

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет інформаційних технологій Кафедра комп'ютерної інженерії та інформаційних систем



ЗАТВЕРДЖУЮ
Говорущенко Т.О.
1 вересня 2024 р.

Загальна інформація

| Позиція | Зміст інформації |
|---------------------------|---|
| Викладач(і) | Лисенко Сергій Миколайович |
| Профайл викладача | http://kiis.khmnu.edu.ua/personnel/lysenko-sergij-mykolajovych/ |
| E-mail викладача(ів) | sprlysenko@gmail.com |
| Контактний телефон | заповнюється за домовленістю |
| Сторінка дисципліни в ІСУ | https://msn.khnu.km.ua/course/view.php?id=6406 |
| Навчальний рік | 2024-2025 |
| Консультації | Очні: середа, 6-а пара, 1-114; п'ятниця, 6-а пара, 1-114; онлайн: за необхідністю та попередньою домовленістю |

Характеристика дисципліни

| Форма навчання | Курс | Семестр | Загальне навантаження | | Кількість годин | | | | | | Форма семестрового контролю | | | |
|------------------|------|---------|-----------------------|------------|-------------------|-----------|--------------------|-------------------|-------------------------------|-------------------------------|-----------------------------|----------------|----------|-------|
| | | | Кредити ЄКТС | Години | Аудиторні заняття | | | | Індивідуальна робота студента | Самостійна робота, в т.ч. ІРС | Курсовий проект | Курсова робота | Залік | Іспит |
| | | | | | Разом | Лекції | Лабораторні роботи | Практичні заняття | | | | | | |
| Д | 2 | парний | 8.0 | 240 | 108 | 36 | 36 | 36 | | 132 | - | - | + | - |
| Разом ДФН | | | 8.0 | 240 | 108 | 36 | 36 | 36 | | 132 | | | 1 | |

Анотація дисципліни

Дисципліна "Об'єктно-орієнтоване проектування" є вибірковою з програмування та характеризується широким міждисциплінарним підходом.

Мета і завдання дисципліни

Дисципліна “Об’єктно-орієнтоване проектування” є однією зі спеціальних профілюючих дисциплін і тому займає провідне місце у підготовці бакалаврів з програмування та характеризується широким міждисциплінарним підходом.

Метою курсу є оволодіння студентами основними принципами проектування та розроблення програмного забезпечення на основі об’єктно-орієнтованого підходу мовою С#, які в подальшому можуть сприяти їх успішному застосуванню в професійній діяльності.

Завдання дисципліни “Об’єктно-орієнтоване проектування” навчити застосовувати принципи алгоритмічного та об’єктно-орієнтованого програмування на всіх етапах життєвого циклу прикладної програмної системи, починаючи з аналізу вимог до програмної системи і її попереднього проектування, і закінчуючи її реалізацією, тестуванням і наступним супроводом.

Очікувані результати навчання.

Студент, який успішно завершив вивчення дисципліни, повинен: використовувати сучасні методи і мови програмування для розроблення алгоритмічного та об’єктно-орієнтованого програмного забезпечення; створювати системне та прикладне програмне забезпечення комп’ютерних систем та мереж мовою С#; забезпечувати проектування та розроблення якісних програмних засобів; розробляти програмне забезпечення для вбудованих і розподілених застосунків, мобільних і гібридних систем; використовувати навички програмування, технології розроблення алгоритмів і комп’ютерних програм мовами високого рівня із застосуванням об’єктно-орієнтованого програмування для розв’язання задач проектування і використання програмно-технічних засобів комп’ютерних систем та мереж.

Тематичний і календарний план вивчення дисципліни

| № тижня | Тема лекції* | Тема практичного заняття* | Тема лабораторної роботи* | Самостійна робота студентів | | |
|---------|--|--|--|---|------|------------|
| | | | | Зміст | Год. | Література |
| 1. | Загальна структура програми на С#. Масиви. Рядки. Читання з файлу та запис у файл. Класи та структури. Інкапсуляція. | Загальна структура програми на С#. Метод Main() і аргументи командного рядка. Масиви. Типи. Рядки. в Оператори та вирази в С#. Вивчення основних операцій з файлами мови С# [1,4]. | Загальна структура програми на С#. Метод Main() і аргументи командного рядка. Масиви. Типи. Рядки. в Оператори та вирази в С#. Вивчення основних операцій з файлами мови С# [1,4]. | Опрацювання лекційного матеріалу. Підготовка до практичної роботи №1 | 7 | [1-10] |
| 2. | Рядки. Оголошення та ініціалізація рядків. Правильні та буквальні рядкові літери | Загальна структура програми на С#. Метод Main() і аргументи командного рядка. Масиви. Типи. Рядки. в Оператори та вирази в С#. Вивчення основних операцій з файлами мови С# [1,4]. | Загальна структура програми на С#. Метод Main() і аргументи командного рядка. Масиви. Типи. Рядки. в Оператори та вирази в С#. Вивчення основних операцій з файлами мови С# [1,4]. | Опрацювання лекційного матеріалу. | 7 | [1-10] |
| 3. | Введення в С#. Мова С# і платформа .NET Літ.: [1,5,7,8, 9, 10] | Класи та структури. Об’єкти. Екземпляри структури і екземпляри класу. Ідентифікація | Класи та структури. Об’єкти. Екземпляри структури і екземпляри | Опрацювання лекційного матеріалу. Підготовка до захисту практичної роботи №1. | 7 | [1-10] |

| | | | | | | |
|----|--|---|--|---|---|--------|
| | | <p>об'єктів і рівність значень. Члени. Статичні члени. Властивості. Властивості, що реалізуються автоматично. Методи. Доступ до методів. Передача за посиланням та передача за значенням. Значення, що повертаються. Конструктори. Конструктори екземплярів. Закриті конструктори. Статичні конструктори. Створення конструктора копії. Деструктори. [1-4].</p> | <p>класу. Ідентифікація об'єктів і рівність значень. Члени. Статичні члени. Властивості. Властивості, що реалізуються автоматично. Методи. Доступ до методів. Передача за посиланням та передача за значенням. Значення, що повертаються. Конструктори. Конструктори екземплярів. Закриті конструктори. Статичні конструктори. Створення конструктора копії. Деструктори. [1-4].</p> | <p>Підготовка до практичної роботи №2</p> | | |
| 4. | <p>Інкапсуляція. Константи і поля для читання Літ.: [3,4,5,8,11]</p> | <p>Класи та структури. Об'єкти. Екземпляри структури і екземпляри класу. Ідентифікація об'єктів і рівність значень. Члени. Статичні члени. Властивості. Властивості, що реалізуються автоматично. Методи. Доступ до методів. Передача за посиланням та передача за значенням. Значення, що повертаються. Конструктори. Конструктори екземплярів. Закриті конструктори. Статичні конструктори. Створення</p> | <p>Класи та структури. Об'єкти. Екземпляри структури і екземпляри класу. Ідентифікація об'єктів і рівність значень. Члени. Статичні члени. Властивості. Властивості, що реалізуються автоматично. Методи. Доступ до методів. Передача за посиланням та передача за значенням. Значення, що повертаються. Конструктори. Конструктори екземплярів. Закриті</p> | <p>Опрацювання лекційного матеріалу.</p> | 7 | [1-10] |

| | | | | | | |
|----|---|---|---|--|---|--------|
| | | конструктора копії. Деструктори. [1-4]. | конструктори. Стагичні конструктори. Створення конструктора копії. Деструктори. [1-4]. | | | |
| 5. | Наслідування. Доступ до членів базового класу з класу-наслідника | Наслідування класів. Модифікатор new. Ключове слово base. Модифікатор sealed. Ключове слово this. Абстрактні та віртуальні методи. Абстрактні та заповані класи та члени класів. Заповані класи та члени класів. Віртуальні методи. Інтерфейси. Явна реалізація інтерфейса. Поліморфізм в C#. Заповання перевизначення віртуальних членів похідними класами. Доступ до віртуальних членів базового класу з похідних класів [2,6,7,10]. | Наслідування класів. Модифікатор new. Ключове слово base. Модифікатор sealed. Ключове слово this. Абстрактні та віртуальні методи. Абстрактні та заповані класи та члени класів. Заповані класи та члени класів. Віртуальні методи. Інтерфейси. Явна реалізація інтерфейса. Поліморфізм в C#. Заповання перевизначення віртуальних членів похідними класами. Доступ до віртуальних членів базового класу з похідних класів [2,6,7,10]. | Опрацювання лекційного матеріалу. Підготовка до захисту практичної роботи №2. Підготовка до практичної роботи №3 | 7 | [1-10] |
| 6. | Перетворення типів Літ.: [1, 5, 9] | Наслідування класів. Модифікатор new. Ключове слово base. Модифікатор sealed. Ключове слово this. Абстрактні та віртуальні методи. Абстрактні та заповані класи та члени класів. Заповані класи та члени класів. Віртуальні методи. Інтерфейси. Явна реалізація інтерфейса. | Наслідування класів. Модифікатор new. Ключове слово base. Модифікатор sealed. Ключове слово this. Абстрактні та віртуальні методи. Абстрактні та заповані класи та члени класів. Заповані класи та члени класів. Віртуальні методи. | Опрацювання лекційного матеріалу. | 7 | [1-10] |

| | | | | | | |
|----|--|---|---|---|---|---------|
| | | <p>Поліморфізм в C#. Запобігання перевизначення віртуальних членів похідними класами. Доступ до віртуальних членів базового класу з похідних класів [2,6,7,10].</p> | <p>Інтерфейси. Явна реалізація інтерфейса. Поліморфізм в C#. Запобігання перевизначення віртуальних членів похідними класами. Доступ до віртуальних членів базового класу з похідних класів [2,6,7,10].</p> | | | |
| 7. | <p>Делегати, події і лямбди Літ.: [2,3,4]</p> | <p>Індексатори. Індексатори в інтерфейсах. Порівняння властивостей та індексаторів. Делегати. Анонімні методи. Делегати з іменованими методами і делегати з анонімними методами. Перевантаження операторів. [2,6,7,10].</p> | <p>Індексатори. Індексатори в інтерфейсах. Порівняння властивостей та індексаторів. Делегати. Анонімні методи. Делегати з іменованими методами і делегати з анонімними методами. Перевантаження операторів. [2,6,7,10].</p> | <p>Опрацювання лекційного матеріалу. Підготовка до захисту практичної роботи №3. Підготовка до практичної роботи №4</p> | 7 | [11-12] |
| 8. | <p>Анонімні метод. Лямбда-вирази в обробці подій Літ.: [2,5,8]</p> | <p>Індексатори. Індексатори в інтерфейсах. Порівняння властивостей та індексаторів. Делегати. Анонімні методи. Делегати з іменованими методами і делегати з анонімними методами. Перевантаження операторів. [2,6,7,10].</p> | <p>Індексатори. Індексатори в інтерфейсах. Порівняння властивостей та індексаторів. Делегати. Анонімні методи. Делегати з іменованими методами і делегати з анонімними методами. Перевантаження операторів. [2,6,7,10].</p> | <p>Опрацювання лекційного матеріалу.</p> | 7 | [11-12] |
| 9. | <p>Колекції. Введення в колекції Літ.: [1,2]</p> | <p>Введення в універсальні шаблони. Універсальні типи під час виконання. Обмеження параметрів типу. Застосування обмежень. Універсальні класи.</p> | <p>Введення в універсальні шаблони. Універсальні типи під час виконання. Обмеження параметрів типу. Застосування обмежень.</p> | <p>Опрацювання лекційного матеріалу. Підготовка до захисту практичної роботи №4. Підготовка до практичної роботи №5</p> | 7 | [1-12] |

| | | | | | | |
|-----|--|--|--|---|---|--------|
| | | <p>Універсальні інтерфейси. Універсальні методи. Універсальні делегати. System.Collections. Generic - простір імен. IList<T> - інтерфейс. [1-4,10].</p> | <p>Універсальні класи. Універсальні інтерфейси. Універсальні методи. Універсальні делегати. System.Collections.Generic - простір імен. IList<T> - інтерфейс. [1-4,10].</p> | | | |
| 10. | <p>Колекції. Введення в колекції Літ.: [4,5,8]</p> | <p>Введення в універсальні шаблони. Універсальні типи під час виконання. Обмеження параметрів типу. Застосування обмежень. Універсальні класи. Універсальні інтерфейси. Універсальні методи. Універсальні делегати. System.Collections.Generic - простір імен. IList<T> - інтерфейс. [1-4,10].</p> | <p>Введення в універсальні шаблони. Універсальні типи під час виконання. Обмеження параметрів типу. Застосування обмежень. Універсальні класи. Універсальні інтерфейси. Універсальні методи. Універсальні делегати. System.Collections.Generic - простір імен. IList<T> - інтерфейс. [1-4,10].</p> | <p>Опрацювання лекційного матеріалу.</p> | 7 | [1-12] |
| 11. | <p>Робота з потоками і файлової системою Літ.: [1,2,4,7,8]</p> | <p>Анонімні функції. Лямбда-вирази. Лямбди операторів. Лямбди зі стандартними операторами запитів. Виведення типу в лямбда-виразах. Область дії змінної в лямбда-виразах. Анонімні методи. Ітератори. Використання виключних ситуацій. Ключові слова try, catch и finally. [1,9,10].</p> | <p>Анонімні функції. Лямбда-вирази. Лямбди операторів. Лямбди зі стандартними операторами запитів. Виведення типу в лямбда-виразах. Область дії змінної в лямбда-виразах. Анонімні методи. Ітератори. Використання виключних ситуацій. Ключові слова</p> | <p>Опрацювання лекційного матеріалу. Підготовка до захисту практичної роботи №5. Підготовка до практичної роботи №6</p> | 7 | [1-12] |

| | | | | | | |
|-----|--|---|---|---|---|--------|
| | | | try, catch и finally. [1,9,10]. | | | |
| 12. | Серіалізація. Введення в серіалізацію об'єктів. Літ.: [1,9,10]. | Анонімні функції. Лямбда-вирази. Лямбди операторів. Лямбди зі стандартними операторами запитів. Виведення типу в лямбда-виразах. Область дії змінної в лямбда-виразах. Анонімні методи. Ітератори. Використання виключних ситуацій. Ключові слова try, catch и finally. [1,9,10]. | Анонімні функції. Лямбда-вирази. Лямбди операторів. Лямбди зі стандартними операторами запитів. Виведення типу в лямбда-виразах. Область дії змінної в лямбда-виразах. Анонімні методи. Ітератори. Використання виключних ситуацій. Ключові слова try, catch и finally. [1,9,10]. | Опрацювання лекційного матеріалу. | 7 | [1-12] |
| 13. | Основи LINQ Літ.: [5,7]. | Обробка подій. Робота з формами та компонентами (Windows Form Application) [1,2,6,7]. | Обробка подій. Робота з формами та компонентами (Windows Form Application) [1,2,6,7]. | Опрацювання лекційного матеріалу. Підготовка до практичної роботи №7. Підготовка до захисту практичної роботи №6. | 8 | [1-12] |
| 14. | Введення в Entity Framework 6. Взаємодія з даними. Що таке Entity Framework. Перше додаток. Підхід Code First. Code First до існуючої бази даних. Угоди по найменуванню в Code First. Автоматизація Code First. Автоматизація Code First і EF Power Tools. Database First. Model First. Літ.: [5,7]. | Обробка подій. Робота з формами та компонентами (Windows Form Application) [1,2,6,7]. | Обробка подій. Робота з формами та компонентами (Windows Form Application) [1,2,6,7]. | Опрацювання лекційного матеріалу. | 8 | [1-12] |
| 15. | Основи Entity Framework. Основні операції з даними. Операції з даними. практичний приклад. Рядок підключення. Навігаційні властивості і завантаження даних. | Обробка подій. Робота з формами та компонентами з використанням графіки (Windows Form Application) [6,9,10,11] | Обробка подій. Робота з формами та компонентами з використанням графіки (Windows Form Application) [6,9,10,11] | Опрацювання лекційного матеріалу. Підготовка до практичної роботи №7. | 8 | [1-12] |

| | | | | | | |
|-----|--|--|--|--|---|--------|
| | Зв'язок один до одного. Зв'язок один до багатьох. Зв'язок один до багатьох. практичний приклад. Зв'язок багато до багатьох. Зв'язок багато до багатьох. практичний приклад. • Ініціалізація бази даних. Паралелізм в Entity Framework. Управління транзакціями. • Міграції. Generic Repository [1,2,6,7] | | | | | |
| 16. | LINQ to Entities. Введення в LINQ to Entities. Вибірка і проєкція з бази даних. Сортування. З'єднання таблиць. Угрупування. Операції з множинами: об'єднання, перетин, різниця. Агрегатні операції. IEnumerable і IQueryable. AsNoTracking Літ.: [3,4,8] | Обробка подій. Робота з формами та компонентами з використанням графіки (Windows Form Application) [6,9,10,11] | Обробка подій. Робота з формами та компонентами з використанням графіки (Windows Form Application) [6,9,10,11] | Опрацювання лекційного матеріалу. | 8 | [1-12] |
| 17. | Робота з SQL. Збережені функції. Збережені процедури. Літ.: [4,6,7,8] | Підсумкове заняття | Підсумкове заняття | Опрацювання лекційного матеріалу. Підготовка до захисту практичної роботи №8. Індивідуальне оцінювання проєкту колеги. Колективне оцінювання проєкту одного з колег. | 8 | [1-12] |
| 18. | Збирання сміття, управління пам'яттю і вказівники. Збиральник сміття в C#. Літ.: [10-14] | Підсумкове заняття | Підсумкове заняття | Опрацювання лекційного матеріалу. | 8 | [1-12] |

Примітка: * Лекції, практичні заняття проводяться по дві години; послідовність проведення занять визначається розкладом (може не відповідати нумерованим тижням)

Політика дисципліни.

Організація освітнього процесу з дисципліни відповідає вимогам положень про організаційне і навчально-методичне забезпечення освітнього процесу, освітній програмі та навчальному плану. Студент зобов'язаний відвідувати лекції, лабораторні заняття згідно з розкладом, не запізнюватися на заняття, завдання виконувати відповідно до графіка. Пропущене

лабораторне заняття студент зобов'язаний опрацювати самостійно у повному обсязі і відзвітувати перед викладачем не пізніше, ніж за тиждень до чергової атестації. До лабораторних занять студент має підготуватися за відповідною темою і проявляти активність. Набуті особою знання з дисципліни або її окремих розділів у неформальній освіті зараховуються відповідно до Положення про порядок перезарахування результатів навчання у ХНУ.

Критерії оцінювання результатів навчання.

Кожний вид роботи з дисципліни оцінюється за **чотирибальною** шкалою. Семестрова підсумкова оцінка визначається як середньозважена з усіх видів навчальної роботи, виконаних і зданих позитивно з врахуванням коефіцієнта вагомості. Вагові коефіцієнти змінюються залежно від структури дисципліни і важливості окремих її видів робіт. При оцінюванні знань студентів використовуються різні засоби контролю, зокрема: усне опитування перед допуском до виконання лабораторної роботи – здійснюється на їх початку; засвоєння теоретичного матеріалу з тем перевіряється тестовим контролем; якість виконання, набуття теоретичних знань і практичних навичок перевіряється шляхом захисту кожної лабораторної роботи та індивідуального завдання згідно з робочою програмою дисципліни і робочим навчальним планом.

Структурування дисципліни за видами робіт і оцінювання результатів навчання студентів денної форми навчання у семестрі за ваговими коефіцієнтами

| | | | |
|---------------------|-----|----------------------------------|------------------------------|
| Аудиторна робота | | Самостійна, індивідуальна робота | Семестровий контроль (залік) |
| V семестр | | | |
| Лабораторні роботи: | | Практичні роботи: | Оцінювання проектів |
| 1-8 | | 1-8 | контрольна робота |
| БК: | 0,4 | 0,2 | 0,4 |
| | | | за рейтингом |
| | | | 0 |

Примітка: БК – ваговий коефіцієнт.

Співвідношення інституційної шкали оцінювання і шкали оцінювання ЄКТС

| Оцінка ЄКТС | Інституційна шкала балів | Інституційна оцінка | Критерії оцінювання | |
|-------------|--------------------------|---------------------|---------------------|---|
| A | 4,75-5,00 | 5 | Зараховано | Відмінно – глибоке і повне опанування навчального матеріалу і виявлення відповідних умінь та навичок. |
| B | 4,25-4,74 | 4 | | Добре – повне знання навчального матеріалу з кількома незначними помилками. |
| C | 3,75-4,24 | 4 | | Добре – в загальному правильна відповідь з двома-трьома суттєвими помилками. |
| D | 3,25-3,74 | 3 | | Задовільно – неповне опанування програмного матеріалу, але достатнє для практичної діяльності за професією. |
| E | 3,00-3,24 | 3 | | Задовільно – неповне опанування програмного матеріалу, що задовольняє мінімальні критерії оцінювання |
| FX | 2,00-2,99 | 2 | Незараховано | Незадовільно – безсистемність одержаних знань і неможливість продовжити навчання без додаткових знань з дисципліни |
| F | 0,00-1,99 | 2 | | Незадовільно – необхідна серйозна подальша робота і повторне вивчення дисципліни. |

Питання для підсумкового контролю з дисципліни

1. Загальні відомості про програми на C#. Структура проектів в C#.
2. Елементи мови. Службові слова. Імена. Коментарі.
3. Розміщення даних у пам'яті. Тип даних. Адресний вираз.
4. Функції. Область існування імені. Область видимості та простору імен.
5. Типи. Базові стандартні типи. Константи. Змінні.
6. Операції C#. Арифметичні операції. Операції присвоєння. Операції відношення та еквівалентності. Логічні операції.
7. Операції адресації та непрямой адресації.
8. Пріоритет і асоціативність операцій. Перевантаження операцій.
9. Масиви у C#. Операції з масивами. Правила роботи з масивами.
10. Структури та об'єднання. Структури в стилі C#.

11. Об'єднання. Об'єднання, що не мають імені. Доступ до членів-даних структури, об'єднання.
12. Екземпляри структури, об'єднання. Масиви структур.
13. Вказівники, зсилки та функції в C#.
14. Локальний та динамічний розподіл пам'яті.
15. Оператори new і delete. Динамічне розміщення об'єктів та вказівників.
16. Правила роботи з вказівниками. Розіменування вказівників.
17. Оператори new[], оператори delete[].
18. Функції в C#. Передача аргументів та повернення результату.
19. Застосування при передачі параметрів специфікації const. Параметри зі значеннями по умовчанняю.
20. Перегрузка функцій. Приведення типів.
21. Функції-члени класу. Передача у функції змінного числа параметрів.
22. Вбудовані функції inline.
23. Шаблони функцій.
24. Область видимості функцій. Правила, що визначають область видимості.
25. Функція main(). Аргументи функції main. Аргументи командного рядка.
26. Основні поняття ООП. Інкапсуляція, поліморфізм, наслідування і засоби їх реалізації.
27. Об'єкт, властивості, методи, події.
28. Класи. Визначення класу. Ідентифікатори класу. Тіло класу.
29. Приховування інформації. Ієрархії класів та наслідування.
30. Інкапсуляція. Керування доступом до класу. Приватні, загальнодоступні, захищені члени класу.
31. Класи пам'яті для об'єктів класів. Область видимості класу. Порожні класи.
32. Вкладені класи. Правила доступу для вкладених класів.
33. Екземпляри класу. Використання членів даних. Статичні члени-данні.
34. Об'єкти класу як члени даних. Вказівники як члени даних.
35. Використання функцій-членів. Прості, статичні функції члени.
36. Конструктори. Деструктори.
37. Дружні функції. Властивості дружніх функцій.
38. Шаблони класів. Вкладені шаблонні класи.
39. Наслідування. Обмеження наслідування. в C#. Просте наслідування.
40. Специфікатори доступу базових класів. Класи для наслідування. Аргументи передані в базовий клас.
41. Порядок виклику конструкторів. Порядок виклику деструкторів.
42. Перетворення типів у похідних класах. Дозвіл області видимості.
43. Використання віртуальних базових класів. Використання віртуальних і невіртуальних базових класів разом.
44. Використання перетворення типів. Правила виклику функцій базових класів.
45. Використання дозволу видимості при множинному наслідуванні.
46. Поліморфізм. Раннє та пізнє зв'язування. Віртуальні функції. Перевизначення функції.
47. Абстрактні класи. Обмеження віртуальних функцій. Віртуальні оператори.
48. Поліморфізм при простому наслідуванні. Поліморфізм при множинному наслідуванні.
49. Виклик поліморфних функцій базового класу. Віртуальні функції та ієрархії класифікацій. Виклик віртуальних функцій у конструкторах.
50. Потік C#. Потоки як узагальнені фільтри. Стандартний потоковий ввід-вивід класів користувача.
51. Потоки і файловий ввід-вивід. Використання текстових файлів для введення. Вивід текстових файлів.
52. Обробка подій. Об'єктно-орієнтована програма, яка керована подіями. Методи опрацювання подій.
53. Шаблони функцій і класів. Основні властивості шаблонів класів. Компонентні функції параметризованих класів.
54. Стандартна бібліотека шаблонів. Класи-контейнери. Асоціативні контейнери. Алгоритми.

9. МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ

Навчальний процес з дисципліни забезпечений необхідними навчально-методичними розробками в модульному середовищі.

10. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. Технічна документація ASP.NET Core Microsoft. <https://docs.microsoft.com/uk-ua/aspnet/core/?view=aspnetcore-6.0>.
2. Дженніфер Грін, Ендрю Стеллман. Head First C#: Посібник для навчання з програмування в реальному світі за допомогою C# та .NET Core 4th Edition. O'Reilly Media. 2021 рік с.800 (англ).
3. Джозеф Альбахарі, Бен Альбахарі. C# 8.0 Pocket Reference: миттєва допомога для програмістів C# 8.0, 1-е видання. O'Reilly Media. 2019 . с.248 (англ).
4. Джозеф Альбахарі, Ерік Йогансен. C# 8.0 in Nutshell: The Definitive Reference 1st Edition. O'Reilly Media. 2020. с. 1088(англ).
5. Ian Griffiths. Programming C# 8.0: Build Windows, Web, and Desktop Applications 1st Edition. O'Reilly Media. 2020. С.802 (англ).
6. Коноваленко І.В. Програмування мовою C# 6.0. Тернопіль, ТНТУ. 2016. С.227.

7. Д. В. Настенко, А. Б. Нестерко. ОБ'ЄКТНО-ОРИЄНТОВАНЕ ПРОГРАМУВАННЯ Частина 1. Основи об'єктно-орієнтованого програмування на мові С#. Київ НТУУ «КПІ» 2016 с.76.
8. Павлишко А.В. Савельєва О.В. АЛГОРИТМІЗАЦІЯ ТА ПРОГРАМУВАННЯ. Одеса .2017 с.123.
9. І. О. ЗАВАДСЬКИЙ, Р. І. ЗАБОЛОТНИЙ , Основи візуального програмування. Київ Видавнича група ВНУ. С.271.
10. Коноваленко І.В., Марущак П.О., Савків В.Б. Програмування мовою С# 7.0. Тернопіль 2017. с.302.
11. Joseph Albahari. С# 7.0 Pocket Reference: Instant Help for С# 7.0 Programmers 1st Edition. O'Reilly Media. 2017. С.240 (англ).
12. Joseph Albahari. С# 10 in a Nutshell. The Definitive Reference. O'Reilly Media.2022. с.1058 (англ)
13. Джозеф Альбахарі, Бен Альбахарі. Кишеньковий довідник С# 10. Миттєва допомога для програмістів С# 10. 1-е видання. O'Reilly Media. 2022. С.240 (англ)
14. Joseph Albahari. С# 9.0 in a Nutshell: The Definitive Reference. O'Reilly Media. 2021. С.1060 (англ)
15. Jeffrey Richter. CLR via С# (Developer Reference). Microsoft Press. С.894 (англ)
16. Електронна бібліотека університету. Доступ до ресурсу: http://lib.khnu.km.ua/asp/php_fpage_lib.php.

Розробник:



д.т.н., проф. Лисенко С.М.

Погоджено:

Зав. каф. КПС:



к.т.н., доц. Засорнова І.О.