

**Чернівецький національний університет імені Юрія Федьковича**  
(повне найменування закладу вищої освіти)

**Факультет математики та інформатики**  
(назва факультету/навчально-наукового інституту)

**Кафедра математичного моделювання**  
(назва кафедри, що забезпечує викладання)

**“ЗАТВЕРДЖУЮ”**

**Декан факультету  
математики та інформатики**

**Ольга МАРТИНЮК**

**2025 року**



**РОБОЧА ПРОГРАМА  
навчальної дисципліни**

**Web-технології та web-програмування**

(назва навчальної дисципліни)

**вибіркова**

(вказати: обов'язкова)

**Освітньо-професійна програма «Системний аналіз»**

(назва програми)

**Спеціальність 124 Системний аналіз**

(вказати: код, назва)

**Галузь знань 12 Інформаційні технології**

(вказати: шифр, назва)

**Рівень вищої освіти перший (бакалаврський)**

(вказати: перший (бакалаврський) / другий (магістерський) / третій (освітньо-науковий))

**Факультет математики та інформатики**

(назва факультету/ навчально-наукового інституту, на якому здійснюється підготовка фахівців за вказаною освітньою програмою)

**Мова навчання українська**

(вказати: на якій мові читається дисципліна)

**Чернівці 2025 рік**

Робоча програма навчальної дисципліни «*Web-технології та web-програмування*» складена відповідно до освітньо-професійної програми «Системний аналіз»

**Розробник:**

Готинчан Тетяна Іванівна, доцент кафедри математичного моделювання, кандидат фіз.-мат. наук, доцент

**Викладачі**, що забезпечують читання даної навчальної дисципліни:

Готинчан Тетяна Іванівна, доцент кафедри математичного моделювання, кандидат фіз.-мат. наук, доцент;

Мельник Василь Сергійович, кандидат фіз.-мат. наук, асистент

**Затверджено** на засіданні кафедри математичного моделювання

*Протокол № 15 від «24» червня 2025 року*

Завідувач кафедри \_\_\_\_\_

 **Ігор ЧЕРЕВКО**

**Схвалено** методичною радою факультету математики та інформатики

*Протокол № 12 від «25» червня 2025 року*

Голова методичної ради \_\_\_\_\_

 **Віра СІКОРА**

**Мета навчальної дисципліни:** Метою вивчення дисципліни є опанування майбутніми фахівцям теоретичних знань і практичних навичок, необхідних для вирішення питань, пов'язаних із проєктуванням та розробкою вебсторінок у глобальній мережі інтернет з використанням сучасних інструментальних засобів.

Курс для фронтенд розробки. Програма курсу передбачає навчити студентів вільно підбирати необхідні технології для створення інтерактивних компонент вебсторінок. Причому поєднувати технології як HTML і CSS, так і JS з відповідними бібліотеками для підвищення продуктивності розробки вебсторінок.

Вивчення мови програмування JS значно прискорює вивчення особливості роботи з фреймворками та бібліотеками, що використовуються на фронтенді.

**Пререквізити:** Основи інтернет-технологій, Програмування

**Результати навчання:**

**знати:** основні аспекти мови сценарію JS, елементи об'єктної моделі документа (DOM), об'єктної моделі браузера (BOM) та методи роботи з ними; асинхронний JS; сучасні бібліотеки JS роботи з датами, графікою, анімацією, асинхронними операціями.

**вміти:** використовувати сучасні інструментальні засоби для створення інтерактивності вебсторінок; аналізувати поставлену задачу, підбирати й застосовувати сучасні технології при створенні вебсторінок відповідно до вимог замовника; модифікувати та розробляти модулі й компоненти з метою підвищення продуктивності роботи вебсторінок.

Дисципліна формує такі компетентності за ОП:

**Загальні компетентності:**

ЗК01. Здатність до абстрактного мислення, аналізу та синтезу.

ЗК02. Здатність застосовувати знання у практичних ситуаціях.

ЗК04. Знання та розуміння предметної області та розуміння професійної діяльності

ЗК07. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

ЗК10. Здатність працювати автономно.

ЗК11. Здатність генерувати нові ідеї (креативність).

**Фахові компетентності спеціальності:**

ФК1. Здатність використовувати системний аналіз як сучасну міждисциплінарну методологію, що базується на прикладних математичних методах та сучасних інформаційних технологіях і орієнтована на вирішення задач аналізу і синтезу технічних, економічних, соціальних, екологічних та інших складних систем.

Наведені результати навчання за відповідною дисципліною співвідносяться із такими **програмними результатами навчання:**

ПР8. Володіти сучасними методами розробки програм і програмних комплексів та прийняття оптимальних рішень щодо складу програмного забезпечення, алгоритмів процедур і операцій.

ПР13. Проєктувати, реалізовувати, тестувати, впроваджувати, супроводжувати, експлуатувати програмні засоби роботи з даними і знаннями в комп'ютерних системах і мережах.

## Опис навчальної дисципліни

### Загальна інформація

| Форма навчання | Рік підготовки | Семестр | Кількість |       | Кількість годин |           |             |             |                   |                        | Вид підсумкового контролю |       |
|----------------|----------------|---------|-----------|-------|-----------------|-----------|-------------|-------------|-------------------|------------------------|---------------------------|-------|
|                |                |         | кредитів  | годин | лекції          | практичні | семінарські | лабораторні | самостійна робота | індивідуальні завдання |                           |       |
| Денна          | 2              | 2       | 3         | 90    | 30              |           |             |             | 30                | 30                     |                           | залік |

### Структура змісту навчальної дисципліни

| Назви змістових модулів і тем навчальних занять                 | Кількість годин |              |   |           |     |           |        |              |    |     |     |      |  |
|---|-----------------|--------------|---|-----------|-----|-----------|--------|--------------|----|-----|-----|------|--|
|   | денна форма     |              |   |           |     |           |        | заочна форма |    |     |     |      |  |
|   | усього          | у тому числі |   |           |     |           | усього | у тому числі |    |     |     |      |  |
|   |                 | л            | п | лаб       | інд | с.р.      |        | л            | п  | лаб | інд | с.р. |  |
| 1   | 2               | 3            | 4 | 5         | 6   | 7         | 8      | 9            | 10 | 11  | 12  | 13   |  |
| <b>Змістовий модуль 1. JS.</b>                                  |                 |              |   |           |     |           |        |              |    |     |     |      |  |
| Тема 1 <i>JS. Загальна характеристика. Типізація. Оператори</i> | 13              | 4            |   | 4         |     | 5         |        |              |    |     |     |      |  |
| Тема 2. <i>Об'єкти. Масиви та колекції. Функції.</i>            | 15              | 6            |   | 4         |     | 5         |        |              |    |     |     |      |  |
| Тема 3. <i>Події. DOM. BOM</i>                                  | 19              | 6            |   | 8         |     | 5         |        |              |    |     |     |      |  |
| Разом за ЗМ1  | <b>47</b>       | <b>16</b>    |   | <b>16</b> |     | <b>15</b> |        |              |    |     |     |      |  |
| <b>Змістовий модуль 2. Бібліотеки. Асинхронний JS</b>           |                 |              |   |           |     |           |        |              |    |     |     |      |  |
| Тема 1. <i>Бібліотеки</i>                                       | 22              | 8            |   | 6         |     | 8         |        |              |    |     |     |      |  |
| Тема 2. <i>Асинхронний JS</i>                                   | 21              | 6            |   | 8         |     | 7         |        |              |    |     |     |      |  |
| Разом за ЗМ 2   | <b>43</b>       | <b>14</b>    |   | <b>14</b> |     | <b>15</b> |        |              |    |     |     |      |  |
| <b>Усього годин</b>   | <b>90</b>       | <b>30</b>    |   | <b>30</b> |     | <b>30</b> |        |              |    |     |     |      |  |

### Тематика лекційних занять з переліком питань

| № | Назва теми (завдання)   |
|---|---|
| 1 | <b>Вступ до JavaScript. Загальна характеристика мови:</b> <ul style="list-style-type: none"> <li>● призначення JS, особливості як мови сценаріїв;</li> <li>● середовища виконання (браузер, Node.js)</li> </ul> |

|    |   |
|----|---|
| 2  | <b>Типізація у JavaScript</b> <ul style="list-style-type: none"> <li>• Примітивні типи даних;</li> <li>• приведення типів;</li> <li>• перевірка типів;</li> <li>• динамічна природа JS.</li> </ul>  |
| 3  | <b>Оператори та вирази</b> <ul style="list-style-type: none"> <li>• Арифметичні, логічні, умовні оператори;</li> <li>• оператори порівняння;</li> <li>• шаблонні рядки.</li> </ul>  |
| 4  | <b>Об'єкти у JavaScript</b> <ul style="list-style-type: none"> <li>• Створення об'єктів;</li> <li>• властивості й методи;</li> <li>• вбудовані об'єкти (Date, Math тощо).</li> </ul>  |
| 5  | <b>Масиви, колекції Set і Map</b> <ul style="list-style-type: none"> <li>• Масиви та методи роботи з ними (map, filter, reduce);</li> <li>• відмінності між масивами та об'єктами;</li> <li>• множини (Set) і відображення (Map) – особливості та застосування.</li> </ul>  |
| 6  | <b>Функції та області видимості</b> <ul style="list-style-type: none"> <li>• Оголошення та виклики;</li> <li>• параметри;</li> <li>• замикання;</li> <li>• стрілкові функції;</li> <li>• контекст this.</li> </ul>  |
| 7  | <b>Події у JavaScript</b> <ul style="list-style-type: none"> <li>• Обробка подій;</li> <li>• моделі подій;</li> <li>• делегування;</li> <li>• приклади взаємодії користувача з елементами.</li> </ul>   |
| 8  | <b>Document Object Model (DOM)</b> <ul style="list-style-type: none"> <li>• Структура DOM;</li> <li>• вибір і зміна елементів;</li> <li>• створення й видалення вузлів;</li> <li>• робота з класами та атрибутами.</li> </ul>   |
| 9  | <b>Browser Object Model (BOM)</b> <ul style="list-style-type: none"> <li>• Об'єкт window;</li> <li>• взаємодія з історією браузера,</li> <li>• таймерами,</li> <li>• діалоговими вікнами,</li> <li>• робота зі сховищем.</li> </ul>   |
| 10 | <b>Бібліотеки JavaScript</b> <ul style="list-style-type: none"> <li>• jQuery як приклад старих бібліотек;</li> <li>• бібліотеки для роботи із графікою, датами, анімацією,</li> <li>• Lodash; огляд сучасних фреймворків (React, Vue) у контексті JS-екосистеми.</li> </ul> |

|    |   |
|----|---|
| 11 | <b>Асинхронний JavaScript</b> <ul style="list-style-type: none"> <li>• Callbacks, Promises, async/await;</li> <li>• робота з API та мережевими запитами (fetch).</li> </ul> |
|----|---|

### Тематика лабораторних занять з переліком питань

| № | Назва теми (завдання)   |
|---|---|
| 1 | <b>Основи використання JS</b> <ol style="list-style-type: none"> <li>1. Засобами HTML і CSS створити веб сторінку, на якій розмістити: <ul style="list-style-type: none"> <li>• навігайну панель з переліком завдань;</li> <li>• блоки реалізації завдань;</li> </ul> </li> <li>2. Блоки завдань зв'язати з відповідними назвами навігаційної панелі. Навігаційна панель має бути закріпленою.</li> <li>3. На навігаційній панелі має бути лого, при наведенні на яке з'являється вікно з фото і короткою інформацією про автора сторінки.</li> <li>4. Стилiзувати сторiнку за власним уподобанням.</li> <li>5. Виберiть з перелiку завдань принаймнi 1 для реалiзацiї.</li> <li>6. Самостiйно запропонуйте i реалiзуйте ще 2 завдання на iншi теми. Тематика не обмежується.</li> <li>7. Передбачити контроль введення вхiдних даних.</li> <li>8. Оформiть лабораторну роботу.</li> <li>9. Захистiть лабораторну роботу до зазначеного строку, вказаного на Moodle.</li> <li>10. Завантажте архiв роботи на Moodle. Можна размiщувати саму лабораторну роботу на GitHub.com чи на власному Google диску, а у текстовому звiтi надавати посилання на неї.</li> </ol>  |
| 2 | <b>Робота з DOM і BOM</b> <ol style="list-style-type: none"> <li>1. Продумайте тематику і макети свого проекту. Використовуйте інструменти чутливого дизайну для вебсторінок..</li> <li>2. При завантаженні сторінки мають завантажуватись лише низка зображень із описом. Опис прихований. Стає доступним, якщо його розгорнути. Дані мають братись із масиву, елементами якого є об'єкти.</li> <li>3. Додатково*. Можливе передбачення поступового завантаження низки елементів за допомогою кнопок.</li> <li>4. Стилiзацiя iнформацiї має вiдповiдати картковому дизайну</li> <li>5. Додатково*. На зображеннях можуть бути размiщеннi додатковi елементи, наприклад розмiри знижок по серединi чи занадто великi тощо. При натисненнi або наведеннi на зображеннi, вони плавно зменшуються, що дає змогу краще розглядити саме зображення.</li> <li>6. Пiсля повного завантаження сторiнки за деякий час має з'являтися немодальне вiкно займаючи частину екрана з пропозицiєю про пiдписку повiдомлень з вашого сайту з двома кнопками прийняття пропозицiї i її вiдхилення. У разi прийняття при повторному завантаженнi сторiнки це повiдомлення не з'являтиметься. При натисненнi кожної iз двох кнопок вiкно ховається. Можливе короткочасне повiдомлення про подяку за приєднання.</li> </ol> |

|   |   |
|---|---|
|   | <p>7. Допишіть дію кнопки погодження підписки так, щоб при повторному завантаженні сторінки це вікно не з'являлось.</p> <p>8. Додайте інтерактивну кнопку навігації вгору. Кнопка має з'являтися, коли сторінка прокручена на 2/3 від висоти браузера.</p> <p>9. Передбачте в певному місці появу модального вікна з рекламою. На модальному вікні кнопка його закриття спочатку не активна. Активною вона стає за певний часовий інтервал.</p> <p>10. Оформіть лабораторну роботу.</p> <p>11. Захистіть лабораторну роботу до зазначеного строку, вказаного на Moodle.</p> <p>12. Завантажте архів роботи на Moodle. Можна розміщувати саму лабораторну роботу на GitHub.com чи на власному Google диску, а у текстовому звіті надавати посилання на неї.</p>  |
| 3 | <p><b>Робота з подіями. Використання бібліотек. Графіка</b></p> <p>1. У вікні формування кошика користувач повинен мати змогу запропонований товар, що поданий у картковому дизайні, сортувати за критеріями:</p> <ol style="list-style-type: none"> <li>2. назвою;</li> <li>3. ціною (у порядку зростання чи зростання).</li> </ol> <p>А також вибирати товар у певному ціновому діапазоні.<br/>Додатково*. Можна передбачити одночасний відбір за декількома критеріями.</p> <p>4. Кожен товар можна додати до кошика. На ваш вибір продумайте як оформити при цьому зазначення кількості вибраного товару. Ціна товару не коригується.<br/>Додатково*. Реалізувати додавання товару до кошика перетягуванням його до піктограми кошик. При цьому з'являється модальне вікно, де за замовчуванням кількість 1. Проте її можна корегувати.</p> <p>5. У вікні кошика має бути змога коригування кількості, видалення найменування та за потреби повернутись і додати нове найменування. А також виводитись остаточна вартість кошика.<br/>Додатково*. Передбачити перевірку чи таке найменування вже введено з метою не дублювання найменувань.</p> <p>6. На сторінці новин має бути стовпчик з із заголовками останніх новин, які завантажуються при відкритті цієї сторінки. Новини мали б оновлюватись по мірі їх надходжень. Розташування новин має бути у хронологічному порядку за датою (формат на вибір) і часом (год і хв). Важливі новини виділяться жирним шрифтом. При натисканні на новину вона розгортається у центральній частині. На мобільних пристроях новина одразу розгортається, а решта посуваються.</p> <p>7. На підставі даних, отриманих про вибір товарів побудувати принаймні три графічні об'єкти: кругову діаграму, гістограму (стовпчикову діаграму), графік, які б демонстрували популярність товарів у розрізах: тип (назва), кількість (абсолютні чи відносні значення).</p> <p>8. Дані мають бути масштабовані і змінюватись при змінні даних вибору.</p> <p>9. Користувач вибирає, який вид графічного о'єкта виводити. Решта приховані.</p> <p>10. Оформіть лабораторну роботу.</p> <p>11. Захистіть лабораторну роботу до зазначеного строку, вказаного на Moodle.</p> |

|   |  |
|---|--|
|   | <p>12. Завантажте архів роботи на Moodle. Можна розміщувати саму лабораторну роботу на GitHub.com чи на власному Google диску, а у текстовому звіті надавати посилання на неї.</p>   |
| 4 | <p><b>Асинхронний JS</b></p> <ol style="list-style-type: none"> <li>1. Оформити валідацію форм авторизації та реєстрації. Перед відправкою на сервер даних має пройти повна валідація на наявність заповнення обов'язкових даних та правильність заповнення їх. Наприклад, пароль має мати не менше певної кількості символів. Користувач має розуміти заповнення яких полів обов'язкове.<br/>Додатково*. Можна передбачити щоб пароль мав символи з набору обов'язкових символів.</li> <li>2. Пропишіть перевірку на збіг введеного паролю та його повторення.</li> <li>3. У разі не проходження валідації на етапі реєстрації користувач має бачити які саме неточності при заповненні даних він допустив.</li> <li>4. У разі не проходження валідації на етапі авторизації користувач, навпаки, не має бачити які саме неточності при заповненні даних він допустив. Лише з'являється інформаційне повідомлення про не проходження авторизації.</li> <li>5. В обох випадках у разі проходження валідації має з'являтися інформаційне повідомлення про її успіх.</li> <li>6. Замовлення з кошика може відправляти лише зареєстрований користувач. Допишіть повідомлення у разі не авторизації його і перенаправлення на авторизацію/реєстрацію.</li> <li>7. Установити локальний сервер.</li> <li>8. Дані товарів, які завантажуються на сторінку, подати у JSON форматі. Надалі використовувати їх, розпарсивши у подальшому.</li> <li>9. Додаткове завдання*. Доповнити кожен елемент товару/пропозиції додатковою властивістю популярності для фіксації кількості їх вибору користувачами. За даними їхньої популярності відобразити для адміністратора діаграми.</li> <li>10. Аналогічно й дані користувачів і адміністратора. Дані адміністратора є з самого початку. Він не реєструється. Продумайте як розрізнити звичайного користувача й адміністратора.</li> <li>11. Редагувати, видаляти, додавати нові товари/пропозиції може лише адміністратор. Користувач може лише переглядати їх.</li> <li>12. Створіть відповідні запити для CRUD операцій для:</li> <li>13. реєстрації/авторизації;</li> <li>14. маніпулювання даними товарів/пропозицій.</li> <li>15. Додаткове завдання*. Створити імітацію особистого кабінету користувача, де він бачить свої особисті дані. Може змінювати пароль і контакту інформацію і можливо бачить зроблені попередні замовлення.</li> <li>16. Оформіть лабораторну роботу.</li> <li>17. Завантажте архів роботи на Moodle. Можна розміщувати саму лабораторну роботу на GitHub.com чи на власному Google диску, а у текстовому звіті надавати посилання на неї.</li> <li>18. Зареєструйтесь на хостингу за уподобанням, наприклад, на netlify.com</li> <li>19. Розмістіть та ресурсі свій проєкт. Продемонструйте роботу викладачу.</li> <li>20. Захистіть лабораторну роботу та розмістіть на сторінці курсу архів проєкту та посилання на хостинг.</li> </ol> |

## Завдання для самостійної роботи студентів

| № | Назва теми   | Завдання для самостійної роботи   | К-сть год. |
|---|--|---|------------|
| 1 | <i>JS. Загальна характеристика. Типізація. Оператори</i> | Особливості типізації та приведення типів   | 1          |
| 2 | <i>JS. Загальна характеристика. Типізація. Оператори</i> | Оператори   | 2          |
| 3 | <i>JS. Загальна характеристика. Типізація. Оператори</i> | Регулярні вирази  | 2          |
| 4 | <i>Об'єкти. Масиви та колекції. Функції.</i>             | Особливості використання функцій.   | 2          |
| 5 | <i>Об'єкти. Масиви та колекції. Функції.</i>             | JSON<br>Відмінності між Map і об'єктами   | 3          |
| 6 | <i>Події. DOM. BOM</i>                                   | Делегування подій   | 5          |
| 7 | <i>Бібліотеки</i>  | Бібліотеки роботи з 3D графікою Three.js, та анімацією Anime.js, GSAP   | 8          |
| 8 | <i>Асинхронний JS</i>                                    | Бібліотеки JQuery, JQuery UI, Lodash (старіші), бібліотеки для роботи із анімацією Three.js,<br>Бібліотеки для асинхронних запитів Axios, Ку. | 7          |

### Методи навчання

Під час викладання навчальної дисципліни використовуються такі методи навчання:

- за джерелом передачі та сприймання студентами навчальної інформації:
  - словесні (лекція, пояснення, бесіда, інструктаж);
  - наочні (ілюстрація, демонстрація);
  - практичні (вправи, проєкти);
- за логікою передачі та сприймання студентами навчальної інформації:
  - інформаційно-рецептивний;
  - репродуктивний;
  - проблемний;
  - частково-пошуковий (евристичний);
- за стимулюванням навчально-пізнавальної діяльності:
  - методи стимулювання пізнавальних потреб та інтересів;
  - методи стимулювання обов'язку та відповідальності.

### Система контролю та оцінювання

**Методи контролю:**

- усне опитування (індивідуальне, фронтальне, групове).
- письмовий контроль (тести);
- практичні методи (виконання лабораторних);
- комп'ютерне тестування (електронні тести).

### Форми контролю:

- поточний контроль – перевірка знань у процесі навчання (усні відповіді, захист захист лабораторних робіт).
- підсумковий контроль – перевірка після завершення навчального курсу (залік).

### Критерії оцінювання поточного та підсумкового контролю

Система оцінювання рівня навчальних досягнень ґрунтується на принципах ECTS і є накопичувальною. Оцінювання знань здійснюється за 100-бальною шкалою. Результати роботи впродовж навчального семестру оцінюються під час поточного та модульного контролю в діапазоні загалом від 0 до 60 балів, а результати підсумкового контролю (заліку) – від 0 до 40 балів.

Протягом семестру студенти виконують 4 лабораторні роботи, кожна з яких є продовженням попередньої. У результаті кожен студент виконує індивідуальний проєкт. Лабораторні роботи оцінюються відповідно балами: 5, 15, 20 і 20 за повне виконання завдань.

Виконання лабораторних передбачає самостійного опрацювання додаткових інформаційних джерел і домашнього доопрацювання над завданнями, розпочатими в аудиторії. У випадку неістотних помилок при виконанні завдань знімається 1-2 бали, а істотних, необґрунтованого застосування методів чи невиконання завдань – 3-5 балів. У разі проходження сертифікованих курсів на навчальних платформах і вчасного подання сертифікатів можливе зарахування сертифікату замість лабораторної роботи з відповідної теми. Додатково можна отримати до 10 балів за виконання додаткових завдань.

Підсумковим контролем з дисципліни є залік у вигляді тестування у системі moodle. Варіант тесту містить 40 питань з однією правильною відповіддю, кожне з яких оцінюється в 1 бал.

Підсумкова оцінка виставляється за результатами суми балів набраних на змістовних модулях під час семестру та підсумковому модулі (заліку).

### Розподіл балів, які отримують студенти

| Поточне оцінювання (аудиторна та самостійна робота) |    |    |                     |    | Кількість балів (залік) | Сумарна к-ть балів |
|---|----|----|---------------------|----|-------------------------|--------------------|
| Змістовий модуль №1                                 |    |    | Змістовий модуль №2 |    |                         |                    |
| T1  | T2 | T3 | T1                  | T2 | 40                      | 100                |
| 10  | 10 | 10 | 15                  | 15 |                         |                    |

T1, T2 ... – теми змістових модулів.

### Шкала оцінювання: національна та ЄКТС

| 100-бальна шкала | Оцінка за національною шкалою | Оцінка за шкалою ЄКТС |                                |            |
|------------------|-------------------------------|-----------------------|--------------------------------|------------|
|                  |                               | Оцінка                | Пояснення за розширеною шкалою |            |
| 90-100           | Зараховано                    | Відмінно              | А                              | відмінно   |
| 80-89            |                               | Добре                 | В                              | дуже добре |
| 70-79            |                               | Задовільно            | С                              | добре      |
| 60-69            |                               | Задовільно            | Д                              | задовільно |
| 50-59            |                               | Задовільно            | Е                              | достатньо  |

|       |              |              |    |   |
|-------|--------------|--------------|----|---|
| 35-49 |              |              | FX | (незадовільно) з<br>можливістю<br>повторного<br>складання   |
| 1-34  | Незараховано | Незадовільно | F  | (незадовільно) з<br>обов'язковим<br>самостійним<br>повторним<br>опрацюванням<br>освітнього компонента<br>до перескладання |

**Перелік питань для самоконтролю та підсумкового контролю навчальних досягнень студентів**

**Загальна характеристика, типізація, оператори**

1. Які особливості мови JavaScript та сфери її застосування?
2. Які існують примітивні типи даних у JavaScript?
3. Що таке динамічна типізація та які її переваги і недоліки?
4. Які вирази та оператори є основними у JavaScript?
5. У чому відмінність між операторами == та ===?

**Об'єкти, масиви, Set, Map**

6. Як створюються об'єкти у JavaScript?
7. Які вбудовані об'єкти існують у мові?
8. Що таке масив і які методи роботи з масивами ви знаєте?
9. У чому відмінність між масивом і об'єктом?
10. Що таке методи map(), filter(), reduce() і як вони застосовуються?
11. Що таке колекція Set і які її особливості?
12. Для чого використовується структура Map у JS?
13. Які відмінності між Map і об'єктами?

**Функції**

14. Які способи оголошення функцій у JavaScript ви знаєте?
15. Що таке стрілкові функції і чим вони відрізняються від звичайних?
16. Що таке замикання і як воно працює?
17. Яке значення має ключове слово this?

**Події, DOM, BOM**

18. Як створюється обробник подій у JavaScript?
19. Що таке делегування подій і коли воно корисне?
20. Що таке DOM та які його основні можливості?
21. Як за допомогою JS змінювати вміст і атрибути HTML-елементів?
22. Що таке BOM і які об'єкти він включає?
23. Як працюють методи setTimeout() і setInterval()?

**Асинхронний JavaScript**

24. Що таке асинхронність у JavaScript і навіщо вона потрібна?
25. Як працюють колбеки (callbacks)?
26. Що таке проміси (Promise) і як їх використовувати?
27. У чому перевага async/await над промісами?

28. Як виконується асинхронний HTTP-запит у JavaScript (метод fetch)?

### Бібліотеки JavaScript

29. Які бібліотеки JavaScript ви знаєте та для чого вони використовуються?

30. Чи є зараз переваги використання бібліотек (наприклад, jQuery, Lodash) у порівнянні з “чистим” JavaScript?

### Зарахування результатів неформальної освіти

Здобувачі вищої освіти має право на участь у неформальній/інформальній освіті.

У межах поточного контролю можуть визнаватися результати неформальної/інформальної освіти за умови наявності сертифікату або освітньої декларації про результати неформальної /інформальної освіти з питань, що відповідає тематиці курсу («Порядок визнання у Чернівецькому національному університеті імені Юрія Федьковича результатів навчання, здобутих шляхом неформальної та/або інформальної освіти», <https://www.chnu.edu.ua/media/4g5fzssb/poriadok-vyznannia-rezultativ-navchannia-zdobutykh-shliakhom-neformalnoi-ta-abo-informalnoi-osvity.pdf> ).

Студентам можуть бути зараховані додаткові бали, отримані через неформальну освіту, до загальної суми балів, набраної з освітньої компоненти, за умови, що результати з проблеми, за якою відбувалося навчання, відповідають тематиці курсу та отримані на ліцензійних платформах.

### Рекомендована література

#### Основна

1. The Modern JavaScript Tutorial. [Електронний ресурс] – Режим доступу до ресурсу: <https://javascript.info/>
2. JavaScript. [Електронний ресурс] – Режим доступу до ресурсу: <https://www.w3schools.com/js/default.asp>
3. Онлайн-підручник з web-сервісів. [Електронний ресурс] – Режим доступу до ресурсу: <http://www.w3schools.com/webservices/>
4. JS підручник. [Електронний ресурс] – Режим доступу до ресурсу: <https://w3schoolsua.github.io/html/index.html>
5. David Flanagan. JavaScript: The Definitive Guide, 7th Edition by David Flanagan Released May 2020 Publisher(s): O'Reilly Media, Inc. ISBN: 9781491952023. [Електронний ресурс] – Режим доступу до ресурсу: <https://www.oreilly.com/library/view/javascript-the-definitive/9781491952016/>
6. JavaScript. Notes for Professionals. [Електронний ресурс] – Режим доступу до ресурсу: <https://books.goalkicker.com/JavaScriptBook/>

#### Додаткова

1. [John Dean](#). Web Programming with HTML5, CSS, and JavaScript. [Електронний ресурс] – Режим доступу до ресурсу: <https://www.pdfdrive.com/web-programming-with-html5-css-and-javascript-e187657772.html>

2. Refactoring: Improving the Design of Existing Code by Martin Fowler, Released November 2018, Publisher(s): Addison-Wesley Professional, ISBN: 9780134757681. [Електронний ресурс] – Режим доступу до ресурсу: <https://www.oreilly.com/library/view/refactoring-improving-the/9780134757681/>
3. JQuery. [Електронний ресурс] – Режим доступу до ресурсу: <https://jquery.com/>
4. JQuery підручник. [Електронний ресурс] – Режим доступу до ресурсу: <https://www.w3schools.com/jquery/default.asp>
5. JQuery UI. [Електронний ресурс] – Режим доступу до ресурсу: <https://jqueryui.com/>

### Інформаційні ресурси

1. Основи, інструменти, оновлення та приклади з web-розробників Google [Електронний ресурс] – Режим доступу до ресурсу: <https://www.html5rocks.com/en/>
2. Сайт розробників GOOGLE. [Електронний ресурс] – Режим доступу до ресурсу: <https://www.developers.google.com>
3. Посилання на електронний курс. [Електронний ресурс] – Режим доступу до ресурсу: <https://moodle.chnu.edu.ua/course/view.php?id=3009>

### Політика академічної доброчесності

Дотримання політики щодо академічної доброчесності учасниками освітнього процесу при вивченні навчальної дисципліни регламентовано такими документами:

✓ «Етичний кодекс Чернівецького національного університету імені Юрія Федьковича» [https://www.chnu.edu.ua/media/jxdfs0zb/etychnyi-kodeks-chernivets\\_kohnatsionalnoho-universytetu.pdf](https://www.chnu.edu.ua/media/jxdfs0zb/etychnyi-kodeks-chernivets_kohnatsionalnoho-universytetu.pdf)

✓ «Положенням про виявлення та запобігання академічного плагіату у Чернівецькому національному університету імені Юрія Федьковича» [https://www.chnu.edu.ua/media/n5nbzwgb/polozhennia-chnu-pro-plahi\\_at-2023plusdodatky-31102023.pdf](https://www.chnu.edu.ua/media/n5nbzwgb/polozhennia-chnu-pro-plahi_at-2023plusdodatky-31102023.pdf)