

**МЕЛІТОПОЛЬСЬКИЙ ДЕРЖАВНИЙ ПЕДАГОГІЧНИЙ УНІВЕРСИТЕТ  
ІМЕНІ БОГДАНА ХМЕЛЬНИЦЬКОГО**

**ФАКУЛЬТЕТ ІНФОРМАТИКИ, МАТЕМАТИКИ ТА ЕКОНОМІКИ**

**Кафедра інформатики і кібернетики**

<b>Назва освітнього компонента</b> <i>Нормативний/вибірковий</i>	<b>ПП-02 Програмування</b> <i>Нормативний</i>
<b>Ступінь освіти</b>	<b>Бакалавр</b>
<b>Освітня програма</b>	Професійна освіта. Комп'ютерні технології
<b>Рік викладання / Семестр / Курс (рік навчання)</b>	2020 – 2021 / III – IV семестр / II курс
<b>Викладач</b>	Ібрагімова Людмила Анатоліївна Чорна Альона Віталіївна
<b>Профайл викладача</b>	<a href="http://inf.mdpu.org.ua/2016/01/16/ibragimova-ljudmila-anatolivna/">http://inf.mdpu.org.ua/2016/01/16/ibragimova-ljudmila-anatolivna/</a> <a href="http://inf.mdpu.org.ua/2017/10/25/chorna-alona-vitaliivna/">http://inf.mdpu.org.ua/2017/10/25/chorna-alona-vitaliivna/</a>
<b>Контактний тел.</b>	+380 97 231 09 82
<b>E-mail</b>	<a href="mailto:ibragimova@mdpu.org.ua">ibragimova@mdpu.org.ua</a> <a href="mailto:chornaa@mdpu.org.ua">chornaa@mdpu.org.ua</a>
<b>Сторінка курсу в ЦОДТ МДПУ ім. Б.Хмельницького</b>	<a href="https://dfn.mdpu.org.ua/course/view.php?id=4738">https://dfn.mdpu.org.ua/course/view.php?id=4738</a> <a href="https://dfn.mdpu.org.ua/course/view.php?id=888">https://dfn.mdpu.org.ua/course/view.php?id=888</a>
<b>Консультації</b>	Очні консультації: згідно графіку роботи кафедри інформатики і кібернетики. Онлайн-консультації: через систему ЦОДТ МДПУ ім. Б. Хмельницького.

### **1. АНОТАЦІЯ**

Навчальна дисципліна «Програмування» з циклу нормативних освітніх компонентів здобувачів вищої освіти за освітньою програмою «Професійна освіта. Комп'ютерні технології» передбачає формування у здобувачів поглиблених знань, умінь і навичок об'єктно-орієнтованого та візуального програмування мовою С#. Здобувачі отримають базу для подальшого вивчення мови програмування Java, дисциплін «Бази даних та інформаційні системи», «Програмування та підтримка веб-застосувань», виконання професійних завдань під час проходження виробничої практики.

Опанування засобів мови програмування С# розраховано на 2 семестри: перший семестр завершується диференційованим заліком, другий – іспитом.

У другому семестрі передбачено проведення навчальної практики.

## **2. МЕТА ТА ЗАВДАННЯ ОСВІТНЬОГО КОМПОНЕНТА**

**Мета викладання навчальної дисципліни:** формування у здобувачів вищої освіти знань, умінь, навичок і здатностей програмування мовою C#, необхідних для розробки програм різного призначення та успішної професійної діяльності на IT-підприємствах.

**Завдання навчальної дисципліни:** ознайомити студентів з основними конструкції мови програмування C#, можливостями технології Windows Forms, реалізацією головних концепцій об'єктно-орієнтованого програмування у мові C#, патернами проектування.

**Метою навчальної практики** є підготовка студентів до практичної діяльності з розробки та впровадження технологій мехатроніки із застосуванням програмних засобів і комп'ютерних технологій на основі поглибленого вивчення можливостей Lego Mindstorms EV3 та Lego WeDo2.0; оволодіння студентами сучасними методами та засобами діяльності в галузі робототехніки; формування в них на базі одержаних у закладі вищої освіти компетентностей здатності до прийняття самостійних рішень під час роботи в реальних виробничих умовах та виховання необхідності систематично поновлювати свої знання і доцільно їх застосовувати у практичній діяльності.

## **3. ПЕРЕЛІК КОМПЕТЕНТНОСТЕЙ, ЯКІ НАБУВАЮТЬСЯ ПІД ЧАС ОПАНУВАННЯ ОСВІТНІМ КОМПОНЕНТОМ**

### **Загальні компетентності**

**ЗК 3.** Здатність застосовувати знання у практичних ситуаціях.

**ЗК 7.** Здатність працювати в команді.

**ЗК 9.** Здатність оцінювати та забезпечувати якість виконуваних робіт.

### **Спеціальні (фахові) компетентності**

**ФК 18.** Здатність проектувати та розробляти програмне забезпечення із застосуванням різних методів та алгоритмів обчислень, структур даних, парадигм програмування на основі побудови відповідних моделей.

**ФК 19.** Здатність забезпечити організацію обчислювальних процесів в інформаційних системах різного призначення з урахуванням архітектури, конфігурування, показників результативності функціонування операційних систем і системного програмного забезпечення.

**ФК 20.** Здатність застосовувати технології та інструментальні засоби для управління процесами життєвого циклу інформаційних і програмних систем відповідно до вимог замовника.

**ФК 22.** Здатність до розробки програмного забезпечення, що використовує комп'ютерні системи і мережі передачі даних.

## **4. РЕЗУЛЬТАТИ НАВЧАННЯ**

**ПРН 19.** Уміння ефективно використовувати математичний апарат у професійній діяльності для розв'язання задач теоретичного та прикладного характеру у процесі аналізу, синтезу та проектування інформаційних систем за галузями.

**ПРН 20.** Знання базових понять теорії алгоритмів, формальних моделей алгоритмів, питань обчислюваності, розв'язності та нерозв'язності масових проблем, понять складності алгоритмів.

**ПРН 21.** Знання структур даних та фундаментальних алгоритмів, методології та інструментальних засобів об'єктно-орієнтованого аналізу та проектування, особливостей різних парадигм програмування, принципів, моделей, методів і технологій проектування і розроблення програмних продуктів різного призначення.

**ПРН 24.** Знання та вміння використовувати стандарти, методи, технології і засоби управління процесами життєвого циклу інформаційних і програмних систем, продуктів і сервісів інформаційних технологій.

**ПРН 25.** Знання принципів, інструментальних засобів, мов програмування та технологій створення веб-ресурсів, баз даних, розподілених застосувань, інтелектуальних інформаційних систем тощо.

**ПРН 32.** Уміння використовувати сучасну комп'ютерну техніку, пакети прикладних програм і програмувати на рівні, достатньому для розв'язування фізичних та математичних задач, комп'ютерного моделювання явищ і процесів, виконання обчислювальних експериментів.

## **5. ОБСЯГ ОСВІТНЬОГО КОМПОНЕНТА**

<b>Вид заняття</b>	<b>Лекції</b>	<b>Лабораторні заняття</b>	<b>Самостійна робота</b>
<b>Кількість годин III семестр</b>	<b>18 годин</b>	<b>42 години</b>	<b>90 годин</b>
<b>Кількість годин IV семестр</b>	<b>40 годин</b>	<b>52 години</b>	<b>118 годин</b>

## **6. ПОЛІТИКА**

Політика академічної поведінки та етики:

- не пропускати та не запізнюватися на заняття за розкладом;
- вчасно виконувати завдання практичних занять та самостійної роботи;
- на заняттях дотримуватись правил роботи у групі, шанобливо ставитись до поглядів один одного;
- вчасно та самостійно виконувати контрольні завдання;
- дотримуватись політики доброчесності під час виконання завдань;
- активно брати участь у побудові і реалізації індивідуальної освітньої траєкторії.

Політика щодо дедлайнів та перескладання: роботи, які здаються із порушенням термінів без поважних причин, оцінюються на нижчу оцінку. Перескладання модулів відбувається із дозволу деканату за наявності поважних причин (наприклад, лікарняний).

Політика щодо академічної доброчесності: дотримуватись Положення про академічну доброчесність.

Політика щодо відвідування: відвідування занять є обов'язковим. За об'єктивних причин (наприклад, хвороба, міжнародне стажування) навчання може відбуватись в он-лайн формі за погодженням із керівником курсу.

## **7. СТРУКТУРА ОСВІТНЬОГО КОМПОНЕНТА**

### **III СЕМЕСТР**

#### **7.1. СТРУКТУРА ОСВІТНЬОГО КОМПОНЕНТА (ЗАГАЛЬНА)**

<b>Кількість годин</b>	<b>Тема</b>	<b>Форма діяльності (заняття, кількість годин)</b>	<b>Література</b>	<b>Завдання</b>	<b>Вага оцінки</b>	<b>Термін виконання</b>
<b>Блок 1. Основні конструкції мови програмування C#</b>						
11	Тема 1. Введення до платформи microsoft.net та мови C#.	Лекція (2 год.), практичне заняття (4)	Основна : 1-3, Допомі	Виконання практичних завдань,	5	Перший періодичний контроль

		год.), самот. робота (5 год.)	жна: 4	захист звіту		
16	Тема 2. Основи мови C#.	Лекція (2 год.), практичне заняття (4 год.), самот. робота (10 год.)	Основна : 1-3, Допомі жна: 4	Виконання практичних завдань, захист звіту	5	Перший періодичний контроль
16	Тема 3. Основні інструкції керування мови C# – розгалуження та цикли.	Лекція (2 год.), практичне заняття (4 год.), самот. робота (10 год.)	Основна : 1-3, Допомі жна: 4	Виконання практичних завдань, захист звіту	5	Перший періодичний контроль
18	Тема 4. Перелічення та масиви. Синтаксис. Базовий клас Array.	Лекція (2 год.), практичне заняття (6 год.), самот. робота (10 год.)	Основна : 1-3, Допомі жна: 4	Виконання практичних завдань, захист звіту	5	Перший періодичний контроль
29	Тема 5. Введення до WINDOWS FORMS.	Лекція (2 год.), практичне заняття (8 год.), самот. робота (19 год.)	Основна : 1-3, Допомі жна: 4	Виконання практичних завдань, захист звіту, тестування	5	Перший періодичний контроль
<b>Блок 2. Додаткові можливості технології Windows Forms</b>						
16	Тема 6. Використання строкового типу даних.	Лекція (2 год.), практичне заняття (4 год.), самот. робота (10 год.)	Основна : 1-3, Допомі жна: 4	Виконання практичних завдань, захист звіту	5	Другий періодичний контроль
16	Тема 7. Робота з файловими потоками даних.	Лекція (2 год.), практичне заняття (4 год.), самот. робота (10 год.)	Основна : 1-3, Допомі жна: 4	Виконання практичних завдань, захист звіту	5	Другий періодичний контроль
28	Тема 8. Використання графічних можливостей технології Windows Forms	Лекція (4 год.), практичне заняття (8 год.), самот. робота (16 год.)	Основна : 1-3, Допомі жна: 4	Виконання практичних завдань, захист звіту, тестування	5	Другий періодичний контроль

## **7.2. СТРУКТУРА ОСВІТНЬОГО КОМПОНЕНТА (ЛЕКЦІЙНИЙ БЛОК)**

<b>Теми лекцій та питання, що вивчаються</b>
<b>Середовище розробника програмування Visual Studio. Net.</b>
1. Вступ програмування Net.
2. Робота в середовище Visual Studio.

3. Windows-форми.
<b>Основні поняття мови програмування с#.</b> 1. Типи даних. 2. Основні функції. 3. Операції.
<b>Керуючі оператори мови високого рівня с#</b> 1. Оператор умови. 2. Вибір. 3. Оператори циклу.
<b>Перелічення та масиви. Синтаксис. Базовий клас Array.</b> 1. Формат перелічення. 2. Синтаксис масиву. 3. Робота з базовим класом Array.
<b>Введення до Windows Forms. Основи використання елементів управління. Стандартні діалогові вікна повідомлень.</b> 1. Введення Windows Forms. 2. Стандартні елементи управління. 3. Діалогові вікна. 4. Методи та свойства управління.
<b>Робота з рядками в мові програмування С#. Робота з типом даних String, StringBuilder.</b> 1. Типи даних для роботи з рядковим типом даних. 2. Функції та методи для роботи з рядками. 3. Тип даних StringBuilder.
<b>Робота з файлами. Компоненти для роботи з файлами.</b> 1. Встановлення зв'язку в програмі. 2. Методи та функції для роботи з файловими змінними.
<b>Використання графічних можливостей технології Windows Forms</b> 1. Базові засади організації графічного інтерфейсу. 2. Динамічне додавання компонентів. 3. Створення власних компонентів.

### 7.3. СТРУКТУРА ОСВІТНЬОГО КОМПОНЕНТА (ЛАБОРАТОРНІ ЗАНЯТТЯ)

Тема лабораторного заняття	Зміст лабораторного заняття
Вивчення середовища розробника Visual Studio. Net.	Виконання завдань. Захист лабораторної роботи.
Лінійні алгоритми. Розв'язання задач с умовним оператором	Виконання завдань. Захист лабораторної роботи.
Циклічні алгоритми.	Виконання завдань. Захист лабораторної роботи.
Написання та розробка алгоритмів та програм з одновимірними масивами.	Виконання завдань. Захист лабораторної роботи.
Написання та розробка алгоритмів та програм з двовимірними масивами.	Виконання завдань. Захист лабораторної роботи.
Написання та розробка алгоритмів та програм з рядковим типом даних.	Виконання завдань. Захист лабораторної роботи.
Робота з файлами. Компоненти для роботи з файлами.	Виконання завдань. Захист лабораторної роботи.
Відкриття та збереження даних у файл.	Виконання завдань. Захист лабораторної роботи.

Створення програм з графічним інтерфейсом користувача.	Виконання завдань. Захист лабораторної роботи.
Заливка та рух графічних примітивів.	Виконання завдань. Захист лабораторної роботи.

#### **7.4. СТРУКТУРА ОСВІТНЬОГО КОМПОНЕНТА (ТЕМИ ДЛЯ САМОСТІЙНОГО ОПРАЦЮВАННЯ)**

<b>Теми для самостійного опрацювання</b>
Призначення Visual Studio, .Net технології. Призначення середовища. Налаштування та запуск програми.
Особливості мов програмування C#. Перетворення типів даних. Математичні функції.
Вибір та умова в мові програмування. Складання програм.
Оператори повторення. Застосування та різниця в призначенні.
Робота з масивами. Багатовимірні масиви та ступінчаті. Написання програм з переліченням.
Використання стандартної панелі інструментів. Панель інструментів та програмування кнопок.
Робота з рядковим типом даних. Пошук та заміна в рядках.
Робота з текстовими файлами. Використання класів при роботі з файлами. Компоненти для відкриття та збереження файлів.
Графічні можливості мови програмування C#. Створення своїх графічних примітивів.

#### **8. МЕТОДИ ТА ФОРМИ КОНТРОЛЮ**

Поточний контроль: усне опитування; захист звітів з лабораторних робіт. Періодичний контроль: тести, презентація результатів проєктної діяльності. Підсумкова конференція за результатами навчальної практики. Підсумковий семестровий контроль.

#### **КРИТЕРІЇ ОЦІНЮВАННЯ ВІДПОВІДНО ДО ВИДІВ КОНТРОЛЮ**

<b>Методи контролю результатів навчання</b>	<b>Максимальна кількість балів та критерії до їх зарахування</b>
<b>Поточний контроль (робота на лабораторних заняттях)</b>	Середньозважена оцінка: 20 балів Максимальна оцінка за 1 заняття – 5 балів
<b>Періодичний контроль</b>	<p><b>Разом: 30 балів</b></p> <p><b>Контрольна робота: 10 балів</b> відповіді на 10 тестових завдань: 1 бал – відповідь вірна, 0 балів – відповідь не вірна</p> <p><b>Індивідуальне завдання: 20 балів</b></p> <ul style="list-style-type: none"> <li>– 1 бал – початковий рівень знань теоретичного матеріалу; елементарні, нестійкі навички роботи з інформаційними технологіями;</li> <li>– 2 бали – відтворення матеріалу за аналогією з елементами логічних зв'язків; стійкі навички виконання елементарних дій з опрацювання інформації на комп'ютері;</li> <li>– 3 бали – студент знайомий з основними поняттями навчального матеріалу; може самостійно відтворити значну частину навчального матеріалу і робити певні узагальнення; під час відповіді спирається на</li> </ul>

	<p>відомі зразки; має стійкі знання;</p> <ul style="list-style-type: none"> <li>– 4 бали – студент уміє застосовувати вивчений матеріал у стандартних ситуаціях; може пояснити основні процеси;</li> <li>– 5-6 балів – студент уміє аналізувати навчальну інформацію, в цілому самостійно застосовувати її на практиці, контролювати власну діяльність; самостійно виправляє вказані помилки;</li> <li>– 7-8 балів – студент частково володіє навчальним матеріалом; початкові уміння узагальнювати і систематизувати навчальну інформацію, знаходити і виправляти допущені помилки; не може обрати раціональний спосіб виконання завдання;</li> <li>– 9-10 балів – студент загалом володіє навчальним матеріалом; самостійно знаходить і виправляє допущені помилки; не може обрати раціональний спосіб виконання завдання; початкові уміння узагальнювати і систематизувати навчальну інформацію;</li> <li>– 11-12 балів – відповідь повна, наведені обґрунтування та сформульовані основні принципи та закономірності, проте в доведенні порушено правильний ланцюг міркувань, наведені приклади частково пояснюють матеріал теоретичного питання;</li> <li>– 13-14 балів – студент володіє достатньо ґрунтовними знаннями, вміє оцінювати нові факти, явища; судження його логічні й достатньо обґрунтовані; має достатні знання володіння комп'ютерними технологіями;</li> <li>– 15-16 балів – студент володіє достатньо ґрунтовними знаннями, вміє оцінювати нові факти, явища; судження логічні й достатньо обґрунтовані; відповідь повна, наведені обґрунтування та сформульовані основні принципи та закономірності, проте в доведенні порушено правильний ланцюг міркувань;</li> <li>– 17-18 балів – студент володіє узагальненими знаннями, вміє самостійно використовувати інформацію відповідно до мети і завдань пізнавальної діяльності;</li> <li>– 19-20 балів – студент має стійкі системні знання та продуктивно їх використовує, стійкі навички керування інформаційною системою в нестандартних ситуаціях; відповідь повна, розгорнута з обґрунтованими твердженнями, сформульовані основні принципи та закономірності, вказані причинно-наслідкові зв'язки, наведено приклади.</li> </ul>
<b>Підсумковий контроль (диф. залік)</b>	<p>Обчислюється як сума балів поточної успішності та періодичного контролю. Максимально – 100 балів.</p>

#### *IV СЕМЕСТР*

##### **7.1. СТРУКТУРА ОСВІТНЬОГО КОМПОНЕНТА (ЗАГАЛЬНА)**

Кількість годин	Тема	Форма діяльності (заняття, кількість годин)	Література	Завдання	Вага оцінки	Термін виконання
<b>Блок 1. Реалізація головних концепцій об'єктно-орієнтованого програмування у мові C#</b>						
19	Тема 1. Графічні можливості мови	Лекція (6 год.), практичне	Основна : 1-3,	Виконання практичних	5	Перший періодичний

	програмування C#.	заняття (8 год.), самостійна робота (5 год.)	Допомі жна: 4	завдань, захист звіту		контроль
11	Тема 2. Використання індексаторів та властивостей.	Лекція (4 год.), практичне заняття (2 год.), самостійна робота (5 год.)	Основна : 1-3, Допомі жна: 4	Виконання практичних завдань, захист звіту	5	Перший періодичний контроль
11	Тема 3. Обробка виключень.	Лекція (2 год.), практичне заняття (4 год.), самостійна робота (5 год.)	Основна : 1-3, Допомі жна: 4	Виконання практичних завдань, захист звіту	5	Перший періодичний контроль
11	Тема 4. Колекції	Лекція (2 год.), практичне заняття (4 год.), самостійна робота (5 год.)	Основна : 1-3, Допомі жна: 4	Виконання практичних завдань, захист звіту	5	Перший періодичний контроль
10	Тема 5. Рядки та регулярні вирази	Лекція (2 год.), практичне заняття (4 год.), самостійна робота (4 год.)	Основна : 1-3, Допомі жна: 4	Виконання практичних завдань, захист звіту	5	Перший періодичний контроль
12	Тема 6. Використання атрибутів.	Лекція (4 год.), практичне заняття (4 год.), самостійна робота (4 год.)	Основна : 1-3, Допомі жна: 4	Виконання практичних завдань, захист звіту, тестування	5	Перший періодичний контроль
<b>Блок 2. Введення в патерни проектування</b>						
11	Тема 7. Уніфікована мова моделювання (UML)	Лекція (4 год.), практичне заняття (2 год.), самостійна робота (5 год.)	Основна : 1-3, Допомі жна: 4	Виконання практичних завдань, захист звіту	5	Другий періодичний контроль
11	Тема 8. Принципи проектування програмного забезпечення на основі аналізу предметної області	Лекція (4 год.), практичне заняття (2 год.), самостійна робота (5 год.)	Основна : 1-3, Допомі жна: 4	Виконання практичних завдань, захист звіту	5	Другий періодичний контроль
9	Тема 9. Введення в шаблони проектування	Лекція (2 год.), практичне заняття (2 год.), самостійна робота (5 год.)	Основна : 1-3, Допомі жна: 4	Виконання практичних завдань, захист звіту, тестування	5	Другий періодичний контроль
13	Тема 10. Породжувальні патерни	Лекція (2 год.), практичне заняття (6 год.), самостійна робота (5 год.)	Основна : 1-3, Допомі жна: 4	Виконання практичних завдань, захист звіту, тестування	5	Другий періодичний контроль
15	Тема 11. Вступ до структурних патернів	Лекція (4 год.), практичне	Основна : 1-3,	Виконання практичних	5	Другий періодичний



		заняття (6 год.), самостійна робота (5 год.)	Допомі жна: 4	завдань, захист звіту, тестування		контроль
17	Тема 12. Застосування патернів проектування процесі написання програмних засобів	Лекція (4 год.), практичне заняття (8 год.), самостійна робота (5 год.)	Основна : 1-3, Допомі жна: 4	Виконання практичних завдань, захист звіту, тестування	5	Другий періодичний контроль
<b>Блок 3. Навчальна практика</b>						
60	Навчальна практика	Практичне заняття (40 год.), самостійна робота (20 год.)	Допомі жна: 5- 10	Виконання практичних завдань, захист звіту	100	19.02.2021

## 7.2. СТРУКТУРА ОСВІТНЬОГО КОМПОНЕНТА (ЛЕКЦІЙНИЙ БЛОК)

Теми лекцій та питання, що вивчаються
<p><b>Графічні можливості мови програмування C#.</b></p> <ol style="list-style-type: none"> <li>Особливості графічного виведення даних.</li> <li>Логічна система координат.</li> <li>Простори імен GDI+.</li> <li>Обробка повідомлення перемальовування.</li> <li>Програмне генерування повідомлення перемальовування.</li> <li>Графічні об'єкти GDI+.</li> <li>Використання пензлів, пер та шрифтів.</li> <li>Робота з графічними зображеннями.</li> </ol>
<p><b>Використання індексаторів та властивостей.</b></p> <ol style="list-style-type: none"> <li>Реалізація принципу поліморфізму у мові C#.</li> <li>Віртуальні методи.</li> <li>Абстрактні класи та методи.</li> <li>Реалізація поліморфної поведінки на базі абстрактного класу.</li> <li>Правила застосування абстрактних класів. Інтерфейси. Реалізація поліморфної поведінки на базі інтерфейсу. Правила застосування інтерфейсів.</li> <li>Принципи перевантаження операцій.</li> <li>Особливості використання функції operator.</li> <li>Індексатори.</li> <li>Властивості</li> </ol>
<p><b>Обробка виключень.</b></p> <ol style="list-style-type: none"> <li>Види помилок у програмах.</li> <li>Проблеми традиційного підходу до обробки помилок.</li> <li>Механізм обробки виключень.</li> <li>Класи виключень стандартної бібліотеки .NET. Синтаксис обробки виключень.</li> <li>Перевірка на арифметичне переповнення.</li> </ol>
<p><b>Колекції.</b></p> <ol style="list-style-type: none"> <li>Основні класи та інтерфейси колекцій.</li> <li>Класи колекцій загального призначення. Застосування колекцій при написанні програмного засобу.</li> </ol>
<p><b>Рядки та регулярні вирази</b></p> <ol style="list-style-type: none"> <li>Основні методи класу Char.</li> <li>Основні методи класу String.</li> <li>Основні елементи класу StringBuilder.</li> </ol>

4. Метасимволи в регулярних виразах.
5. Уточнюючі метасимволи в регулярних виразах. Пошук в тексті за шаблоном.

#### **Використання атрибутів.**

1. Використання атрибутів.
2. Призначені для користувача атрибути.
3. Визначення атрибутивного класу.
4. Приклад настоюваного атрибута. Написання програм з використання атрибутів.

#### **Уніфікована мова моделювання (UML)**

1. Призначення уніфікованої мови програмування.
2. Види діаграм та їх призначення.
3. Особливості побудови діаграм.

#### **Принципи проектування програмного забезпечення на основі аналізу предметної області**

1. Поняття проектування програмного забезпечення на основі аналізу предметної області.
2. Визначення аналізу предметної області, еквівалентність понять сутність предметної області та класу об'єктів. Загальні принципи проектування.
3. Типові причин перепроєктування.
4. Якісні характеристики коду: гнучкість, переносність, розширюваність.
5. Механізми повторного використання коду: спадкування, композиція, делегування обов'язків.
6. Плюси і мінуси описаних механізмів, випадки найефективнішого використання

#### **Введення в шаблони проектування.**

1. Передумови виникнення шаблонів проектування.
2. Забезпечення багаторазового використання коду.
3. Класифікація патернів проектування: патерни проектування класів/об'єктів (GoF), архітектурні системні патерни, патерни інтеграції корпоративних інформаційних систем.
4. Класифікація патернів проектування класів/об'єктів: породжуючі, структурні та поведінкові патерни.
5. Шаблон Абстрактна Фабрика (Abstract Factory): визначення, мета використання, переваги та недоліки використання, застосовність, діаграма класів, учасники шаблону, програмна реалізація.

#### **Породжувальні патерни.**

1. Породжувальні патерни – загальне визначення, призначення, способи використання, випадки застосування:
2. Прототип (Prototype), Одинак (Singleton), Фабричний Метод (Factory Method): визначення, мета використання, переваги та недоліки використання, застосовність, діаграма класів, учасники шаблонів, програмна реалізація.
3. Загальні висновки стосовно областей та способів використання породжуючих патернів, порівняльна характеристика.
4. Приклади практичного застосування породжувальних патернів.

#### **Вступ до структурних патернів.**

1. Загальний опис структурних патернів.
2. Структурні патерни рівня класів та об'єктів. Програмні механізми, що використовуються при застосуванні структурних патернів. Спільність, різниця, способи реалізації. Випадки та приклади застосування.
3. **Шаблони Міст (Bridge) та Адаптер (Adapter)** Визначення патерну Міст. Призначення, синонімічні назви, мотивація використання, застосовність, переваги та недоліки використання, діаграма класів, учасники шаблону, відношення учасників, результати використання, програмна реалізація, відомі застосування. Визначення патерну Адаптер.
4. Призначення, синонімічні назви, мотивація використання, застосовність, переваги та недоліки використання, діаграма класів, учасники шаблону, відношення учасників,

результати використання, програмна реалізація, відомі застосування.

### **Застосування патернів проектування процесі написання програмних засобів**

1. Введення до патернів поведінки.
2. Загальні висновки стосовно патернів проектування класів/об'єктів (GoF).

### **7.3. СТРУКТУРА ОСВІТНЬОГО КОМПОНЕНТА (ЛАБОРАТОРНІ ЗАНЯТТЯ)**

<b>Тема лабораторного заняття</b>	<b>Зміст лабораторного заняття</b>
Створення найпростіших графічних зображень.	Виконання завдань. Захист лабораторної роботи.
Створення простих анімацій в мові програмування C#.	Виконання завдань. Захист лабораторної роботи.
Створення графіку функцій за допомогою компонента Chart та за допомогою класу Graphics.	Виконання завдань. Захист лабораторної роботи.
Використання властивостей та індексаторів при розробці програм	Виконання завдань. Захист лабораторної роботи.
Розробка програм з реалізації виключень. Перевірка значень, які вводяться з клавіатури та їх обробка.	Виконання завдань. Захист лабораторної роботи.
Використання регулярних виразів при розробці додатків мовою C#	Виконання завдань. Захист лабораторної роботи.
Використання стандартних колекцій під час розробки програм.	Виконання завдань. Захист лабораторної роботи.
Використання атрибутів в процесі написання програм.	Виконання завдань. Захист лабораторної роботи.
Використання основних бібліотек платформи MICROSOFT .NET при розробці додатків. Інтерфейси.	Виконання завдань. Захист лабораторної роботи.
Моделювання програмного засобу засобами UML	Виконання завдань. Захист лабораторної роботи.
Приклади використання породжувального патерну Фабричний Метод (Factory Method)	Виконання завдань. Захист лабораторної роботи.
Породжувальний патерн Абстрактна Фабрика (Abstract Factory)	Виконання завдань. Захист лабораторної роботи.
Розв'язання задач із використанням структурного патерну Адаптер (Adapter)	Виконання завдань. Захист лабораторної роботи.
Побудова програмних додатків із використанням структурного патерну Фасад (Facade)	Виконання завдань. Захист лабораторної роботи.
Загальні принципи побудови програмних додатків із використанням шаблонів поведінки. Патерн Ітератор (Iterator)	Виконання завдань. Захист лабораторної роботи.
Розробка програмних додатків із використанням патерну поведінки Спостерігач (Observer)	Виконання завдань. Захист лабораторної роботи.

### **7.4. СТРУКТУРА ОСВІТНЬОГО КОМПОНЕНТА (НАВЧАЛЬНА ПРАКТИКА)**

<b>Термін</b>	<b>Завдання</b>
1-й день	Мета і завдання практики. Вимоги до виконання та оформлення звітів. 1. Знайомство з приводною платформою розробки робота. 2. Налаштування середнього мотора приводної платформи. 3. Особливості роботи з датчиками: кольору, ультразвуку, дотику, гіроскоп.

2-й день	<ol style="list-style-type: none"> <li>1. Вивчення обраного програмного середовища для розробки та програмування робота.</li> <li>2. Налаштування робота на рух по чорній лінії з одним датчиком кольору.</li> <li>3. Визначення відповідного кольору на полотні.</li> </ol>
3-й день	<ol style="list-style-type: none"> <li>1. Знайомство з космічним полем.</li> <li>2. Ознайомлення з моделями космічного поля. Зберіть моделі космічного поля «Станція зв'язку», «Супутник», «Екіпаж», «Кратер і MSL», «Зразки порід», «Сонячна батарея», «Ракета і пускова установка», «Станція на Марсі».</li> <li>3. Місія «Активация зв'язку». Спроектуйте, зберіть і запрограмуйте робота, який зможе переміщатися до супутникової тарілки і приводити її в вертикальне положення. <ul style="list-style-type: none"> <li>● Розробка інструкції із створення робота.</li> <li>● Створення програми для виконання місії.</li> </ul> </li> </ol>
4-й день	<ol style="list-style-type: none"> <li>1. Виконання місії «Комплектація екіпажу». Спроектуйте, побудуйте і запрограмуйте робота, який зможе переміститися на місячну базу, забрати командира екіпажу і висадити її на стартовому майданчику. <ul style="list-style-type: none"> <li>● Розробка інструкції із створення робота.</li> <li>● Створення програми для виконання місії.</li> </ul> </li> </ol>
5-й день	<ol style="list-style-type: none"> <li>1. Виконання місії «Звільнення робота MSL». Спроектуйте, побудуйте і запрограмуйте робота, здатного переміститися до кратера і звільнити робота MSL, шість коліс якого повинні знову опинитися на поверхні Марса. <ul style="list-style-type: none"> <li>● Розробка інструкції із створення робота.</li> <li>● Створення програми для виконання місії.</li> </ul> </li> </ol>
6-й день	<ol style="list-style-type: none"> <li>1. Виконайте місію «Запуск супутника». Спроектуйте, побудуйте і запрограмуйте робота, який зможе помістити Супутник в зазначену область на навчальному полі. <ul style="list-style-type: none"> <li>● Розробка інструкції із створення робота.</li> <li>● Створення програми для виконання місії.</li> </ul> </li> <li>2. Виконайте місію «Доставка зразків порід». Спроектуйте, зберіть і запрограмуйте робота, який зможе переміщатися до зразків порід, збирати їх і доставляти на стартовий майданчик. <ul style="list-style-type: none"> <li>● Розробка інструкції із створення робота.</li> <li>● Створення програми для виконання місії.</li> </ul> </li> </ol>
7-й день	<ol style="list-style-type: none"> <li>1. Виконайте місію «Забезпечення енергопостачання». Спроектуйте, побудуйте і запрограмуйте робота, здатного дістатися до Сонячної панелі і повернути ручку, щоб розкрити її. <ul style="list-style-type: none"> <li>● Розробка інструкції із створення робота.</li> <li>● Створення програми для виконання місії.</li> </ul> </li> </ol>
8-й день	<ol style="list-style-type: none"> <li>1. Виконайте місію «Ініціювання запуску». Спроектуйте, побудуйте і запрограмуйте робота, здатного переміститися до пускової установки і натиснути кнопку, яка запустить Ракету і активує Станцію на Марсі. <ul style="list-style-type: none"> <li>● Розробка інструкції із створення робота.</li> <li>● Створення програми для виконання місії.</li> </ul> </li> </ol>
9-й день	Підготовка звіту з практики.
10-й день	Звіт практиканта перед керівником практики.

## 7.5. СТРУКТУРА ОСВІТНЬОГО КОМПОНЕНТА (ТЕМИ ДЛЯ САМОСТІЙНОГО ОПРАЦЮВАННЯ)

Теми для самостійного опрацювання
Відношення агрегації. Реалізація агрегації у мові C#.
Відношення спадкування. Ієрархія класів. Синтаксис спадкування у мові C#.
Управління механізмом доступу до членів базового класу. Механізми використання захищених членів класу. Використання специфікатора доступу protected для надання членам класу статусу захищеності.
Поняття про віртуальні базові класи. Перевизначення методів. Заборона спадкування.
Рядкове представлення об'єкта.
Показники на похідні типи – підтримка динамічного поліморфізму. Механізми реалізації віртуальних функцій.
Поняття про віртуальні функції. Успадкування віртуальних функцій.
Поняття про суто віртуальні функції та абстрактні класи. Класові ієрархії та абстрактні класи.
Раннє та пізнє зв'язування.
Контейнер. Ієрархія об'єктів.
Організація файлового введення-виведення даних. Відкриття та закриття файлу. Зчитування та запис текстових файлів.
Неформатне введення-виведення даних у двійковому режимі. Зчитування та записування у файл блоків даних. Використання функції eof() для виявлення кінця файлу.

## 8. МЕТОДИ ТА ФОРМИ КОНТРОЛЮ

Поточний контроль: усне опитування; захист звітів з лабораторних робіт. Періодичний контроль: тести, презентація результатів проєктної діяльності. Підсумкова конференція за результатами навчальної практики. Підсумковий семестровий контроль.

### КРИТЕРІЇ ОЦІНЮВАННЯ ВІДПОВІДНО ДО ВИДІВ КОНТРОЛЮ

Методи контролю результатів навчання	Максимальна кількість балів та критерії до їх зарахування
<b>Поточний контроль (робота на лабораторних заняттях)</b>	Середньозважена оцінка: 20 балів Максимальна оцінка за 1 заняття – 5 балів
<b>Періодичний контроль</b>	<p><b>Разом: 30 балів</b></p> <p><b>Контрольна робота: 10 балів</b> відповіді на 10 тестових завдань: 1 бал – відповідь вірна, 0 балів – відповідь не вірна</p> <p><b>Індивідуальне завдання: 20 балів</b></p> <ul style="list-style-type: none"> <li>– 1 бал – початковий рівень знань теоретичного матеріалу; елементарні, нестійкі навички роботи з інформаційними технологіями;</li> <li>– 2 бали – відтворення матеріалу за аналогією з елементами логічних зв'язків; стійкі навички виконання елементарних дій з опрацювання інформації на комп'ютері;</li> <li>– 3 бали – студент знайомий з основними поняттями навчального матеріалу; може самостійно відтворити значну частину навчального</li> </ul>

	<p>матеріалу і робити певні узагальнення; під час відповіді спирається на відомі зразки; має стійкі знання;</p> <ul style="list-style-type: none"> <li>– 4 бали – студент уміє застосовувати вивчений матеріал у стандартних ситуаціях; може пояснити основні процеси;</li> <li>– 5-6 балів – студент уміє аналізувати навчальну інформацію, в цілому самостійно застосовувати її на практиці, контролювати власну діяльність; самостійно виправляє вказані помилки;</li> <li>– 7-8 балів – студент частково володіє навчальним матеріалом; початкові уміння узагальнювати і систематизувати навчальну інформацію, знаходити і виправляти допущені помилки; не може обрати раціональний спосіб виконання завдання;</li> <li>– 9-10 балів – студент загалом володіє навчальним матеріалом; самостійно знаходить і виправляє допущені помилки; не може обрати раціональний спосіб виконання завдання; початкові уміння узагальнювати і систематизувати навчальну інформацію;</li> <li>– 11-12 балів – відповідь повна, наведені обґрунтування та сформульовані основні принципи та закономірності, проте в доведенні порушено правильний ланцюг міркувань, наведені приклади частково пояснюють матеріал теоретичного питання;</li> <li>– 13-14 балів – студент володіє достатньо ґрунтовними знаннями, вміє оцінювати нові факти, явища; судження його логічні й достатньо обґрунтовані; має достатні знання володіння комп'ютерними технологіями;</li> <li>– 15-16 балів – студент володіє достатньо ґрунтовними знаннями, вміє оцінювати нові факти, явища; судження логічні й достатньо обґрунтовані; відповідь повна, наведені обґрунтування та сформульовані основні принципи та закономірності, проте в доведенні порушено правильний ланцюг міркувань;</li> <li>– 17-18 балів – студент володіє узагальненими знаннями, вміє самостійно використовувати інформацію відповідно до мети і завдань пізнавальної діяльності;</li> <li>– 19-20 балів – студент має стійкі системні знання та продуктивно їх використовує, стійкі навички керування інформаційною системою в нестандартних ситуаціях; відповідь повна, розгорнута з обґрунтованими твердженнями, сформульовані основні принципи та закономірності, вказані причинно-наслідкові зв'язки, наведено приклади.</li> </ul>
<b>Підсумковий контроль (іспит)</b>	<p>Теоретичне питання: 45 балів.          Практичне завдання: 45 балів          Тести: 10 балів (10 питань, вірна відповідь – 1 бал)</p>

### **КРИТЕРІЇ ОЦІНЮВАННЯ НАВЧАЛЬНОЇ ПРАКТИКИ**

<b>Вид роботи</b>	<b>Максимальна кількість балів та вимоги до їх накопичення</b>
Виконання навчального плану	0-20 2 бали за кожний день присутності на базі практики
Створення моделі робота	0-10 0 - не надрукована 1-4 - надрукована зі значними помилками 5-7 - надрукована із незначними помилками 8-10 - надрукована точно за моделлю, без помилок
Розробка інструкції із створення робота	0-40 0 – за невиконане завдання;

	<p>1-5 – за епізодично виконане завдання;          6-10 – за невиконане до кінця завдання із серйозними помилками;          11-15 – частково неповне і несвоєчасне виконання завдання;          16-20 - своєчасне виконання завдання;          21-25 - повне і своєчасне виконання завдання, але з незначними недоліками          26-30 - повне і своєчасне виконання завдання та правильне виконання алгоритму</p>
Якість оформлення звіту	<p>0-20          0 - звіт не надано, або він не відповідає встановленим вимогам щодо оформлення та змісту;          1-6 - зауваження щодо оформлення звіту мають істотний характер, однак за змістом звіт засвідчує, що програма практики переважно виконана;          7-12 - звіт уміщує передбачену у вимогах інформацію та засвідчує, що програму практики виконано повністю і на належному рівні, але при цьому містить поодинокі незначні порушення вимог щодо його технічного оформлення;          13-18 - звіт практики оформлено з додержанням усіх установлених вимог; він містить всю необхідну інформацію і засвідчує, що програму практики виконано повністю і на належному рівні, проте є незначні недоліки в описі чи оформленні;          19-20 - звіт практики оформлено з повним додержанням усіх установлених вимог; він містить всю необхідну інформацію і засвідчує, що програму практики виконано повністю і на високому рівні.</p>
Захист роботи	<p>0-10          0 - усний захист не здійснено або під час усного захисту студент не дає задовільних відповідей по суті, не орієнтується у програмі практики або припускається грубих помилок, які свідчать про неналежне виконання програми практики;          1-4 - студент демонструє поверхневі знання, поверхнево орієнтується у змісті поданого звіту, не на всі запитання відповідає по суті;          5-8 - студент демонструє належні знання, переважно орієнтується у змісті поданого звіту та в програмі практики, однак у відповідях на запитання припускається окремих неточностей;          9-10 - студент демонструє ґрунтовні знання, вільно орієнтується у змісті поданого звіту та в програмі практики в цілому, чим підтверджує її виконання; надає правильні й аргументовані відповіді на всі запитання з програми практики.</p>
Разом	100

## **9. РЕКОМЕНДОВАНА ЛІТЕРАТУРА**

### **ОСНОВНА ЛІТЕРАТУРА**

1. Булгакова О.С. Інформатика: візуальне програмування: навч.-метод. посіб. / О.С. Булгакова, В.В. Зосімов, Н.А. Броницька, Н.В. Танкова; за ред. О.С. Булгакової, - Херсон: Олді –Плюс. 2017. 312 с.
2. Ковалюк Т.В. Алгоритмізація та програмування: підручник для студ-тів вищих навч. закладів / Т.В. Ковалюк; за заг. ред. В.В. Пасічника Л.: Магнолія-2006. 2013. 399с.

3. Коноваленко І.В. Програмування мовою С# 7.0 : навчальний посібник / Коноваленко І.В., Марущак П.О., Савків В.Б. – Тернопіль : Тернопільський національний технічний університет імені Івана Пулюя, 2017. – 300 с.

### **ДОПОМІЖНА ЛІТЕРАТУРА**

4. Глибовець А. М. Практикум з мови програмування: навч. посіб./ А. М. Глибовець, М. М. Глибовець, В. С. Проценко. К.: ВД "Києво-Могилянська академія". 2010. 209 с.

5. Валентина Донченко. Робототехніка в Україні: розробки і перспективи [Електронний ресурс]. - М. : Промислове виробництво - 2017, Режим доступу: <https://dou.ua/lenta/articles/robotics-in-ukraine/> - Назва з екрану.

6. Робототехніка і освіта: школа, університет і виробництво: наук.- практ. журн. / Г засн. К.А. Гаврилов, Д.А. Гагаріна, А.Г. Кузнецов та ін .; голів. ред. Д.А. Гагаріна. - К. : Перм, 2018 - 144 с.

7. Філіппов, С. А. Досвід технологічного навчання школярів на основі робототехніки [Текст] / С. А. Філіппов // Теорія і методика навчання технології. Школа і виробництво: навч. посібник / С. А. Філіппов - 2015. - Розд. 1. - С. 21-28.

8. Кадемія М. Ю. Інформаційно-КОМУНІКАЦІЙНІ технології навчання / термінологічний словник // М. Ю. Кадемія. - Л.: Сполом, 2009. - 260 с.

9. Тарапата, В.В. Конструюємо роботів на LEGO MINDSTORMS Education EV3. Таємний код Семюела Морзе / В.В. Тарапата. - 2-е вид. - Москва: Лабораторія знань, 2017. - С. - 53-55

10. Вчимося готувати в середовищі Скретч. Придумай - запрограмують - поділися. Версія 2.0. [Електронний ресурс]: - Євген Патаракіна // - Режим доступу: <http://rcokoit.ru/dld/metodsupport/scratch-2008.pdf>

### **ІНФОРМАЦІЙНІ РЕСУРСИ В ІНТЕРНЕТІ**

1. Сучасні технології програмування [Електронний ресурс]. – Режим доступу: <http://www.ksuonline.ksu.ks.ua>

2. Хронологія робототехніки [Електронний ресурс]. - Мій робот. Режим доступу: [http://www.myrobot.ru/articles/hist\\_2005.php](http://www.myrobot.ru/articles/hist_2005.php)

3. Коротка історія робототехніки [Електронний ресурс]: Киборги / Василь Дорнобергер. - Режим доступу: <http://kiborgs.ru/publ/9-1-0-81>. Доступ з екрану. 117

4. Історія робототехніки: факти [Електронний ресурс] // ПРОробот.ру. / Володимир Канівець. - Режим доступу: [http://www.prorobot.ru/slovarik/is\\_istorii\\_robototehniki.php](http://www.prorobot.ru/slovarik/is_istorii_robototehniki.php). Доступ з екрану.

5. Каталог програм [Електронний ресурс]: - Режим доступу: <http://www.legoengineering.com/category/support/building-instructions/>

6. Офіційний сайт LEGO Education. [Електронний ресурс] - Режим доступу: <http://education.lego.com/>

7. LEGO-education. Початкова школа. [Електронний ресурс]: - Режим доступу: <https://education.lego.com/ru-ru>

8. LEGO Engineering. Inspiration and support for teachers. [Електронний ресурс]: - Режим доступу: <http://www.legoengineering.com/>

9. Книга вчителя LEGO Education WeDo [Електронний ресурс]: - Режим доступу: <https://education.lego.com/ru-ru/product/wedo-2/software>