даних Win32. Підтримувані типи символів Win32 - Їх призначення та відмінності. Принцип представлення символів в кодуваннях ANSI, Unicode, UTF-8. Функції для перекодіровок рядків з одного кодування в інше. Залежність прототипів системних API від налаштувань проекту. Отримання коду останньої помилки та отримання опису помилок із системи.

*Тема лабораторного заняття 1:* Написання першого проекту на Win32. Виведення розміру та діапазонів стандартних типів даних. Розбір завдання на лабораторну роботу №1.

*Самостійна робота здобувача:* Вивчення всіх базових типів даних Win32 та їх трансляція в стандартні типи C++. Ознайомлення з доступними налаштуваннями проекту, передача параметрів під час налагоджувального запуску програми. Стандартні та користувацькі папки для заголовних файлів і бібліотек. Розбір відмінностей функцій main(), wmain(), \_tmain(), WinMain().

##### **Тема 3. Робота з файлами в Win32.**

*Анотація:* Вивчення системних API і підходів в роботі з файлами на Win32. Розглянути такі механізми доступу до файлів: Робота з файлом через файловий вказівник, Робота з файлом за прямими зміщеннями, Асинхронна робота з файлами, Робота з файлами через файлові проекції.

*Тема лекції 3:* Методи роботи з файлами в Win32. Створення та відкриття файлів. Функції роботи з файлами та файловою системою через файловий покажчик. Робота з файлами за поточним покажчиком, прямим зміщенням. Механізм файлових проекцій та функції для роботи з ними. Функції навігації по каталогах. Отримання інформації о файлах.

*Тема лабораторного заняття 2:* Робота з файлами на Win32. Створення двох програм:

а) Створення записника - робота з файлом, що складається з однотипних записів.

б) Створення файлового менеджера - робота з функціями навігації по каталогах і отримання інформації про файли.

*Самостійна робота здобувача:* Вивчення механізму файлових проекцій. Робота з файлами за прямими зміщеннями. Асинхронна робота з файлами.

#### ***Тема 4. Робота з пам'яттю в Win32.***

*Анотація:* Виділення пам'яті в програмі різними способами. Розглянути, що таке купи пам'яті. Вивчення механізмів роботи з пам'яттю через прямі адреси у віртуальній пам'яті.

*Тема лекції 4:* Віртуальна пам'ять у захищеному режимі Windows. Функції роботи з глобальною пам'яттю. Кучі пам'яті та функції роботи з пам'яттю через кучі. Функції роботи з віртуальною пам'яттю за прямими адресами.

*Самостійна робота здобувача:* Оптимізація використання пам'яті через функції роботи за прямими віртуальними адресами. Оптимізація швидкості очищення пам'яті при використанні куп пам'яті.

#### ***Тема 5. Процеси і потоки в Win32.***

*Анотації:* Вивчення об'єктів ядра операційної системи: Процеси і Потоки. Створення дочірніх процесів. Ресурси пов'язані з процесами. Створення багато поточних додатків на Win32. Визначення стека процедур і функцій. Управління потоками і пріоритети потоків. Локальна пам'ять потоку TLS і використання її в складних багато поточних додатках.

*Тема лекції 5:* Процеси потоки та їх ресурси. Вивчення функцій роботи з процесами (отримання списку, інформації про модулі процесу). Створення дочірніх процесів і перехоплення потоків вводу-виводу за допомогою Pipe. Запуск потоків з параметрами і без. Локальна пам'ять потоку - TLS.

*Тема лабораторного заняття 3:* Робота з процесами та потоками на Win32. Необхідно створити два додатки:

а) Програма-архівувальник, яка використовує утиліту 7z для пакування та розпакування файлів в архів і з архіву.

б) Створення багатопотокового додатка із застосуванням TLS пам'яті для зберігання інформації всередині потоку.

*Самостійна робота здобувача:* Більш детальне вивчення системних викликів для отримання списку процесів. Управління пріоритетами потоків, призупинення та перезапуск потоку.

## **Тема 6. Об'єкти синхронізації ядра ОС Windows.**

*Анотація:* Розглянути, що таке об'єкти синхронізації в ОС. Типові завдання синхронізації і застосування об'єктів синхронізації в таких завданнях. Вивчити роботу об'єктів: Mutex, Event, Semaphore, WaitableTimer, CriticalSection.

*Тема лекції 6:* Типові завдання синхронізації: взаємовиключення, паралельна робота, виробник-споживач. Вивчення системних об'єктів синхронізації: Mutex, Event, Semaphore. Конструкція CriticalSection. Таймери очікування. Вивчення win32 api функцій WaitForSingleObject і WaitForMultipleObjects.

*Тема лабораторного заняття 4:* Робота з об'єктами синхронізації на Win32. Необхідно створити два додатки:

а) Додаток, що контролює, скільки екземплярів додатка вже запущено (Mutex)

б) Додаток, який вимірює час роботи 3 потоків з контролем критичної секції та без контролю (CriticalSection)

в) Додаток логер, який дозволяє записувати в файл одночасно обмеженої кількості потоків (Semaphore).

*Самостійна робота здобувача:* Більш детальне вивчення роботи об'єктів Event та WaitableTimer.

## **Модульний контроль 1**

## **МОДУЛЬ 2**

### **Змістовий модуль 2 – Системні об'єкти у .NET Framework/.NET Core**

## **Тема 7. Робота з файлами в .NET Framework**

*Анотація:* Робота із файлами. Базові класи платформи для одержання списку файлів. Відкриття файлів, стандартні класи потоків роботи з файлами. Файлові проєкції та їх реалізація в .NET Framework.

*Тема лекції 7:* Повторення матеріалу з роботи з файлами, вивченого на попередніх курсах. Вивчення механізму файлових проєкцій в .NET Framework.

*Самостійна робота здобувача:* Створення додатків із застосуванням файлових проєкцій.

### ***Тема 8. Процеси і потоки в .NET Framework***

*Анотація:* Знайомство та вивчення класів роботи з процесами, потоками та тасками – Process, Thread, Task.

*Тема лекції 8:* Вивчення класу Process. Отримання списку процесів в системі. Запуск дочірнього процесу і перехоплення потоків вводу/виводу. Клас Thread і делегати для процесу з параметрами і без. Клас Task і запуск паралельних завдань. Директиви await/async запуску паралельних завдань і отримання результату їх виконання.

*Тема лабораторного заняття 5 (частина 1):* Робота з процесами та потоками в .NET. Необхідно створити додатки (повторити теж саме, що і на win32):

а) Програма-архівувальник, яка використовує утиліту 7z для пакування та розпакування файлів в архів і з архіву.

*Самостійна робота здобувача:* Вивчення автоматичного створення завдань на основі асинхронного виклику функцій, що повертають тип Task/Task<>.

### ***Тема 9. Об'єкти синхронізації ядра ОС в .NET Framework***

*Анотація:* Знайомства з класами, що реалізують типові об'єкти синхронізації з ОС – Mutex, Event, Semaphore, Timer.

*Тема лекції 9:* Вивчення об'єктів Mutex, Semaphore, AutoResetEvent, ManualResetEvent, Monitor. Використання оператора lock() для реалізації критичних секцій коду.

*Тема лабораторного заняття 5 (частина 2):* Робота с потоками та об'єктами синхронізації.

а) Додаток, що контролює, скільки екземплярів додатка вже запущено (Mutex)

б) Додаток, який вимірює час роботи 3 потоків з контролем критичної секції та без контролю (Monitor або lock() )

### ***Тема 10. Сокети і модель сокетов Берклі. Сокети в .NET***

*Анотація:* Розглянути що таке Сокет. Ознайомитися з моделлю сокетов Берклі. Вивчити основи передачі даних по мережі за допомогою сокетів. Розглянути базові алгоритми роботи Клієнта і Сервера в моделі WinSock2.

Використання багатопоточності при створенні серверних додатків.  
Бібліотека winsock2.

*Тема лекції 10:* Модель сокетов Берклі. Типові алгоритми побудови сервера і клієнта. Клас Socket і допоміжні типи даних. Створення багатопотокового сервера.

*Тема лабораторного заняття 6:* Створення клієнта і сервера із застосуванням сокетів.

*Самостійна робота здобувача:* Вивчення класів, що реалізують клієнтів до стандартних протоколів прикладного рівня (HTTP, SMTP, та ін.)

### **Змістовий модуль 3 – UWP (Universal Windows Platform)**

#### ***Тема 11. Створення додатків UWP.***

*Анотація:* Архітектура UWP додатків. Знайомство з життєвим циклом додатків UWP. Клас Page і Frame. Нові кореневі навігаційні елементи. Застосовність класів стандартних просторів імен в UWP-додатках.

*Тема лекції 11:* Особливості платформи UWP. Життєвий цикл додатків UWP. Основи створення інтерфейсу в додатках UWP. Класи Page і Frame та історія перемикання інтерфейсів. Ознайомлення з кореневим навігаційним елементом NavigationView.

*Тема лекції 12:* Навігаційні елементи SplitView, Hub. Обробка подій життєвого циклу UWP додатків. Створення багатомовних інтерфейсів, що адаптуються до поточної локалі операційної системи. Клас CustomDialog.

*Тема лабораторного заняття 7:* Створення UWP додатків.

*Самостійна робота здобувача:* Вивчення особливостей створення внутрішньої структури навігаційних елементів NavigationView, SplitView, HubControl. Data binding в UWP додатках.

#### ***Тема 12. Робота з файловою системою на UWP.***

*Анотація:* Розглянути механізми для роботи з файловою системою в UWP. Вивчити класи простору імен Windows.Storage. Асинхронний доступ до методів таких об'єктів. Ознайомитися з допоміжними класами діалогів простору Windows.Storage.Pickers.

*Тема лекції 13:* Вивчення принципів доступу UWP-додатків до файлової системи. Локальне сховище додатків. Вивчення класів StorageFile, StorageFolder, FileIO, KnownFolders, ApplicationData, FolderPicker, FileOpenPicker, FileSavePicker.

*Самостійна робота здобувача:* Вивчення

## **Змістовий модуль 4 – WinUI3 додатки**

### ***Тема 13. Створення додатків на платформи WinUI3.***

*Анотація:* Архітектура додатків WinUI3. Принцип створення інтерфейсів і ключові інтерфейсні класи. Життєвий цикл додатків WinUI3. Особливості додатків WPF і UWP, які перенесені в платформу WinUI3.

*Тема лекції 14:* Особливості платформи WinUI3. Життєвий цикл додатків WinUI3. Основи створення інтерфейсу в додатках WinUI3.

*Тема лекції 15:* Нові елементи керування, які введені в WinUI3 (TeacherTip, Expander, InfoBar). Застосування об'єктів синхронізації та багатопотоковості при створенні сучасних додатків.

*Тема лабораторного заняття 8:* Створення WinUI3 додатків.

*Самостійна робота здобувача:* Вивчення інтерфейсних елементів у платформі WinUI3.

### ***Тема 14. Створення додатків на платформи WinUI3.***

*Анотація:* Можливості додатків WinUI3 щодо роботи з підключеним обладнанням. Простір імен Windows.Devices та підтримувані типи пристроїв.

*Тема лекції 16:* Робота з пристроями. Простір імен Windows.Devices.

*Самостійна робота здобувача:* Ознайомлення з основами роботи з пристроями USB, пристроями введення, MIDI-пристроями тощо.

## **Модульний контроль 2**

## 5. Індивідуальні завдання

Додаткові опціональні завдання для отримання додаткових балів у рейтингу в рамках лабораторних робіт 2 і 6.

## 6. Методи навчання

Словесні, наочні, практичні; пояснювально-ілюстративні, репродуктивні, частково-пошукові; перевірки та оцінювання знань, умінь і навичок, усного викладу знань, закріплення навчального матеріалу, самостійної роботи з осмислення й засвоєння нового матеріалу.

## 7. Методи контролю

Проведення поточного контролю, електронного тестування, модульного контролю, підсумковий контроль у вигляді іспиту.

## 8. Критерії оцінювання та розподіл балів, які отримують здобувачі освіти

### 8.1. Розподіл балів, які отримують здобувачі освіти

Складові навчальної роботи	Бали за одне заняття (завдання)	Кількість занять (завдань)	Сумарна кількість балів
<b>Змістовий модуль 1</b>			
Лабораторні роботи	0..7	3	0..22
Відвідуваність	0..5	4	0..5
Модульний контроль	0..20	1	0..20
<b>Змістовий модуль 2, 3, 4</b>			
Лабораторні роботи	0..7	4	0..28
Відвідуваність	0..5	4	0..5
Модульний контроль	0..20	1	0..20
<b>Усього за семестр</b>			<b>0..100</b>

Білет для іспиту/заліку складається з двох теоретичних питань. В першому і другому питанні студент повинен продемонструвати теоретичні знання, знання стандартних функцій та операторів мов програмування, та привести приклад програмного коду.

## ***Критерії оцінювання роботи здобувача протягом семестру***

**Задовільно (60-74)** – мати знання і уміння для забезпечення програмних результатів навчання. Показати мінімум знань та умінь. Захистити не менше 60% від усіх завдань лабораторних занять. Вміти створювати додатки на основі проектів у VisualStudio. Розуміти різницю між проектами типу Win32, WPF, UWP, WinUI3. Знати основні відмінності цих типів проектів.

**Добре (75-89)** – мати знання, уміння й навички для забезпечення програмних результатів навчання. Твердо знати мінімум, захистити не менше 80% завдань лабораторних занять. Вміти писати алгоритми роботи з файлами на всіх вивчених платформах. Створювати багатопотокові додатки та синхронізувати їх за допомогою типових об'єктів синхронізації Mutex, Semaphore, Event, CriticalSection/Monitor. Вміти організувати алгоритм роботи мережевого сервера на основі сокетів.

**Відмінно (90-100)** – мати знання, уміння й навички, що дадуть змогу самостійно, вільно та обгрунтовано відповідати на будь-які питання курсу. Здати всі контрольні точки та отримати у рейтингу 90+ балів. Досконально знати всі теми та уміти їх застосовувати.

Таблиця 8.2 – Шкала оцінювання: бальна і традиційна

Сума балів	Оцінка за традиційною шкалою	
	Іспит, диференційований залік	Залік
90 – 100	Відмінно	Зараховано
75 – 89	Добре	
60 – 74	Задовільно	
0 – 59	Незадовільно	Не зараховано

## **9. Політика навчального курсу**

**Відвідування занять.** Регуляція пропусків. Інтерактивний характер курсу передбачає обов'язкове відвідування лабораторних занять. Здобувачі, які за певних обставин не можуть регулярно відвідувати лабораторні заняття, повинні протягом тижня узгодити із викладачем графік індивідуального відпрацювання пропущених занять. Окремі пропущені заняття мають бути відпрацьовані на найближчій консультації протягом тижня після пропуску. Здобувачі, які станом на початок екзаменаційної сесії мають понад 70% невідпрацьованих пропущених занять, до відпрацювання не допускаються.

**Дотримання вимог академічної доброчесності.** Під час вивчення навчальної дисципліни здобувачі мають дотримуватися загальноприйнятих морально-етичних норм і правил поведінки, вимог академічної доброчесності, передбачених Положенням про академічну доброчесність

Національного аерокосмічного університету «Харківський авіаційний інститут» (<https://khai.edu/assets/files/polozhennya/polozhennya-pro-akademichnu-dobrochesnist.pdf>). Очікується, що роботи здобувачів будуть їх оригінальними дослідженнями чи міркуваннями. Відсутність посилань на використані джерела, фабрикування джерел, списування, втручання в роботу інших здобувачів становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в письмовій роботі здобувача є підставою для її незарахування викладачем, незалежно від масштабів плагіату чи обману.

**Вирішення конфліктів.** Порядок і процедури врегулювання конфліктів, пов'язаних із корупційними діями, зіткненням інтересів, різними формами дискримінації, сексуальними домаганнями, міжособистісними стосунками та іншими ситуаціями, що можуть виникнути під час навчання, а також правила етичної поведінки регламентуються Кодексом етичної поведінки в Національному аерокосмічному університеті «Харківський авіаційний інститут» (<https://khai.edu/ua/university/normativna-baza/ustanovchi-dokumenti/kodeks-etichnoi-povedinki/>).

## 10. Методичне забезпечення

Навчально-методичний комплекс дисципліни розміщений у системі управління курсами кафедри комп'ютерних систем, мереж і кібербезпеки.

1. Система управління курсами кафедри комп'ютерних систем, мереж і кібербезпеки [Ел. ресурс]. URL: <https://elearn.csn.khai.edu/>

2. Сторінка дисципліни у системі дистанційного навчання «Ментор» [Ел. ресурс]. URL: <https://mentor.khai.edu/course/view.php?id=1634>.

## 11. Рекомендована література

### Базова

1. Johnson M. Hart – Windows System Programming (4th Edition) – Addison-Wesley Professional: ISBN-13: 978-0321657749, 2010 – 832 p.
2. Jeffrey Richter – Advanced Windows (3rd Edition) – Microsoft Press: ISBN-13: 978-1572315488, 1997 – 1200 p.
3. Chris Sells, Ian Griffiths. Programming WPF: Building Windows UI with Windows Presentation Foundation. CA: O'Reilly Media, Inc., 2007. — 864 p. ISBN 978-0-596-51037-4
4. Alvin Ashcraft, Sergii Baidachnyi. Universal Windows Platform Development with C#. Birmingham, UK: Packt Publishing Ltd., 2016. — 452 p. ISBN 978-1-78588-300-2
5. Alvin Ashcraft. Learn WinUI 3: Leverage WinUI and the Windows App SDK to Create Modern Windows Applications with C# and XAML. Birmingham, UK: Packt Publishing Ltd., October 31, 2023. — 386 p. ISBN 978-1-80512-970-7.

## **12. Інформаційні ресурси**

1. Вступ до Win32 – <http://cppstudio.com/post/9384/>
2. Win32 MSDN - <https://docs.microsoft.com/en-us/windows/desktop/apiindex/windows-api-list>
3. Вступ до UWP - <https://docs.microsoft.com/en-us/windows/uwp/get-started/universal-application-platform-guide>  
UWP for Developers - <https://docs.microsoft.com/en-us/windows/uwp/>
4. Вступ до Win UI 3 - <https://learn.microsoft.com/ru-ru/windows/apps/winui/winui3/>
5. Створення першого додатку Win UI 3 - <https://learn.microsoft.com/ru-ru/windows/apps/get-started/start-here?tabs=winget>
6. SDK для Windows - <https://learn.microsoft.com/ru-ru/windows/windows-app-sdk/api/winrt/>