

**Інформаційний пакет освітніх компонент навчального плану
освітньо-професійної програми "Інформаційні системи та технології"
(назва)**

Освітнього рівня бакалавр

Спеціальності 126 " "

Галузь знань 12 Інформаційні технології

1. Назва освітньої компоненти Об'єктно-орієнтоване програмування C#
(назва дисципліни)

2. Тип основна

| 3. Обсяг: | Кредитів ECTS | Годин | За видами занять: | | | | |
|-----------|---------------|-------|-------------------|---------|----------------------|------------------------|--------------------------|
| | | | Лекцій | Семінар | Практичних занять | Лабораторних занять | Самостійна підготовка |
| | 5 | 150 | 18 | | 18 | 18 | 96 |

4. Взаємозв'язок у структурно-логічній схемі

| | |
|--|---|
| Освітні компоненти, які передують вивченню | 1. Вища математика |
| Освітні компоненти для яких є базовою | 1. Моделювання та проектування програмного забезпечення |

5. Компетенції відповідно до ОПІ та вимог роботодавців:

Компетенції відповідно до ОПІ

| | |
|-------|-------|
| Знати | Вміти |
|-------|-------|

| | |
|---|---|
| <p><i>Базові поняття інформатики й сучасних інформаційних систем та технологій, технології безпечної роботи в комп'ютерних мережах, методи створення баз даних та інтернет-ресурсів, технології розроблення алгоритмів і комп'ютерних програм мовами високого рівня із застосуванням об'єктно-орієнтованого програмування для розв'язання задач проектування і використання інформаційних систем та технологій.</i></p> <p><i>Сучасні технології інформаційних систем, практичні навички програмування та використання прикладних і спеціалізованих комп'ютерних систем та середовищ з метою їх запровадження у професійній діяльності.</i></p> | <p>ПП 4. Здатність проектувати, розробляти та використовувати засоби реалізації інформаційних систем, технологій та інфокомунікацій (методичні, інформаційні, алгоритмічні, технічні, програмні та інші).</p> |
| Компетенції відповідно до вимог роботодавців | |
| 1. Поняття класу та членів класу, поняття даних, поняття операторів. | 1. Використовувати основні конструкції об'єктно-орієнтованої мови програмування C# |
| 2. Поняття об'єкту та його членів, порядок розміщення об'єктів у оперативній пам'яті. Поняття видимості членів об'єкту. | 2. Ефективно використовувати оперативну пам'ять. Об'єднувати окремі програмні модулі в одну збірку. |
| 3. Модифікатори доступу. Варіанти використання властивостей і індексаторів. | 3. Приховувати непотрібну інформацію, ефективно обробляти дані. |
| 4. Поняття успадкування, поняття абстрактного класу | 4. Використовувати успадкування для ефективного написання програмного забезпечення. Вносити загальні алгоритми до абстрактного класу. |
| 5. Поняття поліморфізму, інтерфейсу. | 5. Вносити зміни до членів класу використовуючи поліморфізм. Використовувати інтерфейси для стратегічного опису програмного забезпечення. |
| 6. Поняття ієрархії класів, плюси та мінуси створення великих ієрархій, поняття структури, перерахування. | 6. Створювати ієрархію. Підвищувати швидкість обробки даних. Зменшувати ймовірність виникнення помилок. |
| 7. Поняття колекції, Generics, LINQ. | 7. Формувати колекції. Використовувати інформацію різних типів. Робити запити до колекцій. |
| 8. Поняття процесу проектування програмного забезпечення, статичних та динамічних моделей об'єктів. | 8. Проектувати програмне забезпечення. |
| 9. Поняття шаблону проектування. | 9. Здійснювати стандартний підхід до написання коду. |
| 10. Поняття винятку. Основні конструкції з обробки винятків | 10. Зменшувати ймовірність зупинки програмного забезпечення. |
| 11. Поняття потоку. | 11. Здійснювати файлові операції. |
| 12. Поняття делегату, лямбда-виразу. | 12. Передавати методи у якості аргументів. |
| 13. Поняття події. | 13. Здійснювати генерацію та обробку подій. |
| 14. Поняття регулярних виразів. | 14. Здійснювати пошук інформації, використовуючи регулярні вирази. |
| 15. Поняття багатопоточності. | 15. Розробляти ефективне програмне забезпечення шляхом зменшення часу на очікування результату. |

| | |
|--|--|
| 16. Поняття серіалізації, атрибутів. | 16. Зберігати стан об'єктів |
| 17. Поняття перевантаження операторів. | 17. Створювати оператори, які працюють з типами користувача. |
| 18. Поняття збірки сміття. | 18. Видаляти непотрібні об'єкти з оперативної пам'яті. |

6. Результати навчання відповідно до ОПП

1. формування системи понять, знань, умінь і навичок у проектуванні та створенні програмного забезпечення за допомогою об'єктно-орієнтованого програмування

7. План вивчення освітньої компоненти

| Змістовний розділ | Вид заняття | Тема | Знати | Вміти | План заняття | Лекція, методична розробка |
|-------------------|-------------|---|---|---|---------------------------------|---|
| Розділ 1 | | | | | | |
| | Лекція 1 | Тема: Класи та підкласи | 1. Поняття класу та членів класу, поняття даних, поняття операторів. | 1. Використовувати основні конструкції об'єктно-орієнтованої мови програмування C# | посилання на електронний ресурс | http://dl.dut.edu.ua/course/view.php?id=2242 |
| | Лекція 2 | Тема: Внутрішнє представлення об'єктів та таблиця методів | 1. Поняття об'єкту та його членів, порядок розміщення об'єктів у оперативній пам'яті. Поняття видимості членів об'єкту. | 1. Ефективно використовувати оперативну пам'ять. Об'єднувати окремі програмні модулі в одну збірку. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| | Лекція 3 | Тема: Інкапсуляція та приховання інформації | 1. Модифікатори доступу. Варіанти використання властивостей і індексаторів. | 1. Приховувати непотрібну інформацію, ефективно обробляти дані. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| | Лекція 4 | Тема: Успадкування | 1. Поняття успадкування, поняття абстрактного класу | 1. Використовувати успадкування для ефективного написання програмного забезпечення. Вносити загальні алгоритми до абстрактного класу. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| | Лекція 5 | Тема: Поліморфізм. | 1. Поняття поліморфізму, інтерфейсу. | 1. Вносити зміни до членів класу використовуючи поліморфізм. | | http://dl.dut.edu.ua/course/view.php?id= |

| | | | | | |
|-----------|---|---|---|--|---|
| | | | Використовувати інтерфейси для стратегічного опису програмного забезпечення. | | 2242 |
| Лекція 6 | Тема: Ієрархія класів. Структури. Перерахування. | 1. Поняття ієрархії класів, плюси та мінуси створення великих ієрархій, поняття структури, перерахування. | 1. Створювати ієрархію. Підвищувати швидкість обробки даних. Зменшувати ймовірність виникнення помилок. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лекція 7 | Тема: Класи колекцій. | 1. Поняття колекції, Generics, LINQ. | 1. Формувати колекції. Використовувати інформацію різних типів. Робити запити до колекцій. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лекція 8 | Тема: Об'єктно-орієнтоване проектування. | 1. Поняття процесу проектування програмного забезпечення, статичних та динамічних моделей об'єктів. | 1. Проектувати програмне забезпечення. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лекція 9 | Тема: Розподіл поведінки та реалізації. | 1. Поняття шаблону проектування. | 1. Здійснювати стандартний підхід до написання коду. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лекція 10 | Тема: Обробка винятків. Застосування конструкцій checked і unchecked. | 1. Поняття винятку. Основні конструкції з обробки винятків | 1. Зменшувати ймовірність зупинки програмного забезпечення. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лекція 11 | Тема: Взаємодія з файловою системою. | 1. Поняття потоку. | 1. Здійснювати файлові операції. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лекція 12 | Тема: Делегати. Лямбда-вирази. | 1. Поняття делегату, лямбда-виразу. | 1. Передавати методи у якості аргументів. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лекція 13 | Тема: Події. | 1. Поняття події. | 1. Здійснювати генерацію | | http://dl.dut.edu |

| | | | | | |
|---------------------|--|---------------------------------------|---|--|---|
| | | | та обробку подій. | | du.ua/course/view.php?id=2242 |
| Лекція 14 | Тема: Регулярні вирази. | 1. Поняття регулярних виразів. | 1. Здійснювати пошук інформації, використовуючи регулярні вирази. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лекція 15 | Тема: Багатопоточність. | 1. Поняття багатопоточності. | 1. Розробляти ефективне програмне забезпечення шляхом зменшення часу на очікування результату. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лекція 16 | Тема: Серіалізація. | 1. Поняття серіалізації, атрибутів. | 1. Зберігати стан об'єктів | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лекція 17 | Тема: Перевантаження операторів. | 1. Поняття перевантаження операторів. | 1. Створювати оператори, які працюють з типами користувача. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лекція 18 | Тема: Збірка сміття. | 1. Поняття збірки сміття. | 1. Видаляти непотрібні об'єкти з оперативної пам'яті. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Практичне заняття 1 | Тема: Застосування основних конструкцій програмування. | | 1. Використовувати основні конструкції об'єктно-орієнтованої мови програмування C# | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Практичне заняття 2 | Тема: Застосування просторів імен. | | 1. Ефективно використовувати оперативну пам'ять. Об'єднувати окремі програмні модулі в одну збірку. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Практичне заняття 3 | Тема: Застосування властивостей та | | 1. Приховувати непотрібну інформацію, ефективно | | http://dl.dut.edu.ua/course/ |

| | | | | | |
|-----------------------|--|--|---|--|---|
| | індексаторів. | | обробляти дані. | | view.php?id=2242 |
| Практичне заняття 4 | Тема: Застосування успадкування, абстрактних класів. | | 1. Використовувати успадкування для ефективного написання програмного забезпечення. Вносити загальні алгоритми до абстрактного класу. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Практичне заняття 5 | Тема: Застосування інтерфейсів. | | 1. Вносити зміни до членів класу використовуючи поліморфізм. Використовувати інтерфейси для стратегічного опису програмного забезпечення. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Практичне заняття 6 | Тема: Застосування структур, перерахування. | | 1. Створювати ієрархію. Підвищувати швидкість обробки даних. Зменшувати ймовірність виникнення помилок. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Практичне заняття 7 | Тема: Застосування класів колекцій, Generics, LINQ. | | 1. Формувати колекції. Використовувати інформацію різних типів. Робити запити до колекцій. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Практичне заняття 8 | Тема: Проектування ПМ. | | 1. Проектувати програмне забезпечення. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Практичне заняття 9 | Тема: Застосування шаблонів проектування. | | 1. Здійснювати стандартний підхід до написання коду. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лабораторне заняття 1 | Тема: Створення програмного модуля (ПМ) | | 1. Використовувати основні конструкції | | http://dl.dut.edu.ua/course/ |

| | | | | | | |
|-----------------------|--|--|--|---|--|---|
| | | з використанням класів. | | об'єктно-орієнтованої мови програмування C# | | view.php?id=2242 |
| Лабораторне заняття 2 | | Тема: Створення ПМ з використанням просторів імен. | | 1. Ефективно використовувати оперативну пам'ять. Об'єднувати окремі програмні модулі в одну збірку. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лабораторне заняття 3 | | Тема: Створення ПМ з використанням властивостей. | | 1. Приховувати непотрібну інформацію, ефективно обробляти дані. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лабораторне заняття 4 | | Тема: Створення ПМ з використанням успадкування, абстрактних класів. | | 1. Використовувати успадкування для ефективного написання програмного забезпечення. Вносити загальні алгоритми до абстрактного класу. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лабораторне заняття 5 | | Тема: Створення ПМ з використанням інтерфейсів. | | 1. Вносити зміни до членів класу використовуючи поліморфізм. Використовувати інтерфейси для стратегічного опису програмного забезпечення. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лабораторне заняття 6 | | Тема: Створення ПМ з використанням структур, перерахування. | | 1. Створювати ієрархію. Підвищувати швидкість обробки даних. Зменшувати ймовірність виникнення помилок. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лабораторне заняття 7 | | Тема: Створення ПМ з використанням класів колекцій, Generics, LINQ. | | 1. Формувати колекції. Використовувати інформацію різних типів. Робити запити до колекцій. | | http://dl.dut.edu.ua/course/view.php?id=2242 |

| | | | | | |
|------------------------|---|--|--|--|---|
| Лабораторне заняття 8 | Тема: Проектування ПМ. | | 1. Проектувати програмне забезпечення. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лабораторне заняття 9 | Тема: Створення ПМ з використанням шаблонів проектування. | | 1. Здійснювати стандартний підхід до написання коду. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лабораторне заняття 10 | Тема: Створення ПМ з обробкою винятків. | | 1. Зменшувати ймовірність зупинки програмного забезпечення. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лабораторне заняття 11 | Тема: Організація взаємодії з файловою системою. | | 1. Здійснювати файлові операції. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лабораторне заняття 12 | Тема: Створення ПМ з використанням делегатів, лямбда-виразів. | | 1. Передавати методи у якості аргументів. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лабораторне заняття 13 | Тема: Створення ПМ з використанням подій. | | 1. Здійснювати генерацію та обробку подій. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лабораторне заняття 14 | Тема: Створення ПМ з використанням регулярних виразів. | | 1. Здійснювати пошук інформації, використовуючи регулярні вирази. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лабораторне заняття 15 | Тема: Створення ПМ з використанням багатопоточності. | | 1. Розробляти ефективне програмне забезпечення шляхом зменшення часу на очікування результату. | | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лабораторне заняття 16 | Тема: Створення ПМ з використанням серіалізації. | | 1. Зберігати стан об'єктів | | http://dl.dut.edu.ua/course/view.php?id= |

| | | | | | |
|------------------------|--|--|--|---|---|
| | | | | | 2242 |
| Лабораторне заняття 17 | Тема: Створення ПМ із використанням перевантаження операторів. | | | 1. Створювати оператори, які працюють з типами користувача. | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Лабораторне заняття 18 | Тема: Створення ПМ з використанням збірки сміття. | | | 1. Видаляти непотрібні об'єкти з оперативної пам'яті. | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Самостійна робота 1 | Тема: Застосування основних конструкцій програмування. | | | 1. Використовувати основні конструкції об'єктно-орієнтованої мови програмування C# | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Самостійна робота 2 | Тема: Застосування просторів імен. | | | 1. Ефективно використовувати оперативну пам'ять. Об'єднувати окремі програмні модулі в одну збірку. | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Самостійна робота 3 | Тема: Застосування властивостей та індексаторів. | | | 1. Приховувати непотрібну інформацію, ефективно обробляти дані. | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Самостійна робота 4 | Тема: Застосування успадкування, абстрактних класів. | | | 1. Використовувати успадкування для ефективного написання програмного забезпечення. Вносити загальні алгоритми до абстрактного класу. | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Самостійна робота 5 | Тема: Застосування інтерфейсів. | | | 1. Вносити зміни до членів класу використовуючи поліморфізм. Використовувати інтерфейси для | http://dl.dut.edu.ua/course/view.php?id=2242 |

| | | | | |
|----------------------|---|--|---|---|
| | | | стратегічного опису програмного забезпечення. | |
| Самостійна робота 6 | Тема: Застосування структур, перерахування. | | 1. Створювати ієрархію. Підвищувати швидкість обробки даних. Зменшувати ймовірність виникнення помилок. | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Самостійна робота 7 | Тема: Застосування класів колекцій, Generics, LINQ. | | 1. Формувати колекції. Використовувати інформацію різних типів. Робити запити до колекцій. | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Самостійна робота 8 | Тема: Проектування ПМ. | | 1. Проектувати програмне забезпечення. | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Самостійна робота 9 | Тема: Застосування шаблонів проектування. | | 1. Здійснювати стандартний підхід до написання коду. | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Самостійна робота 10 | Тема: Створення ПМ з обробкою винятків. | | 1. Зменшувати ймовірність зупинки програмного забезпечення. | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Самостійна робота 11 | Тема: Організація взаємодії з файловою системою у ПМ. | | 1. Здійснювати файлові операції. | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Самостійна робота 12 | Тема: Створення ПМ з використанням делегатів, лямбда-виразів. | | 1. Передавати методи у якості аргументів. | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Самостійна робота 13 | Тема: Створення ПМ з використанням подій. | | 1. Здійснювати генерацію та обробку подій. | http://dl.dut.edu.ua/course/view.php?id=2242 |

| | | | | |
|----------------------|--|--|--|---|
| Самостійна робота 14 | Тема: Створення ПМ з використанням регулярних виразів. | | 1. Здійснювати пошук інформації, використовуючи регулярні вирази. | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Самостійна робота 15 | Тема: Створення ПМ з використанням багатопоточності. | | 1. Розробляти ефективне програмне забезпечення шляхом зменшення часу на очікування результату. | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Самостійна робота 16 | Тема: Створення ПМ з використанням серіалізації. | | 1. Зберігати стан об'єктів | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Самостійна робота 17 | Тема: Створення ПМ із використанням перевантаження операторів. | | 1. Створювати оператори, які працюють з типами користувача. | http://dl.dut.edu.ua/course/view.php?id=2242 |
| Самостійна робота 18 | Тема: Створення ПМ з використанням збірки сміття. | | 1. Видаляти непотрібні об'єкти з оперативної пам'яті. | http://dl.dut.edu.ua/course/view.php?id=2242 |

8. Мова вивчення освітньої компоненти

(українська, англійська, розділи, що викладаються англійською мовою)

українська

9. Інформаційне забезпечення освітньої компоненти

Рекомендовані джерела та інші навчальні ресурси: вказати підручники, навчальні посібники не пізніше 2010 року видання, які є у нас у бібліотеці на державній мові; електронні ресурси, посилання, електронна бібліотека ДУТ, іншомовні джерела

1. К. Нейгел, Б. Ивѐн, Д. Глинн, К. Уотсон, М. Скиннер. С# 5.0 и платформа .Net 4.5 для профессионалов/ Пер. с агнл. – Компьютерное издательство “Диалектика”, 2014 – 1440 с.
2. Э. Троелсен. Язык программирования С# 5.0 и платформа .Net 4.5 6-е издание / Пер. с агнл. – Издательский дом Питер, 2013. – 1343 с.
3. Д. Албахари, Б. Албахари. С# 5.0 Карманный справочник / Пер. с агнл. – М. Издательский дом “Вильямс”, 2013. – 288 с.
4. Мережа розробників Microsoft “MSDN-the microsoft developer network” – Режим доступу: <https://msdn.microsoft.com>
5. С# 5.0 и платформа .NET 4.5 – Режим доступу: ProfessorWeb.ru

6. Дібрівний О.А., Гребенюк В.В. Вступ до об'єктно орієнтованого програмування С#: Навчальний посібник. – К.: Державний університет телекомунікацій, 2018, - 190сторінок.

10. Методи оцінювання, підсумкові звітності за освітньою компонентою

(заліки, екзамени, курсові проекти, тестування)

Залік, екзамен, курсовий проект

11. Матеріально-технічне забезпечення освітньої компоненти

Visual Studio