



СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ

| | |
|--|--|
| Галузь знань | 12 Інформаційні технології |
| Шифр та назва спеціальності | 121 Інженерія програмного забезпечення |
| Назва освітньо-професійної програми | Освітньо-професійна програма підготовки «Інженерія програмного забезпечення» |
| Рівень вищої освіти | Перший (бакалаврський) |
| Факультет | Інформаційних технологій та кібербезпеки |
| Кафедра | Інженерії програмного забезпечення |
| Статус навчальної дисципліни | ОК-15 ОПП Освітньо-професійна програма підготовки «Інженерія програмного забезпечення» |
| Форма навчання | Денна |

Викладачі

Косирева Ліаліна Анатоліївна
lialina@ukr.net

Багачук Денис Геннадійович
bagachukdg@gmail.com



Старший викладач кафедри Інженерії програмного забезпечення

Доцент кафедри Інженерії програмного забезпечення,
кандидат технічних наук

Загальна інформація про дисципліну

| | |
|-------------------------------|---|
| Анотація до дисципліни | Курс надає майбутнім фахівцям теоретичні знання та практичні навички з об'єктно-орієнтованого програмування та дозволяє сформулювати найважливіші практичні вміння з використанням актуальних підходів до розробки програмних засобів в різних галузях сучасного суспільства. Вивчається одна з найпоширеніших у світі мов об'єктно-орієнтованого програмування – мова C#. |
| Мета дисципліни | Метою викладання навчальної дисципліни є формування базових знань про сучасні технології об'єктно-орієнтованого програмування, практичне засвоєння основних понять і концепцій об'єктно-орієнтованого програмування. |
| Компетентності, | СК-13. Здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення. |

| | |
|--|--|
| формуванню яких сприяє дисципліна | СК-19. Володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних. СК-24. Здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення. СК-26. Здатність до алгоритмічного та логічного мислення. |
| Результати навчання | ПРН-1 Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки. ПРН-3 Знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення. ПРН-5 Знати і застосовувати відповідні математичні поняття, методи доменного, системного й об'єктно-орієнтованого аналізу та математичного моделювання для розробки програмного забезпечення. ПРН-15 Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення. |
| Обсяг дисципліни | Загальний обсяг дисципліни: 8 кредитів ЄКТС (240 годин). Для денної форми навчання: лекції – 28 години, практичні заняття – 30 години; лабораторні заняття – 30 години, самостійна робота – 152 години. |
| Форма підсумкового контролю | Залік. Екзамен. Курсова робота |
| Терміни викладання дисципліни | Дисципліна викладається у 1-му та 2-му семестрах 2-го року навчання |

Програма дисципліни

| | |
|----------------|---|
| Тема 1. | <i>Мова C#. Консольні прикладні проекти C#</i> Співвідношення між мовами програмування C, C++ і C#. Вікно консолі. Типи даних та оператори в C#. Тестування та налагодження програм мовою C#. Одновимірні і багатовимірні масиви. Масиви масивів. Цикли foreach у масивах. Функції у мові C#. Передача аргументів за значенням і за посиланням у функції. Рядки. Клас string та його методи. |
| Тема 2. | <i>Класи та об'єкти у мові C#</i> Класи, об'єкти та властивості (properties) у мові C#. Доступ до членів класу. Методи, їх параметри. Ініціалізація екземплярів класів. Специфікатори доступу. Конструктори та деструктори. Ключове слово this. Перевантаження операцій. Формат перевантаження для унарного та бінарного операторів. |
| Тема 3. | <i>Успадковування та інкапсуляція, поліморфізм</i> Ієрархії нащадків. Управління механізмом доступу до членів базового класу. Особливості використання конструкторів і деструкторів при реалізації механізму успадкування. Виклик затінених методів і базового конструктора. Перекриття методів. Використання модифікатора типу static. Віртуальні методи класів. Поняття поліморфного коду. Використання вказівників на похідні типи як основа динамічного поліморфізму. Абстрактні класи. Використання ключового слова sealed для відвертання спадкоємства. |

| | |
|----------------|---|
| | Властивості. Індексатори. Створення одновимірних та багатовимірних індексаторів. Перевантаження індексаторів. |
| Тема 4. | <i>Візуальне програмування</i> Візуальний інтерфейс середовища Visual Studio .NET. Палітра компонент. Вікно властивостей. Створення віконного додатку. Властивості та методи форми. Події, пов'язані з формою. Основні компоненти. Додатки з кількома формами, додатки MDI. Компоненти-контейнери. Створення і використання динамічних компонент. Підключення подієвих функцій до динамічних компонент. Події клавіатури. |
| Тема 5. | <i>Введення і виведення даних у файли, списки</i> Класи для введення і виведення даних. Класи File і Directory. Класи FileInfo і DirectoryInfo. Імена шляху і відносні шляхи. Класи StreamWriter і StreamReader. Об'єкти цих класів. Клас FileStream. Позиція усередині файлу. Читання і запис у файл. Робота з BinaryWriter і BinaryReader. Компоненти dataGridView. Списки. Черга, стік. Клас List. Клас Stack. Клас універсальної колекції Collection. Клас DateTime для роботи з датами та часом. |
| Тема 6. | <i>Делегати. Події. Інтерфейси</i> Статичні та нестатичні методи. Делегати. Додавання методів в делегат. Події. Додавання обробників подій. Обробка черги повідомлень. Оголошення та реалізація інтерфейсів. Використання інтерфейсних посилань. Інтерфейсні властивості. Перевантаження інтерфейсних методів. Успадковування інтерфейсів. Явна реалізація членів інтерфейсу. Вибір між інтерфейсом і абстрактним класом. Додаток на програмування подій. |
| Тема 7. | <i>Виняткові ситуації та власні компоненти. Патерни проектування</i> Виняткові ситуації і відповідні їм класи у C#. Оператори try та finally. Використання декількох catch-блоків. Ієрархія класів виняткових ситуацій. Створення і використання бібліотек класів. Приклади побудови бібліотек. Клас Component. Побудова нащадків візуальних компонент. Додавання властивостей і обробників подій. Патерни. Види патернів. Патерни, що породжують, структурні та поведінкові патерни. |
| Тема 8. | <i>Побудова графічних зображень у C#</i> Графічний інструментарій C#. Простір імен System.Drawing. Клас Graphics. Ієрархії класів. Побудова графічних зображень з використанням ієрархії класів. Приклад ієрархії класів для геометричних зображень. Візуалізація руху. |

Список рекомендованих джерел

1. В.В.Бублик – Об'єктно-орієнтоване програмування, К.: ІТкнига, 2015. – 624 с.
2. Косирева Л. А. Електронний конспект лекцій з дисципліни «Алгоритмізація та програмування» - Одеса: ОНУ, 2021.
3. Косирева Л.А., Буката Л.М. Об'єктно-орієнтоване програмування. Конспект лекцій з дисципліни "Об'єктно-орієнтоване програмування", ч.1. - Одеса: ДУІТЗ, 2024, 123 с.
4. Коноваленко І.В. Програмування мовою C# 7.0: навчальний посібник / Коноваленко І.В., Марущак П.О., Савків В.Б. Тернопіль : Тернопільський національний технічний університет імені Івана Пулюя, 2017. 300 с. URL: <http://elartu.tntu.edu.ua/handle/lib/22436>.
5. Об'єктно-орієнтоване програмування. Частина 1. Основи об'єктно-орієнтованого програмування на мові C#: Навчальний посібник. / Д.В. Настенко, А. Б. Нестерко. – К.: НТУУ «КПІ», 2016. – 76 с. URL: https://ela.kpi.ua/bitstream/123456789/16671/1/OOP_manual.pdf.

6. Збірник задач з програмування / Глазунова Л.В. та ін. – Одеса: ОНАЗ, 2011.
7. Troelsen A., Pro C# and the .NET Framework – Апресс, 2012. - 1487 с.

Інформація про консультації

Щосереди на протязі семестра з 13⁰⁰ до 14²⁰ год., конференція ZOOM ст. викладача Косиревої Л.А.

Загальна схема оцінювання

| Сума балів за всі види навчальної діяльності | Шкала ЄКТС | Оцінка за національною шкалою | | Нарахування балів | Бали нараховуються таким чином: |
|--|------------|--|---|-------------------|--|
| | | для іспиту | для заліку | | |
| 90-100 | A | Відмінно | зараховано | | <i>Оцінювання знань здобувачів вищої освіти здійснюється за 100-бальною шкалою і становить: за поточну успішність (участь у практичних заняттях, виконання практичних завдань та контрольних робіт) – до 60 балів, за результати екзамену – до 40 балів.</i> |
| 82-89 | B | Добре | | | |
| 74-81 | C | | | | |
| 64-73 | D | | | | |
| 60-63 | E | Задовільно | | | |
| 35-59 | FX | Незадовільно з можливістю повторного складання | Не зараховано з можливістю повторного складання | | |
| 0-34 | F | Незадовільно з обов'язковим повторним вивченням дисципліни | Не зараховано з обов'язковим повторним вивченням дисципліни | | |

Політика опанування дисципліни

Відвідування: Присутність на лекціях практичних, лабораторних заняттях та контрольних заходах (екзамен/залік) є обов'язковою. Відсутність на заняттях може бути лише з поважних причин.

Дотримання принципів академічної доброчесності: Підготовка усіх завдань, письмових робіт і т. ін., що виконуються в межах дисципліни, здійснюється здобувачем вищої освіти самостійно, на засадах академічної доброчесності. Викладач має право для перевірки робіт застосовувати програму **Unichack**.

Умови зарахування пропущених занять: Зарахування пропущених практичних та лабораторних занять проводяться під час консультацій.

Інші умови: Навчально-методичні матеріали дисципліни розміщені на платформі Moodle, за посиланням <http://e-learning2.suitt.edu.ua/course/view?id=844>.