



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ  
УНІВЕРСИТЕТ ІМЕНІ ІВАНА ПУЛЮЯ

Кафедра приладів і контрольно-вимірювальних системи

**МЕТОДИЧНІ ВКАЗІВКИ  
для самостійної роботи студентів**

**з дисципліни**

**ОСНОВИ ІНФОРМАЦІЙНИХ  
СИСТЕМ**

для студентів спеціальності  
*176 – «Мікро- та наносистемна техніка»*

ТЕРНОПІЛЬ 2023

Методичні вказівки щодо самостійної роботи студентів з дисципліни “Основи інформаційних систем” для студентів спеціальності 176 «Мікро- та наносистемна техніка». / Уклад.: М.О.Стрембіцький. – Тернопіль: ТНТУ 2023 – 12 с.

Укладач: к.т.н., Стрембіцький М.О.

Відповідальний за випуск: завідувач кафедри приладів і контрольно-вимірювальних систем Паламар М.І.

Розглянуто та затверджено на засіданні приладів і контрольно-вимірювальних систем Тернопільського національного технічного університету імені Івана Пулюя, протокол № 7 від «1» травня 2023 р.

Схвалено та рекомендовано до друку науково-методичною комісією факультету прикладних інформаційних технологій та електроінженерії ТНТУ, протокол № 10 від «5» травня 2023 р.

Призначені для полегшення засвоєння дисципліни «Основи інформаційних систем» і контролю знань студентів. Складається з урахуванням модульної системи навчання, рекомендацій до самостійної роботи і індивідуальних завдань, тем практичних занять, тестів, контрольних питань, здачі заліку, типової форми та вимог для комплексної перевірки знань з дисципліни.

## ВСТУП

Методичні вказівки до самостійної роботи студентів з дисципліни «Основи інформаційних систем» розроблені відповідно до навчального плану та робочої програми дисципліни і призначені для студентів спеціальності 176 «Мікро- та наносистемна техніка» підготовки освітньо-кваліфікаційного рівня «бакалавр».

Метою вивчення дисципліни "Основи інформаційних систем" є отримання знань про методiku, етапи розробки електронних систем та елементів мікро- та наносистемної техніки.

Завданням дисципліни є забезпечити створення методологічної основи для подальшої підготовки студентів із питань мікро- та наносистемної техніки та підвищення знань у практичній інженерній та науковій роботі.

У результаті вивчення навчальної дисципліни студент повинен

**знати:**

- об'єктно-орієнтованим підходом до програмування;
- принципи розробки ASP.net;
- життєві цикли сторінки, вивчення патернів MVC та ASP.NET.

**вміти:**

- розробки статичних та динамічних WEB-сторінок;
- використання JavaScript, jQuery для обміну інформацією між CDN серверами;
- використовувати модульне тестування ASP.NET MVC програм, а також інструментів автоматичного тестування.

Вивчення дисципліни «Основи інформаційних систем» передбачає засвоєння навчального матеріалу у вільний від аудиторних занять час у формі самостійної та індивідуальної навчально-дослідної роботи, призначеної формувати практичні навички роботи студентів із спеціальною літературою, орієнтувати їх на інтенсивну роботу, критичне осмислення здобутих знань і глибоке вивчення теоретичних і практичних проблем для підвищення якості професійної підготовки.

Самостійна робота над засвоєнням навчального матеріалу з дисципліни може виконуватися в бібліотеці, навчальних кабінетах та лабораторіях, в домашніх умовах. Згідно з навчальним планом дисципліни «Основи інформаційних систем» на самостійну роботу студентів відведено 48 годин.

Раціональна організація самостійної роботи вимагає від студента вмiлого розподілу свого часу між аудиторною і позааудиторною роботою. Виконання завдань із самостійної та індивідуальної роботи є обов'язковим для кожного студента.

Методичні вказівки до самостійної роботи студентів включають: тематичний план навчальної дисципліни; загальні рекомендації до організації самостійної роботи з дисципліни, в тому числі до організації індивідуального наукового-дослідного завдання; систему поточного й підсумкового контролю знань студентів; список рекомендованої літератури.

## 1 ТЕМАТИЧНИЙ ПЛАН НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

При вивченні дисципліни «Основи інформаційних систем» студент повинен ознайомитися з програмою навчальної дисципліни, її структурою, формами й методами навчання, видами й методами контролю знань.

Тематичний план дисципліни складається з двох модулів, які, в свою чергу, складаються із змістових модулів, кожен з яких об'єднує в собі відносно окремий самостійний блок дисципліни, що логічно пов'язує кілька укрупнених навчальних елементів дисципліни.

Навчальний процес здійснюється в таких формах: лекційні, практичні заняття і самостійна робота студента. Тематичний план дисципліни наведено в табл. 1.1

Таблиця 1.1 – Тематичний план дисципліни

№	Тема заняття та короткий зміст	Кількість годин	
		ДФН	ЗФН
1	2	3	4
1.	<b>Тема 1. Алгоритми в програмуванні.</b> Комп'ютерний алгоритм - це обчислювальна процедура, яка приймає набір кінцевих вхідних даних і перетворює їх на вихід, застосовуючи деяку математику та логіку.	1	0,5
2.	<b>Тема 2. ASP.NET.</b> Розуміння ASP.NET. Як ASP.NET робить роботу такою простою. Робота ASP.NET. Переваги ASP.NET. Сфера застосування ASP.NET	1	0,5
3.	<b>Тема 3. Історія WEB-розробки.</b> Традиційна ASP.NET WEB форми. Стек технологій ASP.NET Web Forms. ASP.NET Web Forms.	1	0,5
4.	<b>Тема 4. Веб розробка.</b> Веб стандарти та REST. Екстремальне програмування та розробка через тестування (test-driven development, TDD). Ruby on Rails. Sinatra. Node.js.	1	0,5
5.	<b>Тема 5. Основні переваги ASP.NET MVC.</b> Архітектура MVC. Жорсткий контроль над HTML та HTTP. Тестування. Можливість розробляти у найкращих сегментах платформи ASP.NET. Порівняння з ASP.NET Web Forms. Перехід від Web Forms до MVC.	1	0,5
6.	<b>Тема 6. MVC патерн.</b> Історія MVC. Доменна модель. ASP.NET реалізація MVC. ASP.NET реалізація MVC. Архітектура Model-View. Класична трирівнева (three-tier) архітектура. Класична трирівнева (three-tier) архітектура. Model-View-View Model патерн.	1	0,5

7.	<b>Тема 7. Проблемно-орієнтовне програмування (DDD).</b> Побудова доменної моделі. Повсюдно використовувана мова. Агрегати та спрощення. Визначення репозиторіїв.	1	0,5
8.	<b>Тема 8. Побудова слабозв'язаних компонентів.</b> Використання впровадження залежності (Dependency Injection). Використання контейнера застосування залежності. Ланцюжок залежностей. Управління життєвим циклом об'єкта. Конфігурація значень параметрів конструктора.	1	0,5
9.	<b>Тема 9. Поняття про мову розмітки.</b> Базові конструкції мови HTML. Поняття тегу. Структура HTML-документа. Заголовки. Абзаци та рядки. Одинарні теги.	1	0,5
10.	<b>Тема 10. Нумеровані та марковані списки.</b> Невпорядкований (маркований). Упорядкований (нумерований). Список визначень. Текстові гіперпосилання.	1	0,5
11.	<b>Тема 11. Використання таблиць у HTML-документах.</b> Текст всередині комірок. Горизонтальне вирівнювання вмісту всього рядка. Колірне оформлення таблиць.	2	0,5
12.	<b>Тема 12. Фрейми, їхні теги й атрибути.</b> Формат GIF. Формат PNG. Формат JPEG. Створення тла веб-сторінки. Вставлення зображень на веб-сторінку. Розміщення зображень у тексті. Графічні гіперпосилання. Формування карти гіперпосилань.	2	0,5
Усього годин		14	6

## 2. ЗАГАЛЬНІ РЕКОМЕНДАЦІЇ ДО ОРГАНІЗАЦІЇ САМОСТІЙНОЇ РОБОТИ З ДИСЦИПЛІНИ

Обов'язковим елементом успішного засвоєння навчального матеріалу дисципліни «Основи інформаційних систем» є самостійна робота студентів з вітчизняною і зарубіжною літературою з питань проектування електронних систем.

Самостійна робота є основним засобом оволодіння навчальним матеріалом у час, вільний від нормованих навчальних занять, тобто лекційних і практичних занять (аудиторної роботи).

Основні види самостійної роботи, на які повинні звертати увагу студенти:

- вивчення лекційного матеріалу;
- робота з опрацювання та вивчення рекомендованої літератури;
- підготовка до практичних занять;
- підготовка до дискусій та інших пропонованих викладачем завдань;
- робота над індивідуальним завданням;
- самоперевірка студентом власних знань за запитаннями для самодіагностики;
- підготуватися до дискусії в аудиторії щодо розуміння вивченого матеріалу;
- підготовка до поточного та підсумкового контролю.

*Опрацювання лекційного матеріалу.* У системі різних форм навчально-виховної роботи особливе місце належить лекції, де викладач надає студенту основну інформацію, навчає розмірковувати, аналізувати, допомагає опанувати ключові знання, а також спрямовує самостійну роботу студента.

Зв'язок лекції і самостійної роботи студента розглядається в таких напрямках:

- лекція як головна початкова ланка, що визначає зміст і обсяг самостійної роботи студента;
- методичні прийоми читання лекцій, що активізують самостійну роботу студентів;
- самостійна робота, яка сприяє поглибленому засвоєнню теми на базі прослуханої лекції.

Перший етап самостійної роботи починається з процесу слухання і записування лекції. Правильно складений конспект лекції – найефективніший засіб стимулювання подальшої самостійної роботи студентів. Студент повинен чітко усвідомити, що конспект – це короткий тезовий запис головних положень навчального матеріалу. Складання і вивчення конспекту – перший етап самостійної роботи студента над вивченням теми чи розділу. Конспект допомагає в раціональній підготовці до практичних занять, заліку, у визначенні напряму і обсягу подальшої роботи з літературними джерелами.

Під час підготовки до лекції студент повинен опрацювати матеріал попередньої лекції з використанням підручників та інших джерел літератури. На лекціях висвітлюють тільки основні теоретичні положення та найбільш

актуальні проблеми, тому більшість питань виноситься на самостійне опрацювання.

*Підготовка до практичних занять.* Підготовка до практичних занять розпочинається з опрацювання лекційного та методичного матеріалу до заданого заняття. Студент повинен самостійно ознайомитися з відповідним розділом робочої програми, підготувати відповіді на контрольні запитання, які подані в програмі у певній послідовності згідно з логікою засвоєння навчального матеріалу.

Практичні заняття збагачують і закріплюють теоретичні знання студентів, розвиваючи їх творчу активність, допомагають у набутті практичних навичок роботи за предметом навчальної дисципліни.

У процесі підготовки до практичних занять самостійна робота студентів є обов'язковою частиною навчальної роботи, без якої успішне і якісне засвоєння навчального матеріалу неможливе. Це свідчить про необхідність керування самостійною роботою студентів з боку викладача завдяки проведенню цілеспрямованих організаційних і контрольних заходів.

Відповідно до навчального плану з кожної теми курсу проводяться практичні заняття. Щороку викладачі уточнюють тематичний план проведення семінарських і практичних занять і ознайомлюють з ним студентів на першому занятті.

Викладач у вступній лекції рекомендує студентам основну і додаткову літературу, а також методичні рекомендації до самостійної роботи та до організації практичних занять з дисципліни. У методичних вказівках з кожної теми наведено перелік питань для теоретичної підготовки до заняття.

У разі, коли студент не може самостійно розібратися в якомусь питанні, він може отримати консультацію у викладача (згідно з графіком проведення консультацій викладачами кафедри приладів і контрольно-вимірювальних систем систем). Добре організовані консультації дозволяють спрямувати самостійну роботу в потрібному напрямі, зробити раціональною і підвищити її ефективність.

В таблиці 2.1 представлено перелік тем практичних робіт

Таблиця 2.1 – Перелік тем практичних робіт

№	Тема заняття	Кількість годин	
		ДФН	ЗФН
1.	Створення нового ASP.NET MVC проєкту. Динамічний вивід даних	4	1
2.	Додаток для вводу даних	4	1
3.	Валідація форми RSVP	4	1
4.	Основні засоби html	4	1
5.	Вивчення й практичне застосування стилів CSS	4	1
6.	Застосування JavaScript при створенні Web-Сторінок	4	1
7.	Застосування JavaScript при створенні Web form	4	2
Усього годин		28	8

### 3. ПИТАННЯ ДЛЯ САМОСТІЙНОГО ОПРАЦЮВАННЯ

№	Найменування робіт	Кількість годин	
		ДФН	ЗФН
1	2	3	4
1.	Практична робота №1 Практична робота №2 Практична робота №3 Практична робота №4 Практична робота №5 Практична робота №6 Практична робота №7	<b>28</b> 4 4 4 4 4 4	<b>42</b> 6 6 6 6 6 6
2.	Опрацювання окремих розділів, які не виносяться на лекції: 1. HTML5. 2. Інструменти для модульного тестування. 3. Інструменти автоматизованого тестування інтерфейсу. 4. URL шаблони. 5. Взаємодія з MVC. 6. Постійна доменна модель.	<b>18</b> 3 3 3 3 3 3	<b>32</b> 6 6 6 6 4 4
3.	Підготовка та складання екзаменів, тестування: - Залік - Тест №1 - Тест №2	<b>1,5</b> 0,5 0,5 0,5	<b>1,5</b> 0,5- 0,5 0,5
4.	Екзамен / залік	<b>0,5</b>	<b>0,5</b>
Усього годин		48	76



#### **4. СИСТЕМА ПОТОЧНОГО Й ПІДСУМКОВОГО КОНТРОЛЮ ЗНАНЬ СТУДЕНТІВ**

Оцінювання знань, вмінь і навичок студентів включає ті види занять, які згідно з програмою навчальної дисципліни «Основи інформаційних систем» передбачають лекційні, практичні заняття, самостійну роботу.

Перевірку і оцінювання знань студентів проводять в наступних формах:

- оцінювання роботи і знань студентів під час практичних занять;
- оцінювання виконання і захист практичних робіт;
- складання проміжного контролю знань за змістовими модулями;
- здача заліку.

Для кожного змістовного модуля передбачено певну форму поточного контролю. Результати поточного контролю автоматично, без участі студента, зараховуються при модульному контролі. Студент може покращити результати поточного контролю при модульному контролі через тестування.

Максимальна оцінка при I модульному контролі — 40 балів;

Максимальна оцінка при II модульному контролі — 35 балів.

Підсумковий контроль - залік.

Максимальна оцінка навчальної дисципліни — 100 балів.

## 5. ПЕРЕЛІК КОНТРОЛЬНИХ ЗАПИТАНЬ З ДИСЦИПЛІНИ

1. Що таке ASP.Net?
2. Що ви маєте на увазі під кешуванням у ASP.Net?
3. Які є типи кешування?
4. Що таке життєвий цикл сторінки? Які події відбуваються у життєвому циклі Page?
5. Що таке ViewState? У якому випадку життєвого циклу сторінки доступний ViewState?
6. Яка тривалість життя елементів у ViewState?
7. Який батьківський клас для всіх елементів керування веб-сервером?
8. Як ви зберігаєте значення у ViewState і отримуєте їх?
9. Що таке базовий клас, від якого успадковуються веб-форми?
10. Напишіть код для надсилання електронного листа із програми ASP.Net.
11. Що таке рядок запиту в ASP.Net?
12. Які відмінності між кодом поза і кодом Inline?
13. Що таке директиви в ASP.Net? Перелічіть усі важливі директиви.
14. Що таке режими стану сесії?
15. Що таке багатомовний веб-сайт?
16. Який об'єкт обертає стан або дані користувача?
17. Що таке об'єкт сеансу?
18. Як можна застосувати тему до програми ASP.Net?
19. Що таке веб-API? Який протокол використовується у веб-API?
20. Яку бібліотеку використовують тестувальники та розробники для розробки автоматизованих тестів та створення інструментів тестування?
21. Що таке HTML?
22. Яким чином забезпечується форматування тексту в HTML-документі?
23. Яка мета розробки HTML5?
24. Який синтаксис HTML?
25. Структура HTML-документа?
26. Яким чином здійснюється взаємодія між HTML-документом і сервером?
27. Які текстові редактори для роботи з HTML Ви знаєте?
28. Що означає DOCTYPE у HTML-документі?
29. Що таке заголовок HTML-документа?
30. Що таке тіло HTML-документа?
31. Для чого служить текст між тегами <title>?
32. Яким тегом забезпечуються посилання на інші документи HTML і його атрибути?
33. Яким тегом визначаються параграфи?
34. Яким тегом визначаються заголовки?
35. Яким тегом визначається найбільший заголовок?
36. Яким тегом визначається найменший заголовок?
37. Що визначається тегом <img>?
38. Перелічити HTML - елементи форматування тексту.
39. Які теги списків ви знаєте?

- 40.Що визначає тег <table>?
- 41.Яким тегом визначається заголовок таблиці?
- 42.Для чого використовують тег <div>?
- 43.Для чого використовують тег <span>?
- 44.Що визначають <header>, <nav>, <section>, і <footer> елементи?
- 45.Що представляє собою CSS?
- 46.Як використовується CSS?
- 47.У яких випадках який спосіб застосування CSS кращий?
- 48.Який синтаксис визначення стилю?
- 49.В чому полягає ідея спадкування форматів у документі?
- 50.Що означає клас і як його визначити?
- 51.Що представляє собою JavaScript?
- 52.Поняття об'єктної моделі JavaScript (властивості, методи, події)
- 53.Які є способи розміщення коду JavaScript на HTML-Сторінці?
- 54.Що представляють собою оброблювачі подій?
- 55.Що представляє собою ієрархія класів?
- 56.Які мультимедіа формати Ви знаєте?
- 57.Яка анімація використовується у технології флеш?
- 58.Як здійснюється впровадження QuickTime?
- 59.Що представляє собою формат "AVI"?
- 60.Як забезпечити підтримку формату "MP3"?
- 61.Що представляє собою форма, якими атрибутами вона володіє?
- 62.Яка різниця між GET і POST?
- 63.Назвіть елементи керування формами й для чого вони використовуються?
- 64.Як отримати доступ до даних введених з форми на стороні сервера?

## 6. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

### Базова

1. Голицына О.Л. Программное обеспечение, 2006 г.
2. Семакин И.Г. Основы программирования : Учебник / И.Г Семакин. - М. : Издательский центр "Академия", 2008. - 432с.

### Допоміжна

1. Дунаев В. Самоучитель JavaScript – СПб.: Питер, 2012.
2. Прохоренок Н.А. HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера, 3-е изд.- СПб.: БХВ-Петербург, 2012
3. Крокфорд Д. JavaScript. Сильные стороны: [пер. с англ.] - СПб.: Питер, 2009.
4. Колисниченко Д.Н. Современный сайт на PHP и JavaScript. - СПб.: Питер, 2009.
5. Ташков П.А. Веб-мастеринг: HTML, CSS, JavaScript, PHP, CMS, графика, раскрутка. СПб.: Питер, 2009
6. Хольцнер С. PHP в примерах. Пер. с англ. – М.: ООО «Бином-Пресс», 2007.