

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ МОРСЬКИЙ УНІВЕРСИТЕТ
КАФЕДРА КІБЕРБЕЗПЕКИ ТА ЗАХИСТУ ІНФОРМАЦІЇ**

**МЕТОДИЧНІ ВКАЗІВКИ
до виконання практичних робіт
з освітнього компонента (навчальної дисципліни)
Об'єктно-орієнтоване програмування
для здобувачів першого (бакалаврського) рівня вищої освіти
за спеціальністю
125 Кібербезпека та захист інформації
(частина 1)**

Одеса – 2025

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ МОРСЬКИЙ УНІВЕРСИТЕТ
КАФЕДРА КІБЕРБЕЗПЕКИ ТА ЗАХИСТУ ІНФОРМАЦІЇ**

**МЕТОДИЧНІ ВКАЗІВКИ
до виконання практичних робіт
з освітнього компонента (навчальної дисципліни)
Об'єктно-орієнтоване програмування
для здобувачів першого (бакалаврського) рівня вищої освіти
за спеціальністю
125 Кібербезпека та захист інформації
(частина 1)**

**Затверджено
на засіданні кафедри кібербезпеки та
захисту інформації
Протокол № 1 від 01.09.2025р.**

Одеса – 2025

Методичні вказівки до виконання практичних робіт з освітнього компонента (навчальної дисципліни) Об'єктно-орієнтоване програмування для здобувачів першого (бакалаврського) рівня вищої освіти за спеціальністю 125 Кібербезпека та захист інформації (частина 1) / Укл.: К.О. Трифонова. – Одеса, 2025. – 15 с.

Укладач: Трифонова К.О., ст. викл.

ЗМІСТ

Вступ.....	5
Практична робота №1. Організація класу.....	6
Практична робота №2. Інкапсуляція класу.....	6
Практична робота №3. Перевантаження методів та конструкторів.....	8
Практична робота №4. Перевантаження основних операторів	9
Практична робота №5. Перевантаження спеціалізованих операторів.....	10
Практична робота №6. Індексатори і властивості.....	11
Практична робота №7. Базові та похідні класи.....	12
Практична робота №8. Абстрактні класи	13

ВСТУП

Навчальна дисципліна Об'єктно-орієнтоване програмування є необхідною дисципліною для підвищення рівня теоретичних і прикладних знань, що формують фахівця в галузі інформаційних технологій, які сприяють утворенню у здобувачів поглиблених вмінь та навичок розробки програмного забезпечення із застосуванням сучасних парадигм програмування, тобто підготовці спеціалістів для створення, впровадження та підтримки професійно-орієнтованих комп'ютерних технологій у професійній діяльності.

Дисципліна Об'єктно-орієнтоване програмування відповідає освітньо-професійній програмі, навчальному плану підготовки фахівців першого (бакалаврського) рівня вищої освіти спеціальності 125 Кібербезпека та захист інформації і є складовою циклу дисциплін професійної підготовки обов'язкової частини навчального плану.

Дисципліну викладають впродовж третього семестру першого (бакалаврського) рівня. У процесі навчання передбачено лекції, практичні заняття.

Згідно навчального плану передбачено підсумковий контроль у вигляді екзамену.

Робочу програму навчальної дисципліни укладено згідно з вимогами кредитно-модульної системи організації навчального процесу. Програма визначає обсяги компетентностей, які повинен опанувати здобувач відповідно до освітньо-професійної програми, алгоритму вивчення навчального матеріалу дисципліни Об'єктно-орієнтоване програмування, необхідне методичне забезпечення, складові та технологію оцінювання навчальних досягнень.

Предмет навчальної дисципліни Об'єктно-орієнтоване програмування – сучасні технології розробки програмного забезпечення.

Мета вивчення навчальної дисципліни Об'єктно-орієнтоване програмування – забезпечити формування поглибленої системи теоретичних і практичних знань у галузі розробки програмного забезпечення для реалізації здатності використання інформаційно-комунікаційних технологій, сучасних методів і моделей інформаційної безпеки та/або кібербезпеки.

Завдання вивчення дисципліни:

- формувати теоретичні основи та практичні навички застосування методів вилучення, аналізу, обробки та синтезу інформації для формування технічного завдання при розв'язанні задач в галузі інформаційних технологій;

- формувати теоретичні основи та практичні навички сучасних парадигм програмування та способів їх вибору з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі інформаційних технологій;

- поглибити практичні навички кодування програмного забезпечення для розв'язання задач в галузі інформаційних технологій;

- поглибити практичні навички налагодження програмного забезпечення для розв'язання задач в галузі інформаційних технологій.

Стратегічні цілі дисципліни – націлити майбутніх фахівців на застосування отриманих знань у подальшій професійній підготовці та їх наступній практичній діяльності.

ПРАКТИЧНА РОБОТА №1. ОРГАНІЗАЦІЯ КЛАСУ

Мета роботи

1. Закріплення і поглиблення теоретичних знань, отриманих при вивченні розділу «Організація класу».
2. Набуття практичних навичок розробки та реалізації застосування з використанням організації класу.
3. Набуття практичних навичок налагодження застосування, в якому реалізована організація класу.

Теоретичні відомості

Для виконання лабораторної роботи необхідні теоретичні відомості наводяться в конспекті лекцій у розділі «Організація класу».

Хід роботи

1. Поясніть відмінність понять клас та об'єкт для програми на C#.
2. Перерахуйте складові класу для програми на C#.
3. Наведіть приклад коду на C# оголошення класу, який містить члени даних.
4. Назвіть кількість екземплярів класу, яку можна створювати для програми на C#.
5. Наведіть приклад коду на C# двох екземплярів класу, які посилаються на один і той же об'єкт.
6. Наведіть приклад коду на C# оголошення класу, який містить метод, який не містить вхідних параметрів і нічого не повертає, доступний тільки в межах класу.
7. Наведіть приклад коду на C# оголошення класу, який містить метод, який не містить вхідних параметрів і нічого не повертає, доступний за межами класу.
8. Назвіть оператор, за допомогою якого здійснюється повернення з методу класу типу void для програми на C#.
9. Наведіть приклад коду на C# оголошення класу, який містить метод, що повертає кілька цілих значень.
10. Поясніть відмінність понять параметри та аргументи для методу класу для програми на C#.
11. Наведіть приклад коду на C# оголошення класу, який містить метод із вхідними параметрами.
12. Поясніть наслідки наявності недоступного коду в методі класу для програми на C#.
13. Наведіть приклад коду на C# оголошення класу, який містить конструктор за замовчуванням.
14. Наведіть приклад коду на C# оголошення класу, який містить параметризований конструктор.
15. Поясніть можливість існування класу, для якого не оголошено конструктор для програми на C#.
16. Поясніть механізм ініціалізації змінної типу даних посилань, яка створюється за допомогою оператора new для програми на C#.
17. Поясніть механізм ініціалізації змінної типу даних значення, яка створюється за допомогою оператора new для програми на C#.
18. Поясніть розклад функціонування "збір сміття" для програми на C#.
19. Наведіть приклад коду на C# оголошення класу, який містить деструктор.
20. Наведіть приклад коду на C# оголошення класу, який використовує ключове слово this.

ПРАКТИЧНА РОБОТА №2. ІНКАПСУЛЯЦІЯ КЛАСУ

Мета роботи

1. Закріплення і поглиблення теоретичних знань, отриманих при вивченні розділу «Інкапсуляція класу».
2. Набуття практичних навичок розробки та реалізації застосування з використанням інкапсуляції класу.
3. Набуття практичних навичок налагодження застосування, в якому реалізована інкапсуляція класу.

Теоретичні відомості

Для виконання лабораторної роботи необхідні теоретичні відомості наводяться в конспекті лекцій у розділі «Інкапсуляція класу».

Хід роботи

1. Наведіть способи реалізації інкапсуляції класом для програми на C#.
2. Наведіть приклад задачі, для розв'язку якої доцільно застосування класу з організацією обмеження доступу до членів даних класу для програми на C#.
3. Поясніть наслідки звернення з метода одного класу до закритого члену даних іншого класу для програми на C#.
4. Запропонуйте спосіб організації доступу з метода одного класу до закритого члену даних іншого класу для програми на C#.
5. Запропонуйте організацію доступу для члена даних класу, який використовується лише у класі для програми на C#.
6. Запропонуйте організацію доступу для метода класу, який встановлює значення для закритого члена даних класу для програми на C#.
7. Запропонуйте організацію доступу для конструктора класу, який ініціалізує закриті члени даних класу для програми на C#.
8. Наведіть приклад коду на C# оголошення методу класу, вхідний параметр якого відноситься до типу даних даного класу.
9. Поясніть зміну значення змінної типу даних значення, після виконання деякого методу, який використовує дану змінну в якості аргументу для програми на C#.
10. Поясніть зміну значення змінної типу даних посилання, після виконання деякого методу, який використовує дану змінну в якості аргументу для програми на C#.
11. Наведіть приклад задачі, для розв'язку якої доцільно застосування організації передачі аргументу методу типу даних значення за посиланням для програми на C#.
12. Наведіть приклад коду на C# оголошення методу класу, вхідний параметр якого відноситься до типу даних значення, що передається за посиланням, ініціалізація якого здійснюється до виклику методу.
13. Наведіть приклад коду на C# оголошення методу класу, вхідний параметр якого відноситься до типу даних значення, що передається за посиланням, ініціалізація якого здійснюється після виклику методу.
14. Наведіть приклад задачі, для розв'язку якої доцільно застосування організації передачі аргументу методу типу даних посилання за посиланням для програми на C#.
15. Поясніть наслідки передачі нульової кількості аргументів методу, вхідний параметр якого приймає змінну кількість аргументів для програми на C#.
16. Поясніть наслідки передачі різнотипних аргументів методу, вхідний параметр якого приймає змінну кількість аргументів для програми на C#.
17. Поясніть наслідки передачі масиву аргументів методу, вхідний параметр якого приймає змінну кількість аргументів для програми на C#.

18. Наведіть приклад коду на C# оголошення методу класу, який повертає об'єкт, що відноситься до типу даних даного класу.

19. Наведіть приклад коду на C# оголошення класу, який містить фабрику класу.

20. Наведіть приклад коду на C# оголошення методу класу, який повертає масив типу даних даного класу.

ПРАКТИЧНА РОБОТА №3. ПЕРЕВАНТАЖЕННЯ МЕТОДІВ ТА КОНСТРУКТОРІВ

Мета роботи

1. Закріплення і поглиблення теоретичних знань, отриманих при вивченні розділу «Перевантаження методів та конструкторів».

2. Набуття практичних навичок розробки та реалізації застосування з використанням перевантаження методів та конструкторів.

3. Набуття практичних навичок налагодження застосування, в якому реалізовано перевантаження методів та конструкторів.

Теоретичні відомості

Для виконання лабораторної роботи необхідні теоретичні відомості наводяться в конспекті лекцій у розділі «Перевантаження методів та конструкторів».

Хід роботи

1. Наведіть приклад задачі, для розв'язку якої доцільно застосування перевантаження методів класу для програми на C#.

2. Наведіть приклад коду на C# оголошення класу, який містить кілька перевантажених методів.

3. Поясніть наслідки виклику конструктора, тип даних аргументу якого в точності не співпадає, з жодним типом даних вхідних параметрів перевантажених конструкторів класу для програми на C#.

4. Наведіть приклад коду на C# оголошення класу, який містить кілька перевантажених конструкторів.

5. Наведіть приклад коду на C# оголошення класу, який містить конструктор, що викликає інший конструктор цього класу.

6. Поясніть можливість створення об'єкта за допомогою ініціалізатору деякого класу, який не містить жодного конструктора для програми на C#.

7. Поясніть можливість викликати параметризований метод класу без аргументів для програми на C#.

8. Перерахуйте члени класу, які можуть містити необов'язкові параметри для програми на C#.

9. Поясніть можливість оголошення методу класу, який містить обов'язкові та необов'язкові параметри для програми на C#.

10. Наведіть приклад коду на C# методу класу, який містить необов'язкові параметри.

11. Поясніть можливість виклику методу класу із зазначенням аргументів, які не збігаються з порядком параметрів, вказаних при оголошенні даного методу для програми на C#.

12. Назвіть кількість методів Main(), які можуть бути оголошені в класі для програми на C#.

13. Наведіть приклад задачі, для розв'язку якої доцільно застосування методу класу Main(), який повертає деяке значення для програми на C#.

14. Наведіть приклад задачі, для розв'язку якої доцільно застосування методу класу Main(), який має вхідні параметри для програми на C#.

15. Наведіть приклад задачі, для розв'язку якої доцільно застосування рекурсії для програми на C#.

16. Наведіть приклад задачі, для розв'язку якої доцільно застосування класу, який містить статичний член даних для програми на C#.
17. Наведіть приклад коду на C# оголошення класу, який містить статичну фабрику класу.
18. Поясніть необхідність оголошення методу класу Main() статичним для програми на C#.
19. Наведіть приклад коду на C# оголошення класу, який містить статичний конструктор.
20. Наведіть приклад задачі, для розв'язку якої доцільно застосування статичного класу для програми на C#.

ПРАКТИЧНА РОБОТА №4. ПЕРЕВАНТАЖЕННЯ ОСНОВНИХ ОПЕРАТОРІВ

Мета роботи

1. Закріплення і поглиблення теоретичних знань, отриманих при вивченні розділу «Перевантаження основних операторів».
2. Набуття практичних навичок розробки та реалізації застосування з використанням перевантаження основних операторів.
3. Набуття практичних навичок налагодження застосування, в якому реалізовано перевантаження основних операторів.

Теоретичні відомості

Для виконання лабораторної роботи необхідні теоретичні відомості наводяться в конспекті лекцій у розділі «Перевантаження основних операторів».

Хід роботи

1. Поясніть необхідність оголошення перевантаження оператора класу за допомогою public та static для програми на C#.
2. Назвіть типи даних, які можуть бути використані для повернення значення при перевантаженні оператора класу для програми на C#.
3. Назвіть типи даних, які можуть бути використані в якості вхідних параметрів при перевантаженні оператора класу для програми на C#.
4. Поясніть можливість використання модифікаторів параметра ref і out для вхідних параметрів при перевантаженні оператора класу для програми на C#.
5. Наведіть приклад задачі, для розв'язку якої доцільно застосування перевантаження бінарного арифметичного оператора класу для програми на C#.
6. Наведіть приклад коду на C# оголошення класу, який містить перевантаження бінарного арифметичного оператора.
7. Поясніть можливі типи даних для вхідних та вихідних параметрів оголошення перевантаження бінарного арифметичного оператора класу, за допомогою якого можна сформулювати математичний вираз для програми на C#.
8. Наведіть приклад задачі, для розв'язку якої доцільно застосування перевантаження унарного арифметичного оператора класу для програми на C#.
9. Наведіть приклад коду на C# оголошення класу, який містить перевантаження унарного арифметичного оператора.
10. Поясніть можливі типи даних для вхідних та вихідних параметрів оголошення перевантаження унарного арифметичного оператора класу, за допомогою якого можна сформулювати математичний вираз для програми на C#.
11. Наведіть приклад коду на C# оголошення класу, який містить перевантаження бінарного арифметичного оператора для виконання операції із вбудованим типом даних.
12. Перерахуйте кількість можливих оголошень перевантаження бінарного оператора класу для виконання операції з деяким вбудованим типом даних для програми на C#.

13. Перерахуйте кількість можливих оголошень перевантаження унарного оператора класу для виконання операції з деяким вбудованим типом даних для програми на C#.

14. Наведіть приклад задачі, для розв'язку якої доцільно застосування перевантаження оператора відношення класу для програми на C#.

15. Наведіть приклад коду на C# оголошення класу, який містить перевантаження оператора відношення.

16. Наведіть приклад коду на C# оголошення класу, який містить перевантаження оператора відношення, з наступною можливістю використання його в операторі if.

17. Наведіть приклад коду на C# оголошення класу, який містить перевантаження оператора відношення, з наступною можливістю побудови за допомогою них математичних виразів.

18. Наведіть приклад задачі, для розв'язку якої доцільно застосування перевантаження оператора true/false класу для програми на C#.

19. Наведіть приклад коду на C# оголошення класу, який містить перевантаження оператора true/false, з наступною можливістю використання його в операторі if.

20. Наведіть приклад коду на C# оголошення класу, який містить перевантаження оператора true/false, з наступною можливістю використання його в операторі do-while.

ПРАКТИЧНА РОБОТА №5. ПЕРЕВАНТАЖЕННЯ СПЕЦІАЛІЗОВАНИХ ОПЕРАТОРІВ

Мета роботи

1. Закріплення і поглиблення теоретичних знань, отриманих при вивченні розділу «Перевантаження спеціалізованих операторів».
2. Набуття практичних навичок розробки та реалізації застосування з використанням перевантаження спеціалізованих операторів.
3. Набуття практичних навичок налагодження застосування, в якому реалізовано перевантаження спеціалізованих операторів.

Теоретичні відомості

Для виконання лабораторної роботи необхідні теоретичні відомості наводяться в конспекті лекцій у розділі «Перевантаження спеціалізованих операторів».

Хід роботи

1. Наведіть приклад задачі, для розв'язку якої доцільно застосування перевантаження логічного оператора класу для програми на C#.
2. Наведіть приклад коду на C# оголошення класу, який містить перевантаження логічного оператора.
3. Назвіть типи даних, які можуть бути використані для повернення значення при перевантаженні логічного оператора класу для програми на C#.
4. Назвіть типи даних, які можуть бути використані в якості вхідних параметрів при перевантаженні логічного оператора класу для програми на C#.
5. Наведіть приклад коду на C# оголошення класу, який містить перевантаження логічного оператора, з наступною можливістю використання його в операторі if.
6. Наведіть приклад коду на C# оголошення класу, який містить перевантаження логічного оператора, з наступною можливістю побудови за допомогою них математичних виразів.
7. Наведіть приклад задачі, для розв'язку якої доцільно застосування укороченого логічного оператора класу для програми на C#.
8. Поясніть причину неможливості перевантаження укороченого логічного оператора класу для програми на C#.

9. Поясніть відмінність функціонування логічного оператора від укороченого логічного оператора класу для програми на C#.

10. Наведіть приклад коду на C# оголошення класу, який містить перевантаження логічного оператора, з наступною можливістю застосування укороченого логічного оператора.

11. Наведіть приклад задачі, для розв'язку якої доцільно застосування перевантаження оператора перетворення класу для програми на C#.

12. Наведіть приклад коду на C# оголошення класу, який містить перевантаження оператора явного перетворення.

13. Наведіть приклад коду на C# оголошення класу, який містить перевантаження оператора неявного перетворення.

14. Поясніть наслідки оголошення явного та неявного оператора перетворення класу з одного і того ж вбудованого типу даних у тип даних класу для програми на C#.

15. Поясніть наслідки оголошення явного та неявного оператора перетворення класу з різних вбудованих типів даних у тип даних класу для програми на C#.

16. Наведіть приклади операторів, які можуть бути перевантажені як унарні, так і бінарні оператори класу для програми на C#.

17. Наведіть приклади операторів, які не можуть бути перевантажені в класі для програми на C#.

18. Поясніть причину неможливості перевантаження складеного оператора класу для програми на C#.

19. Поясніть причину неможливості перевантаження оператора приведення класу для програми на C#.

20. Поясніть причину неможливості перевантаження оператора індексації класу для програми на C#.

ПРАКТИЧНА РОБОТА №6. ІНДЕКСАТОРИ І ВЛАСТИВОСТІ

Мета роботи

1. Закріплення і поглиблення теоретичних знань, отриманих при вивченні розділу «Індексатори та властивості».

2. Набуття практичних навичок розробки та реалізації застосування з використанням індексаторів та властивостей.

3. Набуття практичних навичок налагодження застосування, в якому реалізовано індексатори та властивості.

Теоретичні відомості

Для виконання лабораторної роботи необхідні теоретичні відомості наводяться в конспекті лекцій у розділі «Індексатори та властивості».

Хід роботи

1. Поясніть можливість оголошення перевантаження оператора квадратні дужки класу для програми на C#.

2. Наведіть приклад задачі, для розв'язку якої доцільно застосування індексатора класу для програми на C#.

3. Наведіть приклад коду на C# оголошення класу, який містить одновимірний індексатор.

4. Наведіть приклад коду на C# оголошення класу, який містить одновимірний індексатор лише для читання.

5. Наведіть приклад коду на C# оголошення класу, який містить одновимірний індексатор тільки для запису.

6. Поясніть наслідки використання індексатора в правій частині оператора присвоєння для об'єкту, у якого в класі оголошено індексатор тільки для запису для програми на C#.

7. Назвіть типи даних, які можуть бути використані для повернення значення при перевантаженні індексатора класу для програми на C#.

8. Назвіть типи даних, які можуть бути використані в якості вхідних параметрів при перевантаженні індексатора класу для програми на C#.

9. Наведіть приклад коду на C# оголошення класу, який містить два одновимірні перевантажені індексатори.

10. Поясніть наслідки виклику індексатора тип даних аргументу, якого точно не співпадає, ні з одним типом даних вхідних параметрів перевантажених індексаторів класу для програми на C#.

11. Наведіть приклад коду на C# оголошення класу, який містить член даних, що не є масивом, та одновимірний індексатор.

12. Наведіть приклад коду на C# оголошення класу, який містить багатовимірний індексатор.

13. Наведіть приклад задачі, для розв'язку якої доцільно застосування властивості класу для програми на C#.

14. Поясніть відмінність властивості від члена даних класу для програми на C#.

15. Наведіть приклад коду на C# оголошення класу, який містить властивість.

16. Поясніть відмінність властивості від автоматично реалізованої властивості класу для програми на C#.

17. Наведіть приклад коду на C# оголошення класу, який містить автоматично реалізовану властивість і створення об'єкта за допомогою ініціалізатора.

18. Поясніть неможливість організації передачі властивості в якості вхідного параметра методу через ref і out для програми на C#.

19. Наведіть приклад коду на C# оголошення класу, який містить властивість доступну для читання та запису тільки в класі.

20. Наведіть приклад коду на C# оголошення класу, який містить автоматично реалізовану властивість доступну тільки для читання.

ПРАКТИЧНА РОБОТА №7. БАЗОВІ ТА ПОХІДНІ КЛАСИ

Мета роботи

1. Закріплення і поглиблення теоретичних знань, отриманих при вивченні розділу «Базові та похідні класи».
2. Набуття практичних навичок розробки та реалізації застосування з використанням базових та похідних класів.
3. Набуття практичних навичок налагодження застосування, в якому реалізовано базові та похідні класи.

Теоретичні відомості

Для виконання лабораторної роботи необхідні теоретичні відомості наводяться в конспекті лекцій у розділі «Базові та похідні класи».

Хід роботи

1. Перерахуйте основні принципи об'єктно-орієнтованого програмування для програми на C#.
2. Наведіть приклад задачі, для розв'язку якої доцільно застосування успадкування класів для програми на C#.
3. Назвіть кількість класів, які можуть бути базовими для одного похідного класу для програми на C#.

4. Назвіть кількість класів, які можуть бути похідними для одного базового класу для програми на C#.
5. Наведіть приклад коду на C# оголошення базового та похідного класу.
6. Поясніть наслідки звернення до закритих членів базового класу у похідному класі для програми на C#.
7. Запропонуйте спосіб, за допомогою якого можна отримати доступ до закритих членів базового класу у похідному класі для програми на C#.
8. Поясніть наслідки звернення до захищених членів базового класу у похідному класі для програми на C#.
9. Поясніть наслідки звернення до захищених членів класу у незалежному класі для програми на C#.
10. Назвіть кількість екземплярів базового класу, які можна створити для програми на C#.
11. Назвіть кількість екземплярів похідного класу, які можна створити для програми на C#.
12. Наведіть приклад коду на C# оголошення базового класу без конструкторів та похідного класу, який містить конструктор, що викликає конструктор базового класу за замовчуванням.
13. Наведіть приклад коду на C# оголошення базового класу, який містить параметризований конструктор, і похідного класу, який містить конструктор, що викликає конструктор базового класу.
14. Наведіть приклад задачі, для розв'язку якої доцільно застосування приховування імен при спадкуванні класів для програми на C#.
15. Наведіть приклад коду на C# оголошення базового класу та похідного класу, який застосовує приховування методу базового класу.
16. Поясніть наслідки, які виникають через відсутність ключового слова `new` при застосуванні приховування імен при спадкуванні класів для програми на C#.
17. Наведіть приклад коду на C# оголошення базового класу та похідного класу, який викликає метод базового класу у своєму методі.
18. Наведіть приклад коду на C# оголошення базового класу та похідного класу, який викликає прихований метод базового класу у своєму методі.
19. Наведіть приклад задачі, для розв'язку якої доцільно застосування багаторівневої ієрархії класів для програми на C#.
20. Наведіть приклад коду на C# оголошення трьох класів, які утворюють ієрархію.

ПРАКТИЧНА РОБОТА №8. АБСТРАКТНІ КЛАСИ

Мета роботи

1. Закріплення і поглиблення теоретичних знань, отриманих при вивченні розділу «Абстрактні класи».
2. Набуття практичних навичок розробки та реалізації застосування з використанням абстрактних класів.
3. Набуття практичних навичок налагодження застосування, в якому реалізовано абстрактні класи.

Теоретичні відомості

Для виконання лабораторної роботи необхідні теоретичні відомості наводяться в конспекті лекцій у розділі «Абстрактні класи».

Хід роботи

1. Поясніть можливість створення екземпляра похідного класу, у випадку якщо ні в базовому ні в похідному класах не оголошені конструктори для програми на C#.

2. Наведіть приклад коду на C# створення екземпляра похідного класу, в результаті якого доводиться, що конструктори викликаються по порядку виведення їх класів.
3. Поясніть наслідки, у результаті присвоєння змінній посилання на об'єкт базового класу посилання на об'єкт похідного класу для програми на C#.
4. Поясніть наслідки, у результаті присвоєння змінній посилання на об'єкт похідного класу посилання на об'єкт базового класу для програми на C#.
5. Наведіть приклад задачі, для розв'язку якої доцільно застосування віртуального і перевизначеного методу ієрархії класів для програми на C#.
6. Наведіть приклад коду на C# оголошення ієрархії двох класів, базовий клас якої містить віртуальний метод, і похідний клас містить перевизначений метод.
7. Наведіть приклад коду на C# оголошення ієрархії двох класів, базовий клас якої містить віртуальний індексатор, і похідний клас містить перевизначений індексатор.
8. Поясніть результат виклику віртуального методу для змінної посилання на базовий клас, якій присвоєно посилання на похідний клас, що містить перевизначений метод для програми на C#.
9. Поясніть результат виклику віртуального статичного методу для змінної посилання на базовий клас, якій присвоєно посилання на похідний клас, що містить перевизначений статичний метод для програми на C#.
10. Наведіть приклад коду на C#, який демонструє реалізацію динамічного поліморфізму.
11. Наведіть приклад задачі, для розв'язку якої доцільно застосування абстрактного та перевизначеного методу ієрархії класів для програми на C#.
12. Наведіть приклад коду на C# оголошення ієрархії двох класу, базовий клас якої містить абстрактний метод, і похідний клас, містить перевизначений метод.
13. Поясніть наслідки оголошення абстрактного методу в базовому класі, та відсутність перевизначення нього у похідному класі для програми на C#.
14. Поясніть можливість створення екземпляра абстрактного класу для програми на C#.
15. Поясніть наслідки оголошення абстрактних та не абстрактних членів в абстрактному класі для програми на C#.
16. Наведіть приклад задачі, для розв'язку якої доцільно запобігання успадкування ієрархії класів для програми на C#.
17. Наведіть приклад коду на C# оголошення класу, в якому перевантажено оператор рівності та метод Equals(object).
18. Наведіть приклад коду на C# упаковки та розпаковки змінної типу даних посилання.
19. Наведіть приклад коду на C# оголошення та ініціалізації масиву, який містить змінні типу даних посилання і типу даних значень.
20. Наведіть приклад коду на C# оголошення методу, який приймає вхідний параметр будь-якого типу даних.

Навчальне видання

Методичні вказівки до виконання практичних робіт з освітнього компонента (навчальної дисципліни) Об'єктно-орієнтоване програмування для здобувачів першого (бакалаврського)

рівня вищої освіти за спеціальністю 125 Кібербезпека та захист інформації (частина 1) / Укл.:
К.О. Трифонова. – Одеса, 2025. – 15 с.

Укладач: Трифонова К.О., ст. викл.

Підписано до друку _____. Формат 60x84/16. Папір газетний. Друк офсетний. 0,87
ум. друк. арк. 0,94 обл. - вид. арк.
Тираж 100 пр. Зам. №

Одеський національний морський університет
65029, Одеса, вул. Мечникова, 34