

**МЕЛІТОПОЛЬСЬКИЙ ДЕРЖАВНИЙ ПЕДАГОГІЧНИЙ УНІВЕРСИТЕТ  
ІМЕНІ БОГДАНА ХМЕЛЬНИЦЬКОГО**

**ФАКУЛЬТЕТ ІНФОРМАТИКИ, МАТЕМАТИКИ ТА ЕКОНОМІКИ**

**Кафедра інформатики і кібернетики**

<b>Назва курсу</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/ V семестр / III курс
<b>Викладач</b>	Конюхов С.Л.
<b>Профайл викладача</b>	<a href="http://inf.mdpu.org.ua/2017/10/25/konjuhov-sergij-leonidovich/">http://inf.mdpu.org.ua/2017/10/25/konjuhov-sergij-leonidovich/</a>
<b>Контактний тел.</b>	+38 050 9293065
<b>Е-mail:</b>	konukhov@mdpu.org.ua
<b>Сторінка курсу в ЦОДТ МДПУ ім. Б.Хмельницького</b>	<a href="https://dfn.mdpu.org.ua/course/view.php?id=369">https://dfn.mdpu.org.ua/course/view.php?id=369</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 «ПП-02 Програмування» з циклу нормативних компонентів здобувачів вищої освіти за освітньою програмою «Професійна освіта. Комп'ютерні технології» передбачає формування у здобувачів поглиблених знань, умінь і навичок в програмуванні.

Основою вивчення дисципліни є отримання здобувачами комплексу теоретичних знань і практичних навичок зі створення статичних і динамічних web-сторінок. Оволодіння такими знаннями дозволить реалізовувати задачі автоматизації обробки інформації, автоматизації керування об'єктами за допомогою комп'ютерної техніки.

Здобувачі мають можливість ознайомитись з будовою web-додатків, принципами функціонування клієнтської частини, видами програмного забезпечення та етапами розв'язування задач за допомогою комп'ютера. А також здійснювати програмування web-орієнтованих систем різного рівня складності з використанням мови програмування JavaScript, використовувати сучасні готові бібліотеки модулів для побудови професійних web-сайтів.

## **2. МЕТА ТА ЗАВДАННЯ КУРСУ**

Основна мета засвоєння курсу полягає у формуванні у здобувачів вищої освіти компетентностей, необхідних для ефективного виконання завдань з розробки клієнтської частини web-додатків та навчання студентів закладів професійної (професійно-технічної) та фахової передвищої освіти веб-програмування.

Завданнями курсу є ознайомлення здобувачів з фундаментальними теоретичними знаннями з програмування та оптимізації web-сайтів, які використовуються при створенні web-орієнтовних інформаційних систем, а також прикладних практичних навиків із застосуванням інструментарію створення web-сайтів.

## **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.

**ФК 23.** Здатність застосовувати методи та засоби захисту даних в інформаційних системах та комп'ютерних мережах.

## **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 26.** Уміння використовувати методи, технології та інструментальні засоби для проектування і розробки веб-застосувань, проектувати концептуальні, логічні та фізичні моделі баз

даних, розробляти запити до них, створювати розподілені бази даних, інтелектуальні інформаційні системи.

**ПРН 29.** Знання концепції інформаційної безпеки, принципів попередження погроз, безпеки комп'ютерних мереж та інформаційних систем.

**ПРН 31.** Уміння здійснювати підбір і підготовку інформації та задач проектній команді, ставити цілі і формулювати завдання для реалізації проектів у галузях професійної освіти та інформаційних технологій.

**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 КУРСУ

Вид заняття	Загальна кількість	Лекції	Практичні/ лабораторні заняття	Самостійна робота
Кількість годин	120 годин	16 годин	24 годин	80 годин

## 6. ПОЛІТИКА КУРСУ

Політика навчання через дослідження:

➤ Курс є складовою освітньо-професійної програми, тому усі його складові розглядаються у контексті відповідності наукових інтересів бакалаврів.

Політика академічної поведінки та етики:

➤ Не пропускати та не запізнюватися на заняття за розкладом;  
➤ Вчасно виконувати завдання семінарів та питань самостійної роботи;  
➤ Вчасно та самостійно виконувати контрольні-модульні завдання.  
➤ Дотримуватись Кодексу академічної доброчесності, прийнятого у МДПУ імені Богдана Хмельницького [https://mdpu.org.ua/wp-content/uploads/2020/11/Kodeks-akadem-dobrochesnosti\\_2020.pdf](https://mdpu.org.ua/wp-content/uploads/2020/11/Kodeks-akadem-dobrochesnosti_2020.pdf) та Положення про Академічну доброчесність [https://mdpu.org.ua/wp-content/uploads/2020/11/akademichna-dobrochesnist\\_2020.pdf](https://mdpu.org.ua/wp-content/uploads/2020/11/akademichna-dobrochesnist_2020.pdf). Здобувачі освіти мають самостійно виконувати навчальні завдання, завдання поточного та періодичного контролю, самостійні завдання, посилаючись на джерела інформації у разі запозичень ідей, тверджень, відомостей; дотримуватись норм законодавства про авторське право.

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

Політика щодо академічної доброчесності: усі письмові роботи перевіряються на наявність плагіату і допускаються до захисту із коректними текстовими запозиченнями не більше 20%. Списування під час контрольних робіт заборонені (в т.ч. із використанням мобільних девайсів). 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.

Поточний контроль: усне опитування в ході лекцій та практичних занять, перевірка завдань (у тому числі самостійної роботи), оцінювання правильності вирішення тестових та практичних завдань на семінарських заняттях.

## 7. СТРУКТУРА КУРСУ

Кількість годин	Тема	Форма діяльності (заняття, кількість годин)	Література	Завдання	Вага оцінки	Термін виконання
<b>Блок 1. Вступ до програмування з використанням JavaScript</b>						
4	<b>Тема 1. Мова програмування JavaScript</b>	Лекція (2 год.), самостійна робота (2 год.)	Основна: 1-4	Опитування перевірка результатів самостійної роботи	5	впродовж п'ятого навчального семестру (перший періодичний контроль)
8	<b>Тема 2. Вікна, документи, вкладені елементи</b>	Практична робота (2 год.), самостійна робота (6 год.)	Основна: 1-4	Опитування перевірка результатів практичної роботи	5	впродовж п'ятого навчального семестру (перший періодичний контроль)
8	<b>Тема 3. Форми</b>	Практична робота (2 год.), самостійна робота (6 год.)	Основна: 1-4	Опитування перевірка результатів практичної роботи	5	впродовж п'ятого навчального семестру (перший періодичний контроль)
10	<b>Тема 4. Події</b>	Лекція (2 год.), практична робота (2 год.), самостійна робота (6 год.)	Основна: 1-4	Опитування перевірка результатів практичної роботи	5	впродовж п'ятого навчального семестру (перший періодичний контроль)
<b>Блок 2. Розширені можливості JavaScript</b>						
7	<b>Тема 5. Регулярні вираження</b>	Практична робота (2 год.), самостійна робота (5 год.)	Основна: 1-4	Опитування перевірка результатів практичної роботи	5	впродовж п'ятого навчального семестру (перший періодичний контроль)

9	<b>Тема 6. Функції</b>	Лекція (2 год.), практична робота (2 год.), самостійна робота (5 год.)	Основн а: 1-4	Опитування перевірка результатів практичної роботи	5	впродовж п'ятого навчального семестру (перший періодичний контроль)
5	<b>Тема 7. Деякі інші можливості</b>	Самостійна робота (5 год.)	Основн а: 1-4	Опитування перевірка результатів самостійної роботи	5	впродовж п'ятого навчального семестру (перший періодичний контроль)
9	<b>Тема 8. Малювання на Canvas</b>	Лекція (2 год.), практична робота (2 год.), самостійна робота (5 год.)	Основн а: 1-4	Опитування перевірка результатів практичної роботи	5	впродовж п'ятого навчального семестру (перший періодичний контроль)
<b>Блок 3. Об'єктно-орієнтоване програмування</b>						
10	<b>Тема 9. Створення об'єктів</b>	Лекція (2 год.), практична робота (2 год.), самостійна робота (6 год.)	Основн а: 1-4	Опитування перевірка результатів практичної роботи	5	впродовж п'ятого навчального семестру (другий періодичний контроль)
11	<b>Тема 10. Модель DOM</b>	Лекція (2 год.), практична робота (2 год.), самостійна робота (7 год.)	Основн а: 1-4	Опитування перевірка результатів практичної роботи	5	впродовж п'ятого навчального семестру (другий періодичний контроль)
9	<b>Тема 11. Поглиблена характеристика подій</b>	Практична робота (2 год.), самостійна робота (7 год.)	Основн а: 1-4	Опитування перевірка результатів практичної роботи	5	впродовж п'ятого навчального семестру (другий періодичний контроль)
<b>Блок 4. Бібліотеки та інтерфейси</b>						

12	<b>Тема 12. Бібліотека jQuery</b>	Лекція (2 год.), практична робота (4 год.), самостійна робота (6 год.)	Основна: 1-4	Опитування перевірка результатів практичної роботи	5	впродовж п'ятого навчального семестру (другий періодичний контроль)
10	<b>Тема 14. AJAX і COMET</b>	Лекція (2 год.), практична робота (2 год.), самостійна робота (6 год.)	Основна: 1-4	Опитування перевірка результатів практичної роботи	5	впродовж п'ятого навчального семестру (другий періодичний контроль)
4	<b>Тема 15. Intl: інтернаціоналізація у JavaScript</b>	Самостійна робота (4 год.)	Основна: 1-4	Опитування перевірка результатів самостійної роботи	5	впродовж п'ятого навчального семестру (другий періодичний контроль)
4	<b>Тема 16. Сучасні можливості ES-2015</b>	Самостійна робота (4 год.)	Основна: 1-4	Опитування перевірка результатів самостійної роботи	5	впродовж п'ятого навчального семестру (другий періодичний контроль)

## 7.2. СТРУКТУРА КУРСУ (ЛЕКЦІЙНИЙ БЛОК)

Теми лекцій та питання, що вивчаються
<b>Тема 1. Мова програмування JavaScript</b> <ol style="list-style-type: none"> <li>1. Призначення.</li> <li>2. Вбудовування у сторінку.</li> <li>3. Типи даних і перемінні.</li> <li>4. Перетворення типів.</li> <li>5. Глобальний об'єкт <i>Math</i> і його функції.</li> <li>6. Масиви.</li> <li>7. Рядки.</li> <li>8. Дати.</li> <li>9. Вікна.</li> <li>10. Документи.</li> <li>11. Зображення.</li> <li>12. Основи подій</li> </ol>
<b>Тема 2. Форми. Події</b>

1. *Форми. Призначення і особливості використання форм.*
2. *Текстові поля.*
3. *Об'єкти CheckBox і Radio .*
4. *Списки вибору. Кнопки.*
5. *Приховані поля.*
6. *Події. Основні види.*
7. *Присвоювання обробників.*
8. *Властивості подій.*

**Тема 3. Регулярні вираження. Функції**

1. *Регулярні вираження.*
2. *Патерни і прапори.*
3. *Методи RegExр і String.*
4. *Класи і спецсимволи.*
5. *Набори і діапазони [...].*
6. *Квантифікатори +, \*, ? і {n}.*
7. *Дужкові групи.*
8. *Зворотні посилання: \n і \$n.*
9. *Альтернація (або) |.*
10. *Початок ^ і кінець \$ рядка.*
11. *Багаторядковий режим, прапор "m".*
12. *Функції.*
13. *Області видимості.*
14. *Замикання.*
15. *Відмінності між var і let.*
16. *Псевдомасив аргументів "arguments".*
17. *Тип значення і тип посилання.*

**Тема 4. Малювання на Canvas**

1. *Об'єкт canvas.*
2. *Методи.*
3. *Властивості.*
4. *2d-контекст: методи і властивості.*
5. *Базове використання.*
6. *Малювання фігур.*
7. *Встановлення стилів і кольорів.*
8. *Малювання тексту.*

**Тема 5. Створення об'єктів. ООП у функціональному стилі**

1. *Вступ.*
2. *Створення об'єктів і робота з властивостями.*
3. *Методи об'єктів.*
4. *Перебір властивостей об'єкта.*
5. *Конструктор об'єктів.*
6. *Внутрішній і зовнішній інтерфейс.*
7. *Геттери і сеттери.*
8. *Функціональне наслідування.*

**Тема 6. Модель DOM. Поглиблена характеристика подій**

1. *Модель DOM.*
2. *Дерево DOM.*
3. *Робота з DOM з консолі.*
4. *Навігація по DOM-елементам.*
5. *Пошук: getElement\*, querySelector\* та інші способи.*
6. *Внутрішній устрій пошукових методів.*
7. *Властивості вузлів: тип, тег і зміст.*

8. Сучасний DOM: поліфіли.
9. Атрибути і DOM-властивості.
10. Поглиблена характеристика подій.
11. Події від клавіатури.
12. Події від миші.
13. Високорівневі події.
14. Порядок слідування подій.
15. Спливання і занурення подій.
16. Зупинення послідовності обробки подій.

**Тема 7. Бібліотека jQuery**

1. *Поняття про jQuery.*
2. *Селектори jQuery.*
3. *Атрибути jQuery.*
4. *Фільтри.*
5. *Фільтри Форм.*
6. *jQuery CSS.*
7. *DOM Маніпуляції.*
8. *Обхід DOM.*
9. *Події Базові.*
10. *Події Основні.*
11. *Ефекти, анімація.*
12. *Утиліти.*
13. *Функції Ядра.*

**Тема 8. AJAX і COMET**

1. *Вступ до AJAX і COMET.*
2. *Основи XMLHttpRequest.*
3. *XMLHttpRequest POST, форми і кодування.*
4. *XMLHttpRequest: крос-доменні запити.*
5. *XMLHttpRequest: індикація прогресу.*
6. *XMLHttpRequest: поновлюване завантаження.*
7. *COMET з XMLHttpRequest: довгі опитування.*
8. *WebSocket.*
9. *Протокол JSONP.*

### 7.3. СТРУКТУРА КУРСУ (ПРАКТИЧНЕ ЗАНЯТТЯ)

<b>Теми практичних занять</b>
<p><b>Тема:</b> Вікна, документи, вкладені елементи  <i>Виконання практичних завдань згідно варіантів</i></p>
<p><b>Тема:</b> Форми  <i>Виконання практичних завдань згідно варіантів</i></p>
<p><b>Тема:</b> Події  <i>Виконання практичних завдань згідно варіантів</i></p>
<p><b>Тема:</b> Регулярні вирази  <i>Виконання практичних завдань згідно варіантів</i></p>
<p><b>Тема:</b> Функції  <i>Виконання практичних завдань згідно варіантів</i></p>



<p><b>Тема:</b> Малювання на Canvas Виконання практичних завдань згідно варіантів</p>
<p><b>Тема:</b> Створення об'єктів Виконання практичних завдань згідно варіантів</p>
<p><b>Тема:</b> Модель DOM Виконання практичних завдань згідно варіантів</p>
<p><b>Тема:</b> Поглиблена характеристика подій Виконання практичних завдань згідно варіантів</p>
<p><b>Тема:</b> Бібліотека jQuery Виконання практичних завдань згідно варіантів</p>
<p><b>Тема:</b> AJAX і COMET Виконання практичних завдань згідно варіантів</p>

#### 7.4 СТРУКТУРА КУРСУ (ТЕМИ ДЛЯ САМОСТІЙНОГО ОПРАЦЮВАННЯ)

Тема для самостійного опрацювання
<p><b>Тема:</b> Вступ до програмування з використанням JavaScript <b>Питання:</b> Мова програмування JavaScript Відмінність мови JavaScript від C++. Навігатор. Об'єкт Location. Якорі. Гіперпосилання. Приклади використання JavaScript Рядок стану . Таймери. Загрузка зображень. Анімація. Перевірка даних у формі.</p>
<p><b>Тема:</b> Розширені можливості JavaScript <b>Питання:</b> Деякі інші можливості Типи даних: [[Class]], instanceof і качки. Формат JSON, метод toJSON. Методи setTimeout і setInterval. Запуск коду з рядка: eval. Перехоплення помилок "try..catch". Малювання на Canvas Використання зображень. Трансформації. Композиція і обрізка. Прості і розширені анімації. Маніпуляція пікселями. Досягнення областей і доступність. Оптимізація canvas.</p>
<p><b>Тема:</b> Об'єктно-орієнтоване програмування <b>Питання:</b> Створення об'єктів. ООП у прототипному стилі Прототип об'єкта. Властивість F.prototype і створення об'єктів через new. Вбудовані "класи" в JavaScript. Власні класи на прототипах. Наслідування класів у JavaScript. Перевірка класу: "instanceof". Власні помилки, наслідування від Error. Домішки. Модель DOM Методи contains і compareDocumentPosition. Додавання і видалення вузлів. Мультівставка: insertAdjacentHTML і DocumentFragment. Метод document.write. Стили, getComputedStyle. Розміри і прокрутка елементів і сторінки. Координати у вікні і у документі. Докладні відомості про окремі події</p>

Фокусування: focus/blur. Змінювання: change, input, cut, copy, paste. Миша: mouseover/out, mouseenter/leave, Drag'n'Drop, коліщатко, подія wheel, IE8-, виправлення події. Прокручування: подія scroll. Клавіатура: keyup, keydown, keypress. Завантаження документа: DOMContentLoaded, load, beforeunload, unload. Завантаження скриптів, картинок, фреймів: onload и onerror.

**Тема:** Бібліотеки та інтерфейси

**Питання:**

Елементи керування jQuery

User Interface: вступ. Взаємодія. Віджети. Ефекти.

Intl: інтернаціоналізація у JavaScript

Основні об'єкти. Локаль. Рядки, Intl.Collator. Дати, Intl.DateFormatter. Числа: Intl.NumberFormat.

Методи в Date, String, Number. Застарілі версії IE.

Сучасні можливості ES-2015

Вступ до ES-2015. Перемінні: let і const. Деструктуризація. Функції. Рядки. Об'єкти і прототипи.

Класи. Тип даних Symbol. Ітератори. Set, Map, WeakSet і WeakMap. Promise. Генератори. Модулі.

Proху.

## 8. ФОРМИ КОНТРОЛЮ І МЕТОДИ НАВЧАННЯ

Оцінювання результатів навчання здобувачів вищої освіти здійснюється відповідно до «Положення про бально-накопичувальну систему оцінювання результатів навчання здобувачів вищої освіти у МДПУ імені Богдана Хмельницького» (протокол №5 від 24.10.2019) <https://drive.google.com/file/d/1BDRNtAJupqmHkldtICJTKvL-LNTIjWRX/view>.

Усне та письмове опитування, перевірка виконання практичних завдань, презентація результатів проєктної діяльності (захист курсової роботи), тестування, захист лабораторних робіт, поточний контроль, підсумковий семестровий контроль.

**Методи навчання.** Студентсько-центроване навчання. Професійно-орієнтоване навчання, індивідуально-творчий підхід. Очний (*offline*) у вигляді лекційних та семінарських занять. Змішаний (*blended*) через систему Центру освітніх дистанційних технологій МДПУ імені Б.Хмельницького, Zoom, Інтернет. Усі складові курсу розглядаються у контексті відповідності наукових інтересів бакалаврів. Словесні методи (бесіди та дискусії, розповідь, пояснення, лекція); наочні методи (ілюстрування, демонстрування та самостійне спостереження); лабораторні роботи; проблемно-пошукові методи; методи стимулювання та мотивації навчально-пізнавальної діяльності; інтерактивні методи (частково пошукові методи, дослідні методи (проєкт), мозковий штурм, робота в мікрогрупах), самостійна робота студентів.

## 9. СИСТЕМА ОЦІНЮВАННЯ ТА ВИМОГИ

<b>Загальна система оцінювання курсу</b>	<i>За семестр з курсу дисципліни проводяться два періодичні контролю (ПКР), результати яких є складником результатів контрольних точок першої (КТ1) і другої (КТ2). Результати контрольної точки (КТ) є сумою поточного (ПК) і періодичного контролю (ПКР): <math>КТ = ПК + ПКР</math>. Максимальна кількість балів за контрольну точку (КТ) складає 50 балів. Максимальна кількість балів за періодичний контроль (ПКР) становить 60 % від максимальної кількості балів за контрольну точку (КТ), тобто 30 балів. А 40 % балів, тобто решта балів контрольної точки, є бали за поточний контроль, а саме 20 балів. Результати поточного контролю обчислюються як середньозважена оцінок (<math>\bar{X}</math>) за</i>
--	--

	<p>діяльність студента на практичних (семінарських) заняттях, що входять в число певної контрольної точки. Для трансферу середньозваженої оцінки (<math>X_{\text{ср}}</math>) в бали, що входять до 40 % балів контрольної точки (КТ), треба скористатися формулою: <math>ПК = (X_{\text{ср}}) * 20 / 5</math>. Таким чином, якщо за поточний контроль (ПК) видів діяльності студента на всіх заняттях <math>X_{\text{ср}} = 4.1</math> бали, які були до періодичного контролю (ПКР), то їх перерахування на 20 балів здійснюється так: <math>ПК = 4.1 * 20 / 5 = 4.1 * 4 = 16.4 // 16</math> (балів). За періодичний контроль (ПКР) студентом отримано 30 балів. Тоді за контрольну точку (КТ) буде отримано <math>КТ = ПК + ПКР = 16 + 30 = 46</math> (балів).</p> <p>Студент має право на підвищення результату тільки одного періодичного контролю (ПКР) протягом двох тижнів після його складання у випадку отримання незадовільної оцінки. Якщо підсумковим контролем вивчення дисципліни є диференційований або недиференційований залік, то набраних таким чином 60 і більше балів достатньо для його зарахування.</p> <p>Якщо підсумковим контролем є екзамен, на його складання надається 100 балів за виконання тестів (або задач чи завдань іншого виду). Загальний рейтинг з дисципліни (ЗР) складається з суми балів (Е), отриманих на екзамені, і підсумкової оцінки (ПО) та ділиться навпіл. <math>ЗР = (ПО + Е) / 2</math></p>
<p><b>Практичні заняття</b></p>	<p>«5» – студент в повному обсязі володіє навчальним матеріалом, вільно самостійно та аргументовано його викладає під час усних виступів та письмових відповідей, глибоко та всебічно розкриває зміст теоретичних питань та практичних завдань, використовуючи при цьому нормативну, обов'язкову та додаткову літературу. Правильно вирішив усі розрахункові / тестові завдання. Здатен виділяти суттєві ознаки вивченого за допомогою операцій синтезу, аналізу, виявляти причинно-наслідкові зв'язки, формувати висновки і узагальнення, вільно оперувати фактами та відомостями.</p> <p>«4» – студент достатньо повно володіє навчальним матеріалом, обґрунтовано його викладає під час усних виступів та письмових відповідей, в основному розкриває зміст теоретичних питань та практичних завдань, використовуючи при цьому нормативну та обов'язкову літературу. Але при викладанні деяких питань не вистачає достатньої глибини та аргументації, допускаються при цьому окремі несуттєві неточності та незначні помилки. Правильно вирішив більшість розрахункових / тестових завдань. Студент здатен виділяти суттєві ознаки вивченого за допомогою операцій синтезу, аналізу, виявляти причинно-наслідкові зв'язки, у яких можуть бути окремі несуттєві помилки, формувати висновки і узагальнення, вільно оперувати фактами та відомостями.</p> <p>«3» – студент в цілому володіє навчальним матеріалом, викладає його основний зміст під час усних виступів та письмових розрахунків, але без глибокого всебічного аналізу, обґрунтування та аргументації, допускаючи при цьому окремі суттєві неточності та помилки. Правильно вирішив половину розрахункових / тестових завдань. Має ускладнення під час виділення суттєвих ознак вивченого; під час виявлення причинно-наслідкових зв'язків і формулювання висновків.</p> <p>«2» – студент не в повному обсязі володіє навчальним матеріалом. Фрагментарно, поверхово (без аргументації та обґрунтування) викладає його під час усних виступів та письмових розрахунків, недостатньо розкриває зміст теоретичних питань та практичних завдань, допускаючи при цьому суттєві неточності. Правильно вирішив окремі розрахункові / тестові завдання. Безсистемно відділяє випадкові ознаки вивченого; не вміє зробити найпростіші операції аналізу і синтезу; робити узагальнення, висновки.</p>

	<i>«І» – студент виконав менше половини завдання практичної роботи або не виконав зовсім; під час усних відповідей не розкриває зміст теоретичних питань та практичних завдань. Не відповідає на елементарні питання.</i>
<b>Періодичний контроль знань і вмінь студентів</b>	<b>60 балів</b> <i>За кожний ПМК максимум 30 балів: 30 тестових питань, 1 питання оцінюється в 1 бал (за принципом вірна відповідь – 1 бал, не вірна – 0).</i>
<b>Умови допуску до підсумкової контролю</b>	<i>Студент, який навчається стабільно на «відмінні» оцінки і саме такі оцінки має за періодичні контролю, накопичує впродовж вивчення навчального курсу 90 і більше балів, має право не складати екзамен з даної дисципліни. Студент зобов'язаний відпрацювати всі пропущені семінарські заняття протягом двох тижнів. Невідпрацьовані заняття (невиконання навчального плану) є підставою для недопущення студента до підсумкового контролю.</i>

### Шкала оцінювання: національна та ECTS

Сума балів за всі види навчальної діяльності	Оцінка ECTS	Оцінка за національною шкалою	
		для екзамену, курсового проекту (роботи), практики	для заліку
90 – 100	<b>A</b>	відмінно	зараховано
82-89	<b>B</b>	добре	
74-81	<b>C</b>		
64-73	<b>D</b>	задовільно	
60-63	<b>E</b>		
35-59	<b>FX</b>	незадовільно з можливістю повторного складання	не зараховано з можливістю повторного складання
0-34	<b>F</b>	незадовільно з обов'язковим повторним вивченням дисципліни	не зараховано з обов'язковим повторним вивченням дисципліни

## 9. РЕКОМЕНДОВАНА ЛІТЕРАТУРА ТА ІНФОРМАЦІЙНІ РЕСУРСИ

### ОСНОВНА ЛІТЕРАТУРА

1. Мельник Р.А. Програмування веб-застосувань (фронт-енд та бек-енд). Львів: Львівська політехніка, 2018. 248 с.
2. Сьомка Н.К., Філіппова Л.Л. Практикум з WEB-програмування: навч. посіб. Ірпінь : Університет ДФС України, 2016. 352 с.
3. Трофименко О.Г., Козін О.Б., Задерейко О.В., Плачінда О.Є. Веб-технології та веб-дизайн: навч. посібник. Одеса: Фенікс, 2019. 284 с.
4. Elliott E. Programming JavaScript Applications. Robust Web Architecture with Node, HTML5, and Modern JS Libraries. – O'Reilly Media, 2014.

### Допоміжна

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

3. Пасічник О.Г., Пасічник О.В., Стеценко І.В. Основи веб-дизайну: [Навч. посіб.]. Київ: Вид. група BHV, 2009. 336 с.
4. Трегубенко І.Б., Олійник Г.Т., Панаско О.М. Сучасні технології програмування в мережах: навч. посіб. Черкаси: ЧДТУ, 2010. 175 с.

#### **14. Інформаційні ресурси**

1. Elliott, Eric, 2014. Programming JavaScript Applications. [Електронний ресурс]. – Режим доступу: <http://chimera.labs.oreilly.com/books/1234000000262/index.html>.
2. JavaScript. Codecademy. [Електронний ресурс]. – Режим доступу: <https://www.codecademy.com/learn/javascript>.
3. JavaScript Allongé (ES5). Leanpub. [Електронний ресурс]. – Режим доступу: <https://leanpub.com/javascript-allonge/read#leanpub-auto-a-pull-of-the-lever-prefaces>.
4. JavaScript For Cats. An introduction for new programmers. Available at: <http://jsforcats.com/>.
5. JavaScript-Garden. [Електронний ресурс]. – Режим доступу: <http://bonsaiden.github.io/JavaScript-Garden/>.
6. Haverbeke, Marijn, 2014. Eloquent JavaScript. A Modern Introduction to Programming. [Електронний ресурс]. – Режим доступу: <http://eloquentjavascript.net>
7. Osmani, Addy, 2015. Learning JavaScript Design Patterns. [Електронний ресурс]. – Режим доступу: <https://addyosmani.com/resources/essentialjsdesignpatterns/book/#revealingmodulepatternjavascript>.
8. Tutorials Point, 2015. JavaScript. Tutorials Point. [Електронний ресурс]. – Режим доступу: [http://www.tutorialspoint.com/javascript/javascript\\_tutorial.pdf](http://www.tutorialspoint.com/javascript/javascript_tutorial.pdf).