МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЗАХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

ЗАТВЕРДЖУЮ
В. о. декана факультету комп'ютерних інформаційних технологій
_____ Ігор Якименко
"_____"_____2023 р.

ЗАТВЕРДЖУЮ
В. о. проректора з науково-педагогічної роботи
_____ Віктор ОСТРОВЕРХОВ
"_____"_____2023 р.

ЗАТВЕРДЖУЮ
Директор навчально наукового інституту новітніх освітніх технологій
_____ Святослав ПИТЕЛЬ
"____"_____2023 р.

# РОБОЧА ПРОГРАМА
## з дисципліни «Об'єктно-орієнтоване програмування»

Ступінь вищої освіти: бакалавр
Галузь знань – 12 «Інформаційні технології»
Спеціальність – 121 «Інженерія програмного забезпечення»
Освітньо-професійна програма – «Інженерія програмного забезпечення»

Кафедра комп'ютерних наук

| Форма навчання | Курс | Семестр | Лекції (год.) | Лаб. (год.) | ІРС (год.) | Тренінг, КПІЗ (год.) | СРС (год.) | Разом (год.) | Екз. (сем.) |
|---|---|---|---|---|---|---|---|---|---|
| Денна | II | 2,3 | 58 | 58 | 7 | 16 | 101 | 240 | 2 (залік), 3 (екз.) |
| Заочна | II | 2,3,4 | 16 | 8 | - | - | 216 | 240 | 3 (залік), 4 (екз.) |

Тернопіль – ЗУНУ
2023

Робоча програма складена на основі освітньо-професійної програми підготовки бакалаврів галузі знань 12 «Інформаційні технології» спеціальності 121 «Інженерія програмного забезпечення», затвердженої Вченою радою ЗУНУ №___ від _____2023 р.

Робоча програма розроблена доцентом кафедри комп'ютерних наук, к.т.н. Володимиром МАНЖУЛОЮ

Робоча програма затверджена на засіданні кафедри комп'ютерних наук, протокол №1 від 28 серпня 2023р.

Завідувач кафедри д.т.н, професор _____Андрій ПУКАС

Розглянуто та схвалено групою забезпечення спеціальності 121 Інженерія програмного забезпечення, протокол №___ від _____ 2023р.

Голова групи
забезпечення спеціальності,
д.т.н., професор _____Микола ДИВАК

Гарант ОП,
к.т.н., доцент _____ Світлана КРЕПИЧ

**СТРУКТУРА РОБОЧОЇ ПРОГРАМИ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

**“Об’єктно-орієнтоване програмування”**

## 1. Опис дисципліни “Об’єктно-орієнтоване програмування”

| Дисципліна – «Об’єктно-орієнтоване програмування» | Галузь знань, напрям підготовки, ступінь вищої освіти | Характеристика навчальної дисципліни |
|---|---|---|
| Кількість кредитів – 8 | Галузь знань: 12 Інформаційні технології | Статус дисципліни: нормативна<br>Мова навчання: українська |
| Кількість залікових модулів:<br>2 семестр – 3;<br>3 семестр – 4. | Спеціальність: 121 Інженерія програмного забезпечення | Рік підготовки:<br>Денна – 1, 2.<br>Заочна – 1, 2.<br>Семестр:<br>Денна – 2, 3.<br>Заочна –2, 3, 4. |
| Кількість змістових модулів – 6 | Ступінь вищої освіти: бакалавр | Лекції:<br>Денна – 58 год.<br>Заочна – 16 год.<br>Лабораторні заняття:<br>Денна – 58 год.<br>Заочна – 8 год. |
| Загальна кількість годин – 240 | | Самостійна робота:<br>Денна – 101 год.<br>Тренінг – 8<br>Заочна – 216 год.<br>Індивідуальна робота:<br>Денна – 7 год. |
| Тижневих годин: 13 год., з них аудиторних:<br>2 семестр – 5 год.;<br>3 семестр – 3 год. | | Вид підсумкового контролю:<br>Денна – залік, 2; екзамен, 3;<br>Заочна – залік, 3; екзамен, 4. |

## 2. Мета й завдання вивчення дисципліни «Об’єктно-орієнтоване програмування»

### 2.1. Мета вивчення дисципліни

Мета викладання дисципліни “Об’єктно-орієнтоване програмування” полягає у наданні теоретичних знань та практичних навиків програмування з використанням об’єктно-орієнтованих мов програмування.

Мета проведення лекцій полягає у тому, щоб ознайомити студентів із головними методологічними та методичними питаннями об’єктно-орієнтованої парадигми програмування, та інструментарієм для створення об’єктно-орієнтованих програм від моделювання взаємозв’язків між об’єктами до створення програмного коду засобами об’єктно-орієнтованих мов програмування C++ та C#.

Мета проведення лабораторних занять полягає у тому, щоб виробити у студентів практичні навички побудови об’єктно-орієнтованих моделей та їх програмної реалізації засобами об’єктно-орієнтованої мов програмування C++ та C#.

**2.2. Завдання вивчення дисципліни**

Завданням дисципліни є набуття студентами теоретичних знань та практичних навичок принципів об’єктно-орієнтованої декомпозиції складних об’єктів, розробки проектів та написанні на їх основі ефективних програм з використанням всіх переваг об’єктно-орієнтованої парадигми програмування.

У результаті вивчення дисципліни студенти повинні знати:

- фундаментальні концепції ООП;
- принципи програмування та реалізації ООП.
- мови об’єктно-орієнтованого програмування C++ та C#;
- перевантаження операторів;
- наслідування;
- поліморфізм
- шаблони;
- методи обробка виняткових ситуацій;
- ієрархію класів;
- організацію стандартної бібліотеки і контейнерів;

У результаті вивчення дисципліни студенти повинні вміти:

- визначати класи та об’єкти при застосуванні об’єктно-орієнтованих технологій програмування;
- здійснювати об'єктно-орієнтовану декомпозицію предметної області в умовах програмування складних об'єктів і систем за допомогою процедур об'єктно-орієнтованого аналізу, об'єктно-орієнтованого програмування, використовуючи визначення класів, поведінки об'єктів, структури даних та їх взаємозв'язки;
- розробляти об'єктно-орієнтовану модель предметної галузі за допомогою мов об'єктно-орієнтованого моделювання в умовах програмування складних об'єктів та систем, використовуючи об'єктно-орієнтовану нотацію складних систем.

**2.3. Найменування та опис компетентностей, формування котрих забезпечує вивчення дисципліни:**

ФК11. Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення.

ФК13. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення

**2.4. Передумови для вивчення дисципліни.**

Дисципліни, які повинні бути вивчені попередньо:

- «Основи інженерії програмного забезпечення»;
- «Основи програмування»;

**2.5. Результати навчання**

У результаті вивчення курсу «Об’єктно-орієнтоване програмування» студенти повинні:

РН03. Знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення.

РН06. Уміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення.

РН14. Застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення.

РН15. Мотивовано обирати мови програмування та технології розробки для розв’язання завдань створення і супроводження програмного забезпечення.

РН18. Знати та вміти застосовувати інформаційні технології обробки, зберігання та передачі даних.

РН22. Знати та вміти застосовувати методи та засоби управління проектами.

## 3. Зміст дисципліни “Об’єктно-орієнтоване програмування”

### *Змістовий модуль 1.* Основні парадигми ООП.

**Тема 1. Вступ в об’єктно–орієнтоване програмування.**

Історія об'єктно-орієнтованого програмування. Основні поняття. Абстракція даних. Наслідування, інкапсуляція та поліморфізм.

Література: 1,2,10.

**Тема 2. Обєкти та класи.**

Поняття об’єкта в ООП. Стан та поведінка об’єкта. Життєвий цикл об’єкта. Відношення між об’єктами. Поняття класу. Типи класів. Відношення між класами. Інстанціювання.

Література: 1,2,9,10.

**Тема 3. Об’єктно-орієнтований аналіз і декомпозиція складних об’єктів.**

Об’єктно-орієнтований аналіз і декомпозиція складних об’єктів. Об’єктно-орієнтований синтез інформаційних моделей. Використання мови об’єктно-орієнтованого моделювання UML.

Література: 1,2,9,10.

### *Змістовий модуль 2.* Реалізація принципів ООП мовою С++.

**Тема 4. Класи та об’єкти в С++.**

Класи, члени класу. Ініціалізація. Інтерфейс та реалізація. Конструктори і деструктори. Конструювання і знищення об’єктів. Вказівник this. Вказівники на члени класу. Статичні змінні класу. Статичні методи класу. Поняття дружності.

Література: 2,3,6,7,17.

**Тема 5. Наслідування.**

Похідні класи. Модифікатори наслідування. Розширення та обмеження спеціалізації класів Конструктори та деструктори похідних класів. Виклик конструкторів похідних класів. Виклик конструкторів класів ієрархії. Множинне наслідування. Виклик конструкторів при множинному наслідуванні. Передача параметрів конструктору базового класу.

Література: 2,3,6,7,17.

**Тема 6. Віртуальні функції та поліморфізм.**

Віртуальні функції та поліморфізм. Поліморфні об’єкти та контейнери. Чисто віртуальні функції та абстрактні класи. Механізм пізнього зв’язування.

Література: 2,3,6,7,17.

**Тема 7. Перевантаження операторів.**

Перевантаження операторів. Створення операторної функції члена. Створення префіксної і постфіксної форм операторів інкрементації і декрементації. Перевантаження скорочених операторів присвоєння. Обмеження на перевантаження операторів. Перевантаження операторів new і delete. Перевантаження індексаторів. Перевантаження оператор «()». Перевантаження оператора «->«. Перевантаження оператора «,».

Література: 2,3,6,7,17.

**Тема 8. Шаблони.**

Простий шаблон. Шаблони-списки. Параметризовані функції. Правила ототожнення перевантажених параметризованих функцій. Наслідування і шаблони. Узагальнені класи. Застосування аргументів за умовчанням в шаблонних класах. Явні спеціалізації класів.

Література: 2,3,6,7,17.

**Тема 9. Обробка виняткових ситуацій.**

Обробка помилок. Розпізнавання ситуацій. Класифікація ситуацій. Використання ресурсів. Генерація виняткових ситуації. Повторне генерування виняткової ситуації. Функції terminate() і unexpected(). Обробники, пов'язані з функціями terminate() і unexpected(). Класи exception і bad_exception. Перехоплення класів виняткових ситуацій. Застосування декількох операторів catch. Обробка похідних виняткових ситуацій.

Література: 2,3,6,7,17.

***Змістовий модуль 3*. Бібліотека стандартних шаблонів STL.**

**Тема 10. Вступ в STL.**

Склад STL. Предикати та функції-об’єкти. Алокатори. Ітератори. Поняття ітератора.

Література: 2,3,6,7,17.

**Тема 11. Контейнерні класи.**

Поняття контейнера. Типи контейнерів. Контейнер vector. Контейнер deque. Асоціативні контейнери. Контейнер Set. Контейнер Map.

Література: 2,3,6,7,17.

**Тема 12. Алгоритми STL.**

Модифікуючі та немодифікуючі операції. Алгоритми сортування. Алгоритми роботи з множинами.

Література: 2,3,6,7,17.

**Тема 13. Оптимізація обчислень в STL.**

Шаблон класу valarray. Перетин масивів.

Література: 2,3,6,7,17.

***Змістовий модуль 4*. Основи C # і платформа .NET.**

**Тема 14. Мова C # і платформа .NET**

Створення C #. Зв'язок C # з середовищем .NET Framework. Загальномовне виконуюча середу CLR. Загальна система типів CTS.

Загальномовне специфікація CLS. Збірки. Проміжний мова CIL. Класи і простору імен. Загальномовна інфраструктура CLI.

Література: 12,13,14,23,24.

**Тема 15. Основи C #**

Проста програма на C #. Коментарі. Змінні. Область видимості змінних, константи. Типи даних. Літерали. Перетворення типів. Арифметичні оператори. Оператори відносини і логічні оператори. Оператор присвоювання. Порозрядні оператори. Тернарний оператор. Умовні оператори. Циклічні оператори: for, while, do while і foreach. Оператори переходу.

Література: 12,13,14,23,24.

**Тема 16. Масиви, кортежі і рядки**

Масиви. Багатовимірні масиви. Ступінчасті (зубчасті) масиви масивів. Клас Array. Масиви в якості параметрів. Кортежі. Рядки. Клас StringBuilder. Форматуючі рядки. Регулярні вирази.

Література: 12,13,14,23,24.

***Змістовий модуль 5*. Реалізація принципів ООП мовою C#.**

**Тема 17. Класи і об'єкти в C#.**

Класи. Клас Object. Створення об'єктів. Методи. Конструктори. Збірка сміття і деструктори. Ключове слово this. Доступ до членів класу. Модифікатори параметрів. Необов'язкові та іменовані аргументи. Рекурсія. Ключове слово static. Індексатори. Властивості. Модифікатори доступу в аксесорах.

Література: 12,13,14,23,24.

**Тема 18. Перевантаження методів класу**

Перевантаження методів. Перевантаження конструкторів. Перевантаження індексаторів. Основи перевантаження операторів. Перевантаження операторів відношення і операторів true – false. Перевантаження логічних операторів. Оператори перетворення.

Література: 12,13,14,23,24.

**Тема 19. Успадкування та поліморфізм**

Основи успадкування. Захищений доступ і виключення успадкування. Конструктори та успадкування. Успадкування та приховування імен. Посилання на базовий клас і об'єкти похідних класів. Віртуальні методи, властивості і індексатори. Абстрактні класи.

Література: 12,13,14,23,24.

**Тема 20. Інтерфейси, структури та перерахування**

Інтерфейси. Інтерфейсні посилання. Інтерфейсні властивості і індексатори. Спадкування інтерфейсів. Явна реалізація інтерфейсу. Структури. Перерахування.

Література: 12,13,14,23,24.

***Змістовий модуль 6*. Програмування обробки винятків та подій в C#.**

**Тема 21. Обробка винятків в C#**

Основи обробки винятків. Приховати винятків. Клас Exception. Конфігурування стану виключення. Винятки рівня системи та програми. Обробка численних винятків. Оператори throw і finally. Винятки, пов'язані з

пошкодженим станом (Corrupted State Exceptions). Ключові слова checked і unchecked.

Література: 12,13,14,23,24.

**Тема 22. Делегати та лямбда-вирази**

Делегати. Груповий виклик і адресація делегованих методів. Коваріантність і контраваріантних делегатів. Делегати Action <T> і Func <T>. Анонімні методи. Лямбда-вирази.

Література: 12,13,14,23,24.

**Тема 23. Події**

Події. Аксессор подій. Обробка подій в середовищі .NET Framework.

Література: 12,13,14,23,24.

**Тема 24. Узагальнення**

Огляд узагальнень. Узагальнені класи. Обмежені типи. Обмежені класи. Обмежені інтерфейси і конструктори. Обмеження посилального типу і типу значення. Ієрархії узагальнених класів. Засоби узагальнень. Узагальнені методи. Узагальнені структури. Узагальнені делегати. Узагальнені інтерфейси. Модифікація узагальнених методів. Коваріантність і контраваріантних в узагальненнях.

Література: 12,13,14,23,24.

## 4. Структура залікового кредиту з дисципліни "Об'єктно-орієнтоване програмування"

(Денна форма навчання)

| | Кількість годин | | | | | |
|---|---|---|---|---|---|---|
| | Лекції | Лабораторні заняття | Самостійна робота | ІРС | Тренінг | Контрольні заходи |
| Змістовий модуль 1. Основні парадигми ООП. | | | | | | |
| Тема 1. Вступ в об'єктно–орієнтоване програмування | 2 | | 4 | | 2 | Усне опитування та тестування |
| Тема 2. Об'єкти та класи | 2 | 2 | 4 | | | Усне опитування та тестування |
| Тема 3. Об'єктно-орієнтований аналіз і декомпозиція складних об'єктів | 2 | 4 | 4 | | | Усне опитування та тестування |
| Змістовий модуль 2. Реалізація принципів ООП мовою С++. | | | | | | |
| Тема 4. Класи та об'єкти в С++ | 2 | 4 | 4 | 2 | 2 | Усне опитування та тестування |
| Тема 5. Наслідування | 2 | 4 | 4 | | | Усне опитування та тестування |

| | | | | | | |
|---|---|---|---|---|---|---|
| Тема 6. Віртуальні функції та поліформізм | 2 | 4 | 4 | | | Усне опитування та тестування |
| Тема 7. Перевантаження операторів | 2 | 4 | 4 | | | Усне опитування та тестування |
| Тема 8. Шаблони | 2 | 4 | 4 | | | Усне опитування та тестування |
| Тема 9. Обробка виняткових ситуацій | 2 | 4 | 4 | | | Усне опитування та тестування |
| Змістовий модуль 3. Бібліотека стандартних шаблонів STL. | | | | | | |
| Тема 10. Вступ в STL | 2 | 3 | 4 | 2 | 4 | Усне опитування та тестування |
| Тема 11. Контейнерні класи | 4 | 4 | 4 | | | Усне опитування та тестування |
| Тема 12. Алгоритми STL | 4 | 4 | 4 | | | Усне опитування та тестування |
| Тема 13. Оптимізація обчислень в STL | 2 | 4 | 5 | | | Усне опитування та тестування |
| Змістовий модуль 4. Основи C # і платформа .NET. | | | | | | |
| Тема 14. Мова C # і платформа .NET | 2 | | 5 | 1 | 2 | Усне опитування та тестування |
| Тема 15. Основи C # | 2 | 2 | 5 | | | Усне опитування та тестування |
| Тема 16. Масиви, кортежі і рядки | 4 | 2 | 5 | | | Усне опитування та тестування |
| Змістовий модуль 5. Реалізація принципів ООП мовою C#. | | | | | | |
| Тема 17. Класи і об'єкти в C# | 2 | 1 | 5 | 1 | 4 | Усне опитування та тестування |
| Тема 18. Перевантаження методів класу | 2 | 1 | 5 | | | Усне опитування та тестування |
| Тема 19. Успадкування та поліморфізм | 4 | 2 | 5 | | | Усне опитування та тестування |

| | | | | | | |
|---|---|---|---|---|---|---|
| Тема 20. Інтерфейси, структури та перерахування | 2 | 1 | 5 | | | Усне опитування та тестування |
| Змістовий модуль 6. Програмування обробки винятків та подій в C#. | | | | | | |
| Тема 21. Обробка винятків в C# | 2 | 1 | 5 | 1 | 2 | Усне опитування та тестування |
| Тема 22. Делегати та лямбда-вирази | 2 | 1 | 5 | | | Усне опитування та тестування |
| Тема 23. Події | 2 | 1 | 5 | | | Усне опитування та тестування |
| Тема 24. Узагальнення | 4 | 2 | 5 | | | Усне опитування та тестування |
| Разом | 58 | 58 | 101 | 7 | 16 | |

(Заочна форма навчання)

| | Кількість годин | | |
|---|---|---|---|
| | Лекції | Лабораторні заняття | Самостійна робота |
| Змістовий модуль 1. Основні парадигми ООП. | | | |
| Тема 1. Вступ в об’єктно–орієнтоване програмування | 1 | | 8 |
| Тема 2. Об’єкти та класи | | | 8 |
| Тема 3. Об’єктно-орієнтований аналіз і декомпозиція складних об’єктів | 1 | 1 | 8 |
| Змістовий модуль 2. Реалізація принципів ООП мовою C++. | | | |
| Тема 4. Класи та об’єкти в C++ | 1 | 1 | 8 |
| Тема 5. Наслідування | 1 | 1 | 8 |
| Тема 6. Віртуальні функції та поліформізм | 1 | 1 | 8 |
| Тема 7. Перевантаження операторів | | | 10 |
| Тема 8. Шаблони | 1 | | 10 |
| Тема 9. Обробка виняткових ситуацій | | | 10 |
| Змістовий модуль 3. Бібліотека стандартних шаблонів STL. | | | |
| Тема 10. Вступ в STL | | | 10 |
| Тема 11. Контейнерні класи | 1 | | 10 |
| Тема 12. Алгоритми STL | 1 | | 10 |
| Тема 13. Оптимізація обчислень в STL | | | 10 |

| | | | |
|---|---|---|---|
| Змістовий модуль 4. Основи С # і платформа .NET. | | | |
| Тема 14. Мова С # і платформа .NET | 1 | | 10 |
| Тема 15. Основи С # | 1 | | 10 |
| Тема 16. Масиви, кортежі і рядки | 1 | 1 | 10 |
| Змістовий модуль 5. Реалізація принципів ООП мовою С#. | | | |
| Тема 17. Класи і об'єкти в С# | 1 | 1 | 10 |
| Тема 18. Перевантаження методів класу | | | 10 |
| Тема 19. Успадкування та поліморфізм | 1 | 1 | 10 |
| Тема 20. Інтерфейси, структури та перерахування | 1 | 1 | 10 |
| Змістовий модуль 6. Програмування обробки винятків та подій в С#. | | | |
| Тема 21. Обробка винятків в С# | 1 | | 10 |
| Тема 22. Делегати та лямбда-вирази | 1 | | 10 |
| Тема 23. Події | | | 10 |
| Тема 24. Узагальнення | | | 10 |
| Разом | 16 | 8 | 216 |

## 5. Тематика лабораторних занять

Лабораторна робота № 1

Тема: Об’єктно-орієнтований аналіз і декомпозиція складних об’єктів.

Мета: Навчитись будувати моделі складних об’єктів з використанням мови об’єктно-орієнтованого моделювання UML.

Література: 1, 2.

Лабораторна робота № 2

Тема: Конструювання об’єктів конкретних класів.

Мета: Засвоєння методики створення конкретних класів з використанням мови програмування С++.

Література: 2, 4, 12, 17.

Лабораторна робота № 3

Тема: Реалізація наслідування.

Мета: Засвоєння методики реалізації наслідування з використанням мови програмування С++.

Література: 2, 10, 12, 17.

Лабораторна робота № 4

Тема: Поліморфізм.

Мета: Засвоєння методики реалізації поліморфізму з використанням мови програмування С++.

Література: 2, 10, 12, 17.

Лабораторна робота № 5

Тема: Перевантаження операторів.

Мета: Засвоєння методики перевантаження унарних та бінарних операторів для класів користувача.
Література: 2, 6, 11, 17.

Лабораторна робота № 6

Тема: Розробка шаблонів.
Мета: Навчитись використовувати можливості параметричного поліморфізму.
Література: 2, 10, 12, 17.

Лабораторна робота № 7

Тема: Обробка виняткових ситуацій.
Мета: Засвоєння методики відлагодження програм та обробки програм на етапі виконання засобами об'єктно-орієнтованого програмування.
Література: 4, 10, 12, 17.

Лабораторна робота № 8

Тема: Використання STL.
Мета: Засвоєння принципів роботи з STL.
Література: 7, 10, 12, 17.

Лабораторна робота № 9

Тема: Основи C# та .NET.
Мета: Навчитися створювати консольні додатки мовою C# на платформі .NET.
Література: 8, 9, 12, 16.

Лабораторна робота № 10

Тема: Використання масивів, кортежів і рядків у C#.
Мета: Навчитися використовувати масиви, кортежі і рядки у C#.
Література: 2, 3, 11, 17.

Лабораторна робота № 11 (4/1год.)

Тема: Конструювання об'єктів класів у C#.
Мета: Засвоєння методики створення конкретних класів з використанням мови програмування C#.
Література: 4, 8, 12, 13.

Лабораторна робота № 12

Тема: Реалізація наслідування.
Мета: Засвоєння методики реалізації наслідування з використанням мови програмування C#.
Література: 2, 10, 12, 17.

Лабораторна робота № 13

Тема: Поліморфізм.
Мета: Засвоєння методики реалізації поліморфізму з використанням мови програмування C#.
Література: 3, 11, 12, 15.

Лабораторна робота № 14

Тема: Перевантаження методів на C#.

Мета: Засвоєння методики перевантаження унарних та бінарних операторів для класів корисувача.
Література: 2, 10, 12, 17.

Лабораторна робота № 15

Тема: Розробка інтерфейсів.

Мета: Навчитись використовувати можливості поліморфізму на основі інтерфейсів.
Література: 3, 4, 14.

Лабораторна робота № 16

Тема: Обробка виняткових ситуацій на C#.

Мета: Засвоєння методики відлагодження програм та обробки програм на етапі виконання засобами об'єктно-орієнтованого програмування на C#.
Література: 5, 6, 7.

Лабораторна робота № 17

Тема: Використання делегатів.

Мета: Засвоєння принципів роботи з делегатами на C#.
Література: 2, 10, 12, 17.

Лабораторна робота № 18

Тема: Обробка подій.

Мета: Навчитися реалізовувати обробку подій в середовищі .NET. Framework.
Література: 7, 8, 10, 14.

## 6. Комплексне практичне індивідуальне завдання (9 год.)

Індивідуальні завдання з дисципліни виконується самостійно кожним студентом на основі заданого алгоритму. КПІЗ охоплює усі основні теми дисципліни. Метою виконання КПІЗ є оволодіння навичками застосування методів побудови об'єктно-орієнтованої моделі предметної області та її програмної реалізації методами об'єктно-орієнтованого програмування. КПІЗ оформлюється о відповідності з встановленими вимогами. Виконання КПІЗ з одним із обов'язкових складових модулів залікового кредиту з дисципліни.

Варіанти КПІЗ з дисципліни "Об'єктно-орієнтоване програмування"

1. Реалізація об'єктно-орієнтованої програми для імітації роботи ліфта.
2. Реалізація об'єктно-орієнтованої програми для імітації роботи музичного центру.
3. Реалізація об'єктно-орієнтованої програми для імітації роботи холодильника.
4. Реалізація об'єктно-орієнтованої програми для імітації роботи пральної машина.
5. Реалізація об'єктно-орієнтованої програми для імітації роботи охоронної системи приміщення (в тому числі - протипожежна).
6. Реалізація об'єктно-орієнтованої програми для імітації роботи телефонного апарату.

7. Реалізація об’єктно-орієнтованої програми для імітації роботи системи забезпечення безпеки руху.

8. Реалізація об’єктно-орієнтованої програми для імітації роботи автоматичної системи контролю стану систем літака.

9. Реалізація об’єктно-орієнтованої програми для імітації роботи автоматичної системи контролю стану систем автомобіля.

10. Реалізація об’єктно-орієнтованої програми для імітації роботи банкомата.

11. Реалізація об’єктно-орієнтованої програми для імітації роботи автомата по продажу цукерок.

12. Реалізація об’єктно-орієнтованої програми для імітації роботи телевізора.

13. Реалізація об’єктно-орієнтованої програми для імітації роботи системи опрацювання текстів.

14. Реалізація об’єктно-орієнтованої програми для імітації роботи мікрохвильової печі.

15. Реалізація об’єктно-орієнтованої програми для імітації роботи турнікета метрополітену.

16. Реалізація об’єктно-орієнтованої програми для імітації процесу вирощування рослини.

17. Реалізація об’єктно-орієнтованої програми для імітації роботи системи зрошування газону.

18. Реалізація об’єктно-орієнтованої програми для імітації роботи системи кухонного комбайну.

19. Реалізація об’єктно-орієнтованої програми для імітації роботи системи багатофункціонального дистанційного пульта управління (tv, cd, vcd, sat,…).

20. Реалізація об’єктно-орієнтованої програми для імітації роботи тестера (амперметр, омметр, вольтметр).

21. Реалізація об’єктно-орієнтованої програми для імітації роботи інженерного калькулятора (dec, hex, bin, …) .

22. Реалізація об’єктно-орієнтованої програми для імітації роботи системи відеоспостереження.

23. Реалізація об’єктно-орієнтованої програми для імітації роботи системи опалення приміщень (газ. котел, насос, …).

24. Реалізація об’єктно-орієнтованої програми для опису функціонування факультету ВУЗу.

## 7. Самостійна робота

1 Об’єктний підхід в об’єктно-орієнтованому проектуванні.
2 Класифікація в об’єктно-орієнтованому аналізі.
3 Об’єктно-орієнтоване проектування. Система позначень.
4 Діаграми класів.
5 Діаграми переходів станів.
6 Діаграми об’єктів.
7 Часові діаграми.
8 Модульні діаграми.

9 Діаграми процесів.
10 Ідентифікація зв’язків між класами та об’єктами.
11 Об’єктно-орієнтоване програмування в життєвому циклі розробки програми.
12 Переваги та недоліки об’єктно-орієнтованого програмування.
13 Відношення між об’єктами.
14 Відношення між класами.
15 Взаємозв’язки класів та об’єктів.
16 Перевантаження операторів.
17 Наслідування.
18 Віртуальні функції.
19 Абстрактні класи.
20 Контроль доступу до членів класу.
21 Шаблони класів.
22 Параметризовані функції.
23 Обробка виняткових ситуацій.
24 Множинне наслідування. Керування доступом.
25 Інформація про тип на етапі виконання.
26 Стандартні контейнери.
27 Визначення нового контейнера.
28 Алгоритми стандартної бібліотеки.
29 Ітератори і розподіл пам’яті.
30 Потоки вводу/виводу.
31 Стандартні бібліотеки чисельних методів.
32 Процесс розробки програмного забезпечення, його мета, головні етапи.
33 Цілі програмування та розробки програмного забезпечення.
34 Використання змінних.
35 Загальновживані оператори.
36 Умовні конструкції.
37 Цикли. Типи циклів.
38 Процедури.
39 Обробка помилок.
40 Скалярні стандартні типи даних в мові C#.
41 Скалярні типи користувача в мові C#.
42 Основні операції в мові C#.
43 Загальна структура програми мовою C#.
44 Умовні оператори мови C#.
45 Оператори циклу у мові C#.
46 Введення і виведення інформації у мові C#.
47 Методи в мові C#.
48 Рекурсія. Пряма та непряма рекурсія.
49 Типи параметрів у мові C#.
50 Масиви у мові C#.
51 Основні принципи об'єктно-орієнтованого підходу в програмуванні.
52 Поняття інкапсуляції. Організація взаємодії об'єктів.
53 Типи-значення та типи-посилання.
54 Способи створення об’єктів.

55 Способи знищення об’єктів. Збирач сміття.
56 Базові елементи програми. Як вони відображаються графічно.
57 Принцип компіляції у .NET.
58 Структура простої програми мовою C#.
59 Основні скалярні типи даних мови C#.
60 Правила іменування змінних.
61 Основні операції мови C#..
62 Оператори інкременту та декременту..
63 Явне і неявне перетворення типів.
64 Принцип обробки виключень в мові C#.
65 Клас в мові C#.
66 Методи класу. Звертання до методу класу.
67 Конструктори класу.
68 Деструктор класу.
69 Утилізація об’єктів, що не використовуються програмою.
70 Масиви в мові C.
71 Цикл foreach.
72 Модифікатори параметрів методів класу.
73 Структури в мові C#.
74 Колекції в мові C#.
75 Перезавантажені методи класу.
76 Зв’язок між типом та екземпляром масива.
77 Основні характеристики об’єкта: ідентичність, стан і поведінку.
78 Різниця між типами-значеннями та типами-посиланнями.
79 Основні простори імен у .NET.
80 Особливості конструктора за замовчуванням.
81 Роль збирача сміття.

## 8. Тренінг з дисципліни (8 год.)

Тематика: реалізація консольного програмного додатку.

Порядок проведення:

- здійснити об’єктно-орієнтований аналіз предметної області;
- реалізувати об’єктна-орієнтовану модель на основі методів об’єктно-орієнтоване проектування;
- реалізувати додаток на мові об’єктно-орієнтованого програмування;
- провести тестування програми;
- представити результати роботи у вигляді короткого звіту: діаграми варіантів використання та класів, реалізація, результати тестування.

Література: 2, 10, 12, 17.

## 9. Методи оцінювання.

В процесі вивчення дисципліни “Об’єктно-орієнтоване програмування” використовуються наступні методи оцінювання навчальної роботи студента:

- поточне опитування;
- стандартизовані тести;

- залікове модульне тестування та опитування;
- презентації результатів виконання завдань;
- оцінювання виконання КПІЗ;
- ректорська контрольна робота;
- тренінги;
- залік (семестр – 2);
- екзамен (семестр – 3).

## 10. Критерії, форми поточного та підсумкового контролю

Підсумковий бал (за 100-бальною шкалою) з дисципліни "Об'єктно-орієнтоване програмування" визначається як середньозважена величина, в залежності від питомої ваги кожної складової залікового кредиту:

**Семестр – 2.**

| Заліковий модуль 1 | Заліковий модуль 2 (ректорська контрольна робота) | Заліковий модуль 3 (підсумкова оцінка за КПІЗ, враховуючи поточне опитування) | Разом |
|---|---|---|---|
| 30 % | 40 % | 30 % | 100 % |
| 1. Усне або письмове опитування під час заняття – 40 балів.<br>2. Лабораторні роботи – 60 балів | 1.Тестові завдання (30 тестів по 2 бали за тест) – 60 балів<br>2. Завдання 1 – 20 балів<br>3. Завдання 2 – 20 балів | 1. Виконання та захист КПІЗ – 80 балів<br>2. Виконання завдань під час тренінгу – 20 балів | 100 |

**Семестр – 3.**

| Заліковий модуль 1 | Заліковий модуль 2 (ректорська контрольна робота) | Заліковий модуль 3 (підсумкова оцінка за КПІЗ, | Заліковий модуль 4 (письмовий екзамен) | Разом |
|---|---|---|---|---|
| 20% | 20 % | 20 % | 40% | 100 % |
| 1. Усне або письмове опитування під час заняття – 40 балів.<br>2. Лабораторні роботи – 60 балів | 1.Тестові завдання (30 тестів по 2 бали за тест) – 60 балів<br>2. Завдання 1 – 20 балів<br>3. Завдання 2 – 20 балів | 1. Виконання та захист КПІЗ – 80 балів<br>2. Виконання завдань під час тренінгу – 20 балів | 1.Тестові завдання (30 тестів по 2 бали за тест) – 60 балів<br>2. Завдання 1 – 20 балів<br>3. Завдання 2 – 20 балів | 100 |

Шкала оцінювання:

| За шкалою Університету | За національною шкалою | За шкалою ECTS |
|---|---|---|
| 90-100 | відмінно | A (відмінно) |
| 85-89 | добре | B (дуже добре) |

| | | |
|---|---|---|
| 75-84 | | C (добре) |
| 65-74 | задовільно | D (задовільно) |
| 60-64 | | E (достатньо) |
| 35-59 | незадовільно | FX (незадовільно з можливістю повторного складання) |
| 1-34 | | F (незадовільно з обов'язковим повторним курсом) |

## 11. Інструменти, обладнання та програмне забезпечення, використання яких передбачає навчальна дисципліна

| № | Найменування | Номер теми |
|---|---|---|
| 1. | Мультимедійний проектор | 1-9 |
| 2. | Проекційний екран | 1-9 |
| 3. | Комунікаційне програмне забезпечення (Internet Explorer, Google Chrome, Firefox) | 1-9 |
| 4. | Операційна система Windows, наявність доступу до мережі Internet | 1-9 |
| 5. | Персональні комп'ютери | 1-9 |
| 6. | Комунікаційне програмне забезпечення (Zoom) для проведення занять у режимі онлайн (за необхідності) | 1-9 |
| 7. | Комунікаційна навчальна платформа (Moodle) для організації дистанційного навчання (за необхідності) | 1-9 |
| 8. | Базове програмне забезпечення Microsoft Office | 1-9 |
| 9. | Спеціалізоване програмне забезпечення:<br>- Середовище розробки Microsoft Visual Studio | 3-9 |

## Рекомендовані джерела інформації


1. Stroustrup Bjarne. A Tour of C++, Second Edition. – July 20, 2018
2. C/C++ language and standard libraries reference: [Електронний ресурс]. – Режим доступу: https://msdn.microsoft.com/en-us/library/hh875057.aspx.
3. C++ Crash Course: A Fast-Paced Introduction. / Lospinoso Josh. ISBN 1593278885. – 2019. – 792c.
4. Stephen Prata. C++ Primer Plus. – Pearson India; 6th edition. – 2015.
5. http://cppstudio.com – Сайт з ресурсами по вивченню C++
6. http://www.cplusplus.com/ – Сайт з найновішою довідкою по C++
7. https://msdn.microsoft.com/ – керівництво по продуктах Microsoft
8. International Standard ISO/IEC 14882:2014(E) – Programming Language C++, ISBN13: 978-0321563842: [Електронний ресурс]. – Режим доступу: https://isocpp.org/std/thestandard.
9. Геннадій Галисєєв. Системне програмування: Видавництво Університет «Україна», 2019.

10. Scott Meyers. Effective Modern C++: 42 Specific Ways to Improve Your Use of C++11 and C++14. – 1st Edition. O′Reilly LCF Publishing. – 2014. – 334 с.
11. Пекарський Б.Г. Основи програмування: Навчальний посібник. – Кондор, – 2018. – 364 с.
12. Bjarne Stroustrup. The C++ Programming Language. – Addison-Wesley Professional; 4th edition. – 2013. – 1376 с.
13. Jospeh Albahari. C# 8.0 Pocket Reference. – 1st edition, O′Reilly LCF Publishing. C#. – 1st edition, Learn Coding Fast
14. Mark J. Price C# 8.0 and .NET Core 3.0. – 4th edition. Packt Publishing
15. Andrew Stellman. Head First C#. – 4th edition, O'Reilly Media, Inc. – 2020.
16. Mikael Olsson. C# 10 Quick Syntax Reference. – 4th Ed. Apress. – 2023
17. Joe Mayo. C# Cookbook: Modern Recipes for Professional Developers. 1st Ed.
18. C# OOP (Object-Oriented Programming) - W3Schools. [Електронний ресурс]. – Режим доступу: https://www.w3schools.com/cs/cs_oop.php
19. Learn C#. Free courses, tutorials, videos, and more about learning the programming language C#. [Електронний ресурс]. – Режим доступу: https://dotnet.microsoft.com/en-us/learn/csharp
20. Object-Oriented programming (C#). [Електронний ресурс]. – Режим доступу: https://learn.microsoft.com/en-us/dotnet/csharp/fundamentals/tutorials/oop.