



**ДЕРЖАВНИЙ ТОРГОВЕЛЬНО-ЕКОНОМІЧНИЙ
УНІВЕРСИТЕТ**
Факультет інформаційних технологій
Кафедра комп'ютерних наук та інформаційних систем

СИЛАБУС (SYLLABUS)
Дисципліна «Технології прикладного програмування /
Applied Programming Technologies»

ІНФОРМАЦІЯ ПРО ВИКЛАДАЧА

Викладач	Томашевська Тетяна Володимирівна
Науковий ступінь	Кандидат технічних наук
Вчене звання	Доцент
Посада	Доцент кафедри комп'ютерних наук та інформаційних систем
Адреса кафедри	м. Київ, вул. Кіото 19, каб. Б-507, Б-525
E-mail	compdep@knute.edu.ua
Консультації	Відповідно до графіку індивідуальних консультацій на сайті кафедри

ПОЛІТИКА АКАДЕМІЧНОЇ ДОБРОЧЕСНОСТІ

<https://knute.edu.ua/file/MzEyMQ==/c12a9f74e87d9154696ca0f761da2e5c.pdf>

Дотримання академічної доброчесності передбачає:

- самостійне виконання навчальних завдань, завдань поточного та підсумкового контролю результатів навчання (для осіб з особливими освітніми потребами ця вимога застосовується з урахуванням їхніх індивідуальних потреб і можливостей);
- посилання на джерела інформації у разі використання не авторських ідей, розробок, тверджень, відомостей і т.п.;
- дотримання норм законодавства про авторське право і суміжні права;
- надання достовірної інформації про результати власної наукової діяльності, використані методики досліджень і джерела інформації.

Порушенням академічної доброчесності вважається:

- академічний плагіат – оприлюднення (частково або повністю) наукових (творчих) результатів, отриманих іншими особами, як результатів власного дослідження (творчості) та/або відтворення опублікованих текстів (оприлюднених творів мистецтва) інших авторів без зазначення авторства;

- самоплагіат – оприлюднення (частково або повністю) власних раніше опублікованих наукових результатів як нових наукових результатів;
- фабрикація – вигадання даних чи фактів, що використовуються в наукових дослідженнях;
- фальсифікація – свідомо зміна чи модифікація вже наявних даних, що стосуються наукових досліджень.

За порушення академічної доброчесності здобувачі освіти можуть бути притягнені до академічної відповідальності:

- повторне проходження оцінювання (модульний контроль, іспит, залік тощо);
- повторне проходження відповідного освітнього компонента освітньо-професійної програми;
- відрахування з Університету;
- позбавлення наданих університетом пільг;
- відмова у присудженні відповідного ступеня вищої освіти;

ПОЛІТИКА ЩОДО ВІДВІДУВАННЯ ЗАНЯТЬ

- відвідування занять є обов'язковим;
- за об'єктивних причин (наприклад, хвороба, міжнародне стажування та ін.) навчання може відбуватись в он-лайн формі за погодженням із викладачем дисципліни.

ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Назва дисципліни / тип дисципліни	Технології прикладного програмування / обов'язкова
Навчальний рік	2023-2024
Факультет	Факультет інформаційних технологій
Курс	3
Семестр	1
Освітній ступінь	Бакалавр
Галузь знань	12 «Інформаційні технології»
Спеціальність	126 «Інформаційні системи та технології»
Загальна характеристика	Кількість годин –180 Кількість кредитів – 6 Види занять: лекції, лабораторні, самостійна робота. Співвідношення аудиторних годин і годин самостійної роботи - 70/110 Мова викладання – українська Форма викладання – очна
Підсумковий контроль	Екзамен
Програмне забезпечення	MS Visual Studio 13 та вище

Обладнання	Проектор, комп'ютерна техніка із встановленим програмним забезпеченням та доступом до мережі Інтернет.
Необхідні попередні дисципліни	Дисципліна «Алгоритмізація та програмування»; дисципліна «Проектування інформаційних систем»
Методика вивчення	Методика вивчення дисципліни полягає у набутті студентами знань теоретичного і практично-прикладного характеру під час лекцій, лабораторних занять, самостійної роботи та вивчення першоджерел і навчально-методичної літератури.
Мета і завдання	<p>Метою вивчення дисципліни «Технології прикладного програмування» є формування у студентів навичок оволодіння технологіями обробки структурованих даних, опанування технологій структурного, модульного та об'єктно-орієнтованого програмування на базі мови програмування C++.</p> <p>Завданням вивчення дисципліни є теоретична та практична підготовка студентів з таких питань: парадигми програмування (структурного, модульного, об'єктно-орієнтованого) та засоби сучасних мов програмування для реалізації різних концепцій; засоби створення ієрархічної об'єктної структури з використанням базового об'єкта, подальше розширення його властивостей з використанням статичних та віртуальних методів; сучасні інструментальні засоби для створення прикладного програмного забезпечення. Автоматизація програмування: технологія RAD (Rapid Application Development); особливості програмування в операційних системах родини Windows, система управління повідомленнями, структура програми для Windows;</p> <p>програмування графічного інтерфейсу користувача, ресурси прикладної програми. Функції API (Application Programming Interface); технологія роботи з даними, використання ADO.NET для обробки баз даних; специфікація COM (Component Object Model) як основа технології створення компонент ActiveX, поняття інтерфейсу компоненти, поліморфізм, таблиця віртуальних функцій, множинні екземпляри; застосування стандартної бібліотеки шаблонів STL при розробці програм; динамічне компонування прикладної програми; супроводження програм, виправлення програмного коду.</p>
Місце дисципліни в освітньо-професійній програмі	
Загальні компетентності	<p>КЗ 2 Здатність застосовувати знання у практичних ситуаціях.</p> <p>КЗ 3 Здатність до розуміння предметної області та професійної діяльності</p> <p>КЗ 8 Здатність оцінювати та забезпечувати якість виконуваних робіт.</p>
Фахові компетентності (результати навчання)	<p>КС 3 Здатність до проектування, розробки, налагодження та вдосконалення системного, комунікаційного та програмно-апаратного забезпечення інформаційних систем та технологій, Інтернету речей (IoT), комп'ютерно-інтегрованих систем та системної мережної структури, управління ними.</p> <p>КС 4 Здатність проектувати, розробляти та використовувати засоби реалізації інформаційних систем, технологій та інфокомунікацій (методичні, інформаційні, алгоритмічні, технічні, програмні та інші).</p> <p>КС 10 Здатність вибору, проектування, розгортання, інтегрування, управління, адміністрування та супроводжування інформаційних систем, технологій та інфокомунікацій, сервісів та інфраструктури організації</p>
Програмні	ПР 3 Використовувати базові знання інформатики й сучасних

результати навчання	<p>інформаційних систем та технологій, навички програмування, технології безпечної роботи в комп'ютерних мережах, методи створення баз даних та інтернет-ресурсів, технології розроблення алгоритмів і комп'ютерних програм мовами високого рівня із застосуванням об'єктно-орієнтованого програмування для розв'язання задач проектування і використання інформаційних систем та технологій.</p> <p>ПР 5 Аргументувати вибір програмних та технічних засобів для створення інформаційних систем та технологій на основі аналізу їх властивостей, призначення і технічних характеристик з урахуванням вимог до системи і експлуатаційних умов; мати навички налагодження та тестування програмних і технічних засобів інформаційних систем та технологій.</p> <p>ПР 6 Демонструвати знання сучасного рівня технологій інформаційних систем, практичні навички програмування та використання прикладних і спеціалізованих комп'ютерних систем та середовищ з метою їх запровадження у професійній діяльності.</p> <p>ПР 7 Обґрунтувати вибір технічної структури та розробляти відповідне програмне забезпечення, що входить до складу інформаційних систем та технологій.</p>
----------------------------	---

ТЕМАТИКА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

РОЗДІЛ 1. Основи програмування за допомогою Application Programming Interface (API)

Тема 1.1. Знайомство з середовищами розроблення програм

Види програмного забезпечення для розроблення програм в операційній системі Windows. Знайомство з інструментальними засобами MS Visual Studio. Майстри для створення проектів різних типів. Поняття проекту в MS Visual C++. Основні структурні компоненти проекту. Аналіз програмного забезпечення, що створюється майстрами MS Visual C++. Компіляція та запуск програм на виконання. Інтерактивні засоби Microsoft Visual C++ для налагодження програм.

Тема 1.2. Основи програмування в операційній системі Windows

Структура програми Application Programming Interface (API) для Windows. Функція WinMain. Вікно як основний об'єкт програми для Windows. Клас вікна та його реєстрація. Типи вікон. Створення вікна програми. Цикл оброблення повідомлень. Механізм передачі повідомлень. Структура віконної процедури.

Тема 1.3. Керування виведенням інформації в системі Windows

Контекст пристрою. Види графічних об'єктів. Режими відображення та різні системи координат. Робота з текстом. Робота з файлами. Функції API для керування графічним виводом.

РОЗДІЛ 2. Розроблення програм у системах Windows

Тема 2.1. Створення MFC-програм

Призначення бібліотеки класів MFC (Microsoft Foundation Classes). Аналіз класів для програм з одним та декількома документами. Призначення та структура карти повідомлень (message maps). Макроси, що використовуються в карті повідомлень. Створення та редагування карти повідомлень за допомогою майстра. Створення функцій для оброблення повідомлень

Windows (Windows message handler). Поняття команди (command). Створення функцій для оброблення команди.

Тема 2.2. Створення графічного інтерфейсу користувача в MFC-програмах

Формування ресурсів вікна діалогу. Створення класу вікна діалогу. Обмін даними між програмою та вікном діалогу за допомогою змінних класу. Робота з елементами управління різних типів. Формування ресурсів меню. Створення функції для оброблення команди меню. Формування ресурсів панелей інструментів. Створення функції для оброблення команди кнопки панелі інструментів.

Тема 2.3. Виведення даних у вікно документа в MFC-програмах

Призначення класу CPaintDC. Клас відображення CView. Використання функцій класу для виведення інформації документа. Робота з графічними об'єктами в MFC-програмах. Графічні примітиви. Призначення полів структури Windows LOGFONT. Використання олівців (pens) та пензликів (brushes).

Тема 2.4. Створення додатку на основі діалогових вікон

Призначення класу додатку CApp, структура файлу оголошення класу та файлу функцій. Призначення та реалізація класів відображення CDialog. Перевантаження функцій та створення повідомлень і функцій обробки повідомлень. Створення власних класів на основі класів MFC. Застосування класів CMetaFileDC збереження документа при роботі додатку).

Тема 2.5. Створення додатку на основі концепції DocView

Призначення класу додатку CApp, класів CMainFrame, CDoc та CView структура файлу оголошення класів та файлу функцій. Реалізація класів відображення CDialog та CView. Створення повідомлень і функцій обробки повідомлень. Створення власних класів на основі класів MFC. Застосування класів похідних вікон CChildFrame та імплементації контейнеру CCtrlItem. Застосування класу CDoc для збереження документа додатку.

Тема 2.6. Робота з базами даних у MFC-програмах

Сучасні технології доступу до реляційних баз даних. Технологія ODBC. Використання класу CDataBase для оброблення бази даних. Створення вікон діалогу для відображення записів таблиць бази даних. Використання класів CRecordset, CRecordView для роботи з записами таблиць бази даних.

Тема 2.7. Основи моделі компонентних об'єктів Microsoft

Основні поняття технології ActiveX. Модель компонентних об'єктів (COM - Component Object Model) як основа технології ActiveX. Поняття компоненти. Архітектура прикладної програми на основі компонент. Поняття інтерфейсу компоненти. Взаємодія з компонентою за допомогою її інтерфейсів. Призначення інтерфейсу IUnknown. Віртуальні функції інтерфейсу IUnknown. Запит інтерфейсу компоненти за допомогою функції QueryInterface. Керування життям компоненти. Функції AddRef та Release.

Тема 2.8. Динамічне компонування програми

Поняття динамічної бібліотеки DLL (Dynamic Link Library). Переваги динамічного компонування програми. Експорт функції із DLL. Розбиття монолітної програми на файли клієнта та компоненти. Розміщення компоненти в динамічній бібліотеці.

Перелік навчальних робіт з дисципліни «Основи теорії інформаційних систем»

Види робіт	К-сть балів
Лабораторне заняття №1. Тема: «Знайомство з середовищами	5

розроблення програм».	
Лабораторне заняття №2. Тема: «Розробка програми з використанням функцій WinAPI».	5
Лабораторне заняття №3. Тема: «Програмування із застосуванням основних елементів управління».	5
Лабораторне заняття №4. Тема: «Створення MFC- програми на основі одновіконного проекту».	5
Лабораторне заняття №5. Тема «Розробка MFC-програми на основі вікна діалогу для демонстрації роботи елементів управління».	5
Лабораторне заняття №6. Тема: «Організація виведення у вікно документа».	5
Лабораторне заняття №7. Тема: «Розробка програми-діалогу з використанням Windows Forms».	5
Лабораторне заняття №8. Тема: «Розробка текстового та графічного редактора з використанням Windows Forms».	5
Лабораторне заняття №9. Тема: «Розробка програмного додатку по обробці БД».	5
Лабораторне заняття №10. Тема: «Створення та застосування компонент засобами бібліо-теки MFC».	5
Лабораторне заняття №11. Тема: «Розробка програми з використанням власної динамічної бібліотеки».	5
Модульний контроль	15
Разом: Аудиторна робота	70
Самостійна робота (СР)	30
Всього:	100

КОНТРОЛЬ ТА КРИТЕРІЇ ОЦІНЮВАННЯ ЗНАТЬ СТУДЕНТІВ

При вивченні дисципліни використовуються наступні форми контролю знань студентів: поточний; модульний; підсумковий.

Поточний контроль передбачає перевірку теоретичних питань, самостійної роботи, практичних робіт та усне опитування по кожній практичній роботі. По даному виду контролю оцінювання знань здійснюється у відповідності до бального розподілу наведеного в попередній таблиці.

Модульний контроль передбачає виконання модульної контрольної роботи. Всі завдання оцінюються в 15 балів. Перше завдання (теоретичне) – 5 балів, друге завдання (практичне) – 5 балів, третє завдання (практичне) – 5 балів.

Формою підсумкового контролю є екзамен. Екзаменаційна оцінка (100 балів) є результатом виконання одного теоретичного питання (40 балів) та двох практичних завдань (2x30=60 балів).

Результуюча оцінка з дисципліни визначається як середня від балів набраних протягом семестру та отриманих на іспиті.

СПИСОК РЕКОМЕНДОВАНИХ ДЖЕРЕЛ

Основний:

1. Fox D. Software Application Development: A Visual C++, MFC, and STL Tutorial. Chapman & Hall/CRC Computer and Information Science, 2018. 1151 p.
2. Gregoire M. Professional C++, 5th Edition. Wrox. 2021. 1285 p.
3. A Tour of C++ (2nd Edition) (C++ In-Depth Series) / by Bjarne Stroustrup. Addison-Wesley Professional, July 9, 2018. 256 p.
4. Halterman R.L. Fundamentals of C++ Programing. School of Computing Southern Adventist University. 2023. 785 p.
5. Hogenson G. C++/CLI: The Visual C++ Language for .Net. Apress, 2016. 414 p