

Міністерство освіти і науки України
Український державний університет імені Михайла Драгоманова

«ЗАТВЕРДЖЕНО»

на засіданні Вченої ради
УДУ імені Михайла Драгоманова
Протокол № 7 від
15.07.2024 2024 р.

Проректор з навчально-
методичної роботи,
професор Роман Вернидуб



ПРОГРАМА
навчальної дисципліни

ПРОГРАМУВАННЯ ГРАФІЧНОГО ІНТЕРФЕЙСУ

Освітньо-професійна програма	<i>Середня освіта (Інформатика)</i>
Галузь знань	<i>01 Освіта / Педагогіка</i>
Спеціальність	<i>014 Середня освіта</i>
Предметна спеціальність	<i>014.09 Інформатика</i>
Рівень вищої освіти	<i>перший (бакалаврський)</i>

Шифр за навчальним планом *ПП 2.12*

Київ – 2024

Розробники програми:

Ігор Твердохліб – доцент кафедри інформаційних технологій і програмування Українського державного університету імені Михайла Драгоманова, кандидат педагогічних наук, доцент

Сергій Оніщенко – аспірант III року навчання спеціальності 014.09 «Середня освіта (Інформатика)» Українського державного університету імені Михайла Драгоманова

Рецензенти:

Петро Малезик – доцент кафедри комп'ютерної та програмної інженерії УДУ імені Михайла Драгоманова, доктор педагогічних наук, доцент

Тарас Сіткар – доцент кафедри комп'ютерних технологій Тернопільського національного педагогічного університету імені Володимира Гнатюка, кандидат педагогічних наук, доцент

Обговорено та рекомендовано до видання на засіданні Науково-методичної ради Українського державного університету імені Михайла Драгоманова «27» лютого 2024 року, протокол № 6.

Голова НМР Університету



Роман Вернидуб

(Ім'я, прізвище)

I. Пояснювальна записка

Програма нормативної навчальної дисципліни «Програмування графічного інтерфейсу» складена відповідно до освітньо-професійної програми «Середня освіта (Інформатика)» підготовки фахівців освітнього рівня бакалавр, галузі знань «01 Освіта / Педагогіка», спеціальності «014.09 Середня освіта (Інформатика)».

Предметом вивчення навчальної дисципліни «Програмування графічного інтерфейсу» є сучасні засоби, прийоми та методи програмування додатків з графічним інтерфейсом.

Міждисциплінарні зв'язки. Дисципліна «Програмування графічного інтерфейсу» розрахована на студентів, які вивчили дисципліни «Основи сучасної інформатики», «Алгоритми і структури даних», «Логічні основи інформатики», «Програмування», «Об'єктно-орієнтоване програмування».

Вивчення курсу забезпечує необхідний рівень знань для опанування наступних розділів програмування: «Створення додатків для мобільних пристроїв» «Кросплатформне програмування».

Мета і завдання навчальної дисципліни.

Метою вивчення дисципліни «Програмування графічного інтерфейсу» є формування у студентів, що навчаються за освітньо-професійною програмою «Середня освіта (Інформатика)», знань про сучасні засоби програмування додатків з графічним інтерфейсом.

Для досягнення мети вивчення дисципліни «Програмування графічного інтерфейсу» потрібно розв'язати такі **завдання**:

- ✓ розкрити місце і значення дисципліни в загальній і професійній освіті;
- ✓ з'ясувати психолого-педагогічні аспекти засвоєння предмету, взаємозв'язки курсу з іншими навчальними дисциплінами циклу програмної інженерії;
- ✓ сприяти розвитку алгоритмічного мислення, навичок системного підходу до вивчення програмування;
- ✓ сформуванню знання студентів про сучасні засоби програмування графічного інтерфейсу;
- ✓ сформуванню у студентів знання, навички та уміння аналізувати зміст поставлених задач, створювати інформаційну модель задачі, формалізувати вхідні та вихідні дані, обирати доцільні структури даних, створювати графічний інтерфейс програмного додатку;
- ✓ сформуванню у студентів уміння створювати додатки з графічним інтерфейсом для розв'язування задач за допомогою комп'ютера.

Основні програмні результати навчання і компетентності згідно з вимогами освітньо-професійної програми:

Результати навчання	Компетентності
<p>ПРН2. Знати методи розробки та дослідження алгоритмів розв'язування задач з інформатики, знати методики оцінювання ефективності алгоритмів; володіти мовами програмування різних видів, розуміти їх переваги для розв'язання базових задач інформатики.</p> <p>ПРН23. Здатність використовувати ІКТ для подання, редагування, збереження та перетворення текстових, числових, графічних, звукових та відео даних на уроці, в позакласній і позашкільній роботі; виважено добирати й застосовувати відповідне програмне забезпечення для розв'язування прикладних задач.</p> <p>ПРН24. Здатність розробляти алгоритми розв'язування задач з інформатики, аналізувати складність та ефективність алгоритмів; реалізовувати алгоритми мовами програмування.</p> <p>ПРН42. Володіння основами професійної мовленнєвої культури.</p> <p>ПРН45. Презентувати, обговорювати та захищати власні погляди в усній, письмовій формах та за допомогою ІКТ.</p> <p>ПРН46. Доносити до фахівців і нефахівців певну інформацію, ідеї, проблеми, рішення та власний досвід у галузі професійної діяльності.</p>	<p>ЗК3. Здатність вчитися та оволодівати сучасними знаннями.</p> <p>ЗК4. Здатність до пошуку, опрацювання, аналізу та критичного оцінювання даних з різних джерел.</p> <p>ЗК11. Здатність грамотно спілкуватися державною мовою як усно, так і письмово.</p> <p>ФК4. Здатність застосовувати теоретичні знання, демонструвати розуміння наукових фактів, концепцій, теорій, принципів і методів, що стосуються інформатики та ІКТ, при вирішенні професійних завдань.</p> <p>ФК7. Здатність оволодівати навичками використання сучасних ІКТ і застосовувати їх в практичній та професійній діяльності.</p>

II. Орієнтовний тематичний план

Дисципліна «Програмування графічного інтерфейсу» за навчальним планом підготовки бакалавра належить до циклу професійної підготовки (нормативна частина), на вивчення якої відводиться 6 кредитів ЄКТС (180 навчальних годин). Значну кількість годин відведено на самостійну навчально-пізнавальну діяльність студентів, що полягає у підготовці до аудиторних занять, виконанні завдань, що пропонуються на лекційних та лабораторних заняттях, виконанні індивідуальних завдань, підготовці до модульного контролю, тощо.

<i>Назви модулів і тем</i>	<i>Кількість годин (денна форма навчання)</i>				<i>Кількість годин (заочна форма навчання)</i>			
	<i>Аудиторні</i>	<i>Лекції</i>	<i>Лабораторні</i>	<i>СРС</i>	<i>Аудиторні</i>	<i>Лекції</i>	<i>Лабораторні</i>	<i>СРС</i>
<i>Всього:</i>	78	38	40	102	20	10	10	160
<i>Модуль 1. Основи програмування мовою C#.</i>								
Тема 1.1. Введення в .NET.	2	2		2				4
Тема 1.2. Основи мови C#.	6	2	4	6	1	1		12
Тема 1.3. Засоби об'єктно-орієнтованого програмування.	8	4	4	8	3	1	2	12
Тема 1.4. Шаблони. Інтерфейси. Події.	8	4	4	8				12
<i>Всього за модулем 1:</i>	24	12	12	24	4	2	2	40
<i>Модуль 2. Створення додатків з графічним інтерфейсом.</i>								
Тема 2.1. Огляд API-інтерфейсів для створення графічних настільних додатків.	2	2		6				8
Тема 2.2. Введення в XAML.	2	2		6				8
Тема 2.3. Особливості створення додатку UWP.	4	2	2	6				10
Тема 2.4. Огляд основних візуальних елементів управління.	4	2	2	6				10
Тема 2.5. Управління компонованням вмісту з використанням панелей.	4	2	2	6	1	1		10
Тема 2.6. Форматування візуальних елементів управління.	4	2	2	6	1	1		10
Тема 2.7. Робота з даними компонентів.	4	2	2	6	1	1		10
Тема 2.8. Введення даних за допомогою компонентів типу кнопок.	6	2	4	6	3	1	2	10
Тема 2.9. Компоненти для виведення та введення текстових даних.	6	2	4	6	3	1	2	10
Тема 2.10. Подання даних у вигляді списку.	6	2	4	8	3	1	2	12
Тема 2.11. Створення меню і панелі інструментів.	6	2	4	8	3	1	2	12
Тема 2.12. Служби візуалізації графіки.	6	4	2	8	1	1		10
<i>Всього за модулем 2:</i>	54	26	28	78	16	8	8	120

На лекційних заняттях розглядаються засоби мови програмування C# для створення додатків з графічним інтерфейсом; систематизуються та узагальнюються знання, навички та уміння, набуті при вивченні суміжних дисциплін.

На лабораторних заняттях приділяється значна увага практичним навичкам застосування засобів мови C# для побудови графічного інтерфейсу при створенні програмних додатків. Метою практичних занять є розвиток у студентів алгоритмічного мислення, навичок та умінь самостійного пошуку та опрацювання інформаційних джерел, доцільного застосування набутих знань.

Вивчення навчальної дисципліни супроводжується використанням навчально-методичної літератури, перелік якої додається, та активним використанням сучасних інформаційних технологій і мережі «Інтернет».

III. Зміст навчальної дисципліни за модулями і темами

Зміст курсу «*Програмування графічного інтерфейсу*» подано у вигляді модулів, до кожного з яких наведено тематику занять та анотації до них.

МОДУЛЬ 1. Основи мови програмування C#.

Тема 1.1. ВВЕДЕННЯ В .NET.

Огляд складових .NET. Поняття загальної системи типів. Встановлення .NET. Огляд середовища Visual Studio .NET. Робота з проектами і рішеннями. Створення додатків мовою C# за допомогою Visual Studio.

Тема 1.2. ОСНОВИ МОВИ C#.

Структура простої програми: з використанням методу Main(); з використанням операторів верхнього рівня. Засоби консольного введення/виведення даних. Системні типи даних і їх псевдоніми. Оголошення і ініціалізація змінних (default, new). Ієрархія класів для типів даних. Члени числових, логічних, символьних типів. Особливості числових літералів. Тип великих чисел BigInteger. Типи DateTime і TimeSpan. Операції даними простих типів. Інструкції управління порядком виконання програми. Особливості роботи з рядками і масивами.

Тема 1.3. ЗАСОБИ ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПРОГРАМУВАННЯ.

Синтаксис оголошення класів. Оголошення об'єктів. Синтаксис оголошення методу. Властивості класу. Перевантаження методів. Конструктори. Деструктори. Інкапсуляція. Успадкування. Поліморфізм. Перевизначення методів. Вкладені класи. Абстрактні класи. Часткові класи. Перевантаження операторів.

Тема 1.4. ШАБЛОНИ. ІНТЕРФЕЙСИ. ПОДІЇ.

Шаблони методів і класів. Обмежені шаблони Анонімні типи. Оголошення, реалізація та використання інтерфейсів. Інтерфейси як параметри. Перевантаження інтерфейсних методів. Делегати. Події. Генерація події. Реєстрація об'єкта як опрацювача події. Використання власних делегатів. Анонімні методи.

МОДУЛЬ 2. Створення додатків з графічним інтерфейсом.

Тема 2.1. ОГЛЯД АРІ-ІНТЕРФЕЙСІВ ДЛЯ СТВОРЕННЯ ГРАФІЧНИХ НАСТІЛЬНИХ ДОДАТКІВ.

Додатки Windows Forms. Структура проекту Windows Forms. Створення простого додатку Windows Forms. Панель інструментів. Візуальний конструктор. Переваги та недоліки Windows Forms. Інтерфейс Windows Presentation Foundation (WPF). Структура проекту Universal Windows Platform (UWP). Використання мови XAML. Створення простого додатку Universal Windows Platform. Переваги використання інтерфейсу WPF.

Тема 2.2. ВВЕДЕННЯ В XAML.

Редактор, простір імен, службові слова XAML. Управління видимістю класів і змінних-членів. Елементи і атрибути. Поняття синтаксиса «властивість-елемент». Поняття уточнених властивостей. Поняття уточненої розмітки.

Тема 2.3. ОСОБЛИВОСТІ СТВОРЕННЯ ДОДАТКУ UWP.

Властивості, методи, події форми. Встановлення властивостей та опрацювачів подій з використанням вікна властивостей. Поняття події. Типи подій. Опрацювання подій в редакторі XAML. Вікно Структура документа. Опрацювання закриття об'єкта Window. Перехоплення подій мишки і клавіатури.

Тема 2.4. ОГЛЯД ОСНОВНИХ ВІЗУАЛЬНИХ ЕЛЕМЕНТІВ УПРАВЛІННЯ.

Компоненти .NET. Простір імен Windows.UI.Xaml.Controls. Основні елементи управління (компоненти форми), для користувацького введення. Бокові елементи вікон і елементів управління. Елементи управління мультимедіа. Елементи управління для компонування. Загальні діалогові вікна.

Тема 2.5. УПРАВЛІННЯ КОМПОНУВАННЯМ ВМІСТУ З ВИКОРИСТАННЯМ ПАНЕЛЕЙ.

Основні елементи управління типу панелей. Позиціонування вмісту панелей Canvas, StackPanel, Grid. Стрічки перегляду панелей. Групування візуальних елементів.

Тема 2.6. ФОРМАТУВАННЯ ВІЗУАЛЬНИХ ЕЛЕМЕНТІВ УПРАВЛІННЯ.

Базові властивості форматування: типи та призначення. Редагування властивостей за допомогою вікна властивостей та редактора коду XAML. Вкладені компоненти. Стили форматування.

Тема 2.7. РОБОТА З ДАНИМИ КОМПОНЕНТІВ.

Способи опрацювання даних компонентів. Опрацювання даних через властивості компонента. Встановлення зв'язку між змінними і компонентами.

Тема 2.8. ВВЕДЕННЯ ДАНИХ ЗА ДОПОМОГОЮ КОМПОНЕНТІВ ТИПУ КНОПОК.

Компоненти: командна кнопка Button, обрання альтернативи RadioButton, множинне обрання CheckBox, автоматичне повторення дій RepeatButton, перемикачі ToggleButton і ToggleSwitch, кнопка-гіперпосилання HyperlinkButton, кнопка для відкриття меню DropDownButton.

Тема 2.9. КОМПОНЕНТИ ДЛЯ ВИВЕДЕННЯ ТА ВВЕДЕННЯ ТЕКСТОВИХ ДАНИХ.

Виведення неформатованого тексту TextBox. Виведення форматованого тексту RichTextBox. Виведення та редагування неформатованого тексту TextBox. Виведення та редагування форматованого тексту RichEditBox. Введення паролю PasswordBox.

Тема 2.10. ПОДАННЯ ДАНИХ У ВИГЛЯДІ СПИСКУ.

Обрання одного елемента зі списку ComboBox. Табличне відображення даних – в рядках і стовпцях, GridView. Обрання елементів зі списку ListBox. Відображення даних у вигляді вертикального списку ListView.

Тема 2.11. СТВОРЕННЯ МЕНЮ І ПАНЕЛІ ІНСТРУМЕНТІВ.

Структурна організація меню. Створення меню. Контекстне меню. Динамічна зміна меню. Створення панелі інструментів. Створення рядку стану.

Тема 2.12. СЛУЖБИ ВІЗУАЛІЗАЦІЇ ГРАФІКИ.

Візуалізація графічних даних за допомогою фігур. Додавання прямокутників, еліпсів і ліній на поверхню Canvas. Параметри фарбування і нанесення ліній. Графічні трансформації. Візуалізація графічних даних з використанням малюнків і геометричних об'єктів. Служби анімації. Використання медіаплеєра.

IV. Засоби діагностики успішності навчання

За результатами роботи на лабораторних заняттях, виконання завдань для самостійного опрацювання, підготовки та виступу з доповіддю на заняттях, модульних тестів, захисту результатів лабораторних робіт, студенти накопичують певну кількість балів, відповідно до якої відбувається оцінювання якості їх навчальних досягнень.

Побудова програми за блочно-модульною схемою спрямована на максимальну індивідуалізацію процесу навчання. Структура програми дібрана так, щоб надати студентам можливість навчатися в індивідуальному темпі та орієнтуватися на певні рівні вимог щодо засвоєння навчального матеріалу.

На лекційних заняттях розглядаються фундаментальні теоретичні питання; систематизуються та узагальнюються знання, навички й уміння, набуті під час вивчення суміжних дисциплін.

На лабораторних заняттях студенти знайомляться із сучасними програмними і апаратними засобами, набувають уміння і навички роботи із ними. Одним із основних завдань під час проведення лабораторного практикуму є набуття умінь та навичок розв'язувати задачі за допомогою комп'ютера. Метою проведення лабораторних занять є розвиток у студентів навичок самостійного використання набутих знань, навичок та умінь і забезпечення засвоєння основних понять навчальної дисципліни.

Викладання навчального курсу проводиться з використанням системи управління навчальними ресурсами MOODLE, де розміщені навчально-методичні та дидактичні матеріали до курсу: список навчально-методичної літератури; конспекти (відеозаписи) лекцій, завдання до лабораторних та самостійних робіт; тестові завдання до теоретичного матеріалу, модульного та підсумкового тестування; доступ до спеціального програмного забезпечення; посилання на ресурси неформальної освіти та ін.

На консультаціях зі студентами передбачається з'ясування і обговорення проблемних питань, що стосуються виконання самостійних завдань до лекційних і лабораторних занять, незрозумілих студентами теоретичних питань тощо. Реалізація вищезгаданих вимог забезпечує один з головних напрямків професійної підготовки сучасного фахівця і дозволяє йому активно використовувати сучасні цифрові технології в професійній діяльності.

Контроль знань студентів здійснюється за модульно-рейтинговою системою. Навчальна діяльність студентів протягом семестру оцінюються за 100-бальною системою відповідно до рейтингового регламенту факультету.

Рейтинг студента з дисципліни складається з балів, які зараховуються за виконання та захист лабораторних робіт і виконання екзаменаційної роботи.

Основним засобом діагностики успішності навчання упродовж семестру є лабораторна робота, яку студент самостійно виконує і захищає на лабораторному занятті.

V. Форма підсумкового контролю успішності навчання – екзамен

Екзамен проводиться у формі письмової екзаменаційної роботи.

Рейтинговий регламент факультету математики, інформатики та фізики. Шкала відповідності

Семестрова складова рейтингової шкали дорівнює 60 балів і визначається як сума балів за виконання та захист лабораторних робіт. Екзаменаційна складова рейтингової шкали дорівнює 40 балів і визначається за результатами екзаменаційної роботи.

Отже, рейтингова шкала з дисципліни визначається як сума семестрової і екзаменаційної складових.

VI. Інформаційні джерела для вивчення курсу

Основні:

1. Коноваленко І.В. Програмування мовою С# 6.0. : навчальний посібник для технічних спеціальностей вищих навчальних закладів. Тернопіль:ТНТУ.2016. 229 с.
2. Коноваленко І.В., Марущак П.О. Платформа .NET та мова програмування С# 8.0: навчальний посібник. Тернопіль: ФОП Паляниця В.А., 2020. 320 с.

Додаткові:

3. Фленов М. Библия С#. 3-е изд., перераб. и доп. СПб.: БХВ-Петербург, 2016. 544 с.
4. Фленов М. Библия С#. 4-е изд., перераб. и доп. СПб.: БХВ-Петербург, 2019. 512 с.
5. Троелсен Э., Джепикс Ф. Язык программирования С# 9 и платформа .NET 5: основные принципы и практики программирования, том 1, 10-е изд. пер. с англ. Ю.Н. Артеменко. Киев. : «Диалектика», 2022. 770 с.
6. Троелсен Э., Джепикс Ф. Язык программирования С# 9 и платформа .NET 5: основные принципы и практики программирования, том 2, 10-е изд. пер. с англ. Ю.Н. Артеменко. Киев. : «Диалектика», 2022. 632 с.

Інтернет ресурси:

7. С# Підручник. URL: https://w3schoolsua.github.io/cs/cs_intro.html#gsc.tab=0
8. Програмуємо на С++, С#, Java... URL: <https://programer.in.ua/index.php/pochatkiivtsiu/rozrobka-ihor-dlia-pochatkiivtsiv-na-c>
9. Документація по С#. URL: <https://learn.microsoft.com/en-us/dotnet/csharp/tour-of-csharp/>
10. Елементи керування інтерфейсу користувача та класи, які підтримують наявні та спеціальні елементи керування UWP. URL: <https://learn.microsoft.com/uk-ua/uwp/api/windows.ui.xaml.controls?view=winrt-22621>