

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА

ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК ТА КІБЕРНЕТИКИ

Кафедра прикладної статистики



РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
ІНСТРУМЕНТАЛЬНІ СЕРЕДОВИЩА
ТА ТЕХНОЛОГІЇ ПРОГРАМУВАННЯ

для студентів

галузь знань **12 «Інформаційні технології»**
(шифр і назва)
спеціальність **124 «Системний аналіз»**
(шифр і назва спеціальності)
освітній рівень **бакалавр**
(молодший бакалавр, бакалавр, магістр)
освітня програма **«Системний аналіз»**
(назва освітньої програми)

вид дисципліни **вибіркова**

Форма навчання	денна
Навчальний рік	2021/2022
Семестр	3, 4
Кількість кредитів ECTS	7
з них семестр 3	4
семестр 4	3
Мова викладання, навчання та оцінювання	українська
Форма заключного контролю	залік, іспит


Пролонговано: на 20 22 / 20 23 р.р.) «31» 08 2022р.
(прізвище, ПІБ, дата)
на 20 ___ / 20 ___ р.р.) « ___ » ___ 20 ___ р.



Розробники: Омельчук Людмила Леонідівна, к.ф.-м.н., доцент кафедри «Теорії та технології програмування»

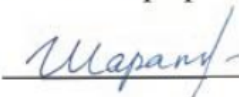
Шарапов Михайло Михайлович, к.ф.-м.н., доцент кафедри прикладної статистики

ЗАТВЕРДЖЕНО
Завідувач кафедри Прикладної Статистики


_____ (Лебедєв Є.О.)

Протокол № 1 від «27» серпня 2020 р.

Схвалено Гарантом освітньо-професійної програми першого рівня вищої освіти

«Системний аналіз»  М.М. Шарапов

«28» серпня 2020 року

Схвалено науково-методичною комісією факультету комп'ютерних наук та кібернетики

Протокол від «28» серпня 2020 року № 1

Голова науково-методичної комісії _____ (Омельчук Л.Л.)
(підпис)  (прізвище та ініціали)

«28» серпня 2020 року

1. Мета дисципліни – засвоєння знань з інструментальних середовищ та технологій програмування. Оволодіння базовими навичками проектування програмних систем, роботи з найбільш вживаними шаблонами проектування, набуття навичок об'єктно-орієнтованого програмування та оволодіння мовою програмування С#, набуття навичок використання інструментальних середовищ програмування, та використання технологій роботи з даними та технологій створення вебзастосунків.

2. Попередні вимоги до опанування або вибору навчальної дисципліни:

1. *Знати:* основні поняття програмування та принципи розробки програм; базові класичні алгоритми та різновиди структур даних

2. *Вміти:* проектувати, розробляти та тестувати програми на базовому рівні.

3. *Володіти елементарними навичками:* програмування мовами С, С++.

3. Анотація навчальної дисципліни:

Навчальна дисципліна “Інструментальні середовища та технології програмування” є вибірковою складовою освітньо-професійної програми підготовки фахівців за першим (бакалаврським) рівнем вищої освіти *галузі знань 12 „Інформаційні технології” зі спеціальності 124 „Системний аналіз”, освітньо-професійної програми – „Системний аналіз”*.

Дана дисципліна є обов'язковою навчальною дисципліною за *програмою “Системний аналіз”*.

Дисципліна викладається у 3 та 4 семестрах 2 курсу в обсязі 210 год. (7 кредитів ECTS) зокрема: лекції – 54 год., лабораторні – 48 год., консультації – 2 год., самостійна робота – 106 год. У курсі передбачено 3 частини та 3 контрольні роботи. Семестр 3 завершується заліком, семестр 4 завершується іспитом.

В результаті вивчення навчальної дисципліни студент повинен

знати основні поняття об'єктно-орієнтованого програмування, основні етапи життєвого циклу ПС, шаблони, антишаблони та принципи об'єктно-орієнтованого проектування програмного забезпечення, основи реляційних баз даних та мови запитів SQL, технології розробки інформаційних програмних систем, принципи роботи технологій доступу до даних на прикладі ADO.Net, основи HTML, CSS, JavaScript, базові елементи програмної інженерії, принципи роботи технологій створення веб-застосунків на прикладі ASP.Net.

вміти працювати з технологією ADO.Net на автономному рівні, працювати з технологією ADO.Net Entity Framework, працювати з технологією ASP.Net.

Для допуску до дисципліни „Інструментальні середовища та технології програмування” освітньо-професійної програми «Системний аналіз» студент повинен опанувати компетентності та результати навчання, які надає дисципліна „Програмування” освітньо-професійної програми «Системний аналіз». Дисципліна „Інструментальні середовища та технології програмування” пов'язана з дисципліною „Основи проектування баз даних”.

4. Завдання (навчальні цілі):

набуття знань, умінь та навичок (компетенцій) на рівні новітніх досягнень у програмуванні, відповідно до освітньої кваліфікації «Бакалавр з комп'ютерних наук». Зокрема, розвивати:

- здатність застосовувати знання у практичних ситуаціях;
- здатність оцінювати та забезпечувати якість виконуваних робіт;
- здатність використовувати сучасні інформаційні технології для комп'ютерної реалізації математичних моделей та прогнозування поведінки конкретних систем а саме: об'єктно-орієнтований підхід при проектуванні складних систем різної природи, прикладні математичні пакети, застосування баз даних і знань.

5. Результати навчання за дисципліною:

Результат навчання (1. знати; 2. вміти; 3. комунікація; 4. автономність та відповідальність)		Форми (та/або методи і технології) викладання і навчання	Методи оцінювання та пороговий критерій оцінювання (за необхідності)	Відсоток у підсумковій оцінці з дисципліни
Код	Результат навчання			
PH1.1	Знати основні поняття об'єктно-орієнтованого програмування, основи реляційних баз даних та мови запитів SQL, технології розробки інформаційних програмних систем	Лекція	Контрольна робота, іспит	17%
PH1.2	Знати принципи роботи технологій доступу до даних на прикладі ADO.Net	Лекція		
PH1.3	Знати основи HTML, CSS, JavaScript	Лекція	Контрольна робота, іспит	17% 15%
PH1.4	Знати базові елементи програмної інженерії, основні етапи життєвого циклу ПС, шаблони, антишаблони та принципи об'єктно-орієнтованого проектування програмного забезпечення.	Лекція		
PH1.5	Знати принципи роботи технологій створення веб-застосунків на прикладі ASP.Net	Лекція		
PH2.1	Вміти застосовувати на практиці інструментальні програмні засоби проектування та розробки програмного забезпечення.	лабораторна робота, самостійна робота	Захист лабораторної роботи, іспит	33%
PH2.2	Вміти працювати з технологією ADO.Net, Entity Framework Core.	лабораторна робота, самостійна робота	Захист лабораторної роботи, іспит	33%
PH2.3	Вміти працювати з технологією ASP.Net Core	лабораторна робота, самостійна робота	Захист лабораторної роботи, іспит	23%
PH4.1	Здатність до пошуку, оброблення та аналізу інформації з різних джерел	лабораторна робота, самостійна робота	Захист лабораторної роботи	10%

6. Співвідношення результатів навчання дисципліни із програмними результатами навчання

Результати навчання дисципліни	PH 1.1	PH 1.2	PH 1.3	PH 1.4	PH 1.5	PH 2.1	PH 2.2	PH 2.3	PH 4.1
Програмні результати навчання (з опису освітньої програми)									
ПРО8. Володіти сучасними методами розробки програм і програмних комплексів та прийняття оптимальних рішень щодо				+	+	+			+

складу програмного забезпечення, алгоритмів процедур і операцій									
ПР11. Знати і вміти застосовувати на практиці системи управління базами даних і знань та інформаційні системи.	+	+					+		
ПР13. Проектувати, реалізовувати, тестувати, впроваджувати, супроводжувати, експлуатувати програмні засоби роботи з даними і знаннями в комп'ютерних системах і мережах.	+			+		+	+	+	+

7. Схема формування оцінки.

7.1 Форми оцінювання студентів:

- семестрове оцінювання:

Перший семестр:

1. Контрольна робота (тест) 1: РН 1.1., РН 1.2 — 20 балів / 12 балів.
2. Контрольна робота (тест) 2: РН1.3 - 20 балів / 12 бали.
3. Домашні завдання 1-7: РН 1.1., РН 1.2, РН1.3- 15 балів / 9 балів.
4. Захист лабораторної роботи 1 (проекту): РН 2.1, РН3.1, РН4.1, РН4.2 – 15 балів / 9 балів.
5. Захист лабораторної роботи 2 (проекту): РН 2.1, РН3.1, РН4.1, РН4.2 – 15 балів / 9 балів.
6. Захист лабораторної роботи 3 (проекту): РН 2.1, РН3.1, РН4.1, РН4.2 – 15 балів / 9 балів.

Другий семестр:

1. Контрольна робота (тест): РН 1.1., РН 1.2 — 10 балів/6 балів.
2. Контрольна робота (тест): РН1.3, РН 1.4., РН 1.5 - 10 балів/6 балів.
3. Лабораторна робота 1 (етапи 1.0-1.7): РН 2.1, РН2.3 – 20 балів/12 балів.
4. Лабораторна робота 2 (етапи 2.0-2.4): РН 2.2 – 20 балів/12 балів.

- підсумкове оцінювання (у формі іспиту):

Перший семестр (у формі заліку)

Згідно пп. 4.6.1 та 7.1.5 «Положення про організацію освітнього процесу у Київському національному університеті імені Тараса Шевченка» залік виставляється на підставі поточного контролю (див. семестрове оцінювання) як сума оцінок/балів за всіма успішно оціненими результатами навчання; оцінки нижче від мінімального порогового рівня до підсумкової оцінки не додаються. До заліку допускаються всі студенти.

Другий семестр (у формі іспиту)

- максимальна кількість балів які можуть бути отримані студентом: 40 балів;
- результати навчання які будуть оцінюватись: РН1.1, РН1.2, РН1.3, РН1.4, РН1.5, РН2.1, РН2.2, РН2.3;
- форма проведення і види завдань: письмова робота.

Види завдань:

Структура екзаменаційної роботи та критерії оцінювання:

1. Теоретичне запитання з БД (РН1.1).
2. Теоретичне запитання з SQL (РН1.1).
3. Теоретичне запитання з технологій доступу до даних (РН1.2).
4. Теоретичне запитання ADO.NET (РН1.2).
5. Теоретичне запитання ADO.NET(РН1.2).
6. Теоретичне запитання з Entity Framework, або програмної інженерії (в залежності від варіанту) (РН1.2, РН1.4).
7. Тестове запитання з HTML, CSS, JavaScript (РН1.3).
8. Письмове завдання з ASP.NET (РН1.5).
9. Тестове запитання з Entity Framework (РН1.2).
10. Тестове запитання з HTML, CSS, JavaScript (РН1.3).
11. Тестове запитання з LINQ (РН1.4).
12. Письмове запитання з ASP.NET (РН1.5).
13. Письмове з БД (розробити діаграму БД) (РН1.1).

Критерії оцінювання екзаменаційної роботи

Завдання	Вид завдання	Максимальний відсоток (бал)	Всього відсотків (балів)
----------	--------------	-----------------------------	--------------------------

Завдання 1, 2, 3, 4, 5, 6	Письмове запитання з курсу	по 6% (1.6 балів)	36% (14.4 бали)
Завдання 7, 9, 10	Тестові завдання	по 5% (2 бали)	15% (6 балів)
Завдання 8	Тестове завдання	9% (3.6 балів)	9% (3.6 балів)
Завдання 11	Тестове завдання	10% (4 бали)	10% (4 бали)
Завдання 12, 13	Письмове завдання	по 15% (6 балів)	30% (12 балів)
Всього			100% (40 балів)

Перший семестр

Умови домашніх завдань

1. Use-case-діаграма та діаграма класів для заданої предметної області. Посібник [10] ст. 34-35.
2. Діаграма класів та діаграма послідовності використання для заданої предметної області. Посібник [10] ст. 36-37.
3. Завдання з посібника [2] ст. 61-77.
4. Завдання з посібника [2] ст. 87-106.
5. Завдання з посібника [2] ст. 156-192.
6. Завдання з посібника [2] ст. 233-235, 236-243.
7. Завдання з посібника [2] ст. 258-262.

Умови лабораторних робіт:

Лабораторна робота 1: Розробка програми для роботи з електронними таблицями (аналіз та обчислення виразів).

Лабораторна робота 2: Робота з XML-документами.

Лабораторна робота 3: Розробка програм обробки файлів та редагування текстів.

Деталізовані умови розміщено в посібнику [2].

Запитання для підготовки до контрольних робіт першого семестру

1. Основні принципи моделювання програмних систем.
2. Життєвий цикл ПС. Моделі життєвого циклу ПС. Ітеративні моделі життєвого циклу. Керування ризиками.
3. Життєвий цикл ПС. Моделі життєвого циклу ПС. Модель Гантера «Фази-Функції».
4. Діаграми класів та шаблони проектування. Структура шаблонів. Класифікація шаблонів: породжуючі, структурні, поведінки.
5. Різновиди шаблонів проектування.
6. Шаблон „абстрактна фабрика”.
7. Шаблон „прототип”.
8. Шаблон „одиначний об’єкт”.

9. Шаблон „міст”.
10. Шаблон „ланцюг обов’язків”.
11. Шаблон „декоратор”.
12. Шаблон „ітератор”.
13. Антишаблони.
14. Огляд технології .NET Framework.
15. Середовище CLR.
16. Бібліотека класів Framework Class Library (FCL).
17. Загальна структура C#-програми.
18. Типи значень (вбудовані типи значень, порядковий тип(enum), структури (struct)). Типи посилань (вбудовані типи посилань (object, string), масиви, делегати, інтерфейси, класи).
19. Ітератори.
20. Робота з рядками.
21. Перетворення типів в C#.
22. Оператори обробки виключень в C#.
23. Неявні типи, анонімні типи (тип var) та тип dynamic.
24. Робота з файлами в директорії.
25. Створення підкаталогів.
26. Робота з файлами (читання, запис, створення, видалення).
27. Основні компоненти XML-документа.
28. Моделі DOM та SAX для обробки XML.
29. Застосування XmlDocument. Застосування XmlReader.
30. LINQ to XML.
31. Порівняння LINQ to XML та DOM.
32. Мова запитів XPath.
33. Трансформація XML.
34. Трансформація XML з використанням XSLT.
35. Шаблон XSL.
36. Лямбда-вирази. Призначення та синтаксис.
37. LINQ (Language Integrated Query – мова інтегрованих запитів). Призначення та синтаксис.
38. LINQ to Objects
39. LINQ to DataSet та SQL
40. LINQ to Entities

Другий семестр

Умови лабораторних робіт

Лабораторна робота № 1. Розробка веб-додатку

Набір технологій:

- Entity Framework Core (EF Core) Database First – ORM Framework
- ASP.NET MVC
- Bootstrap

Етапи:

1.0 Початковим етапом до виконання лабораторних робіт є формулювання персонального завдання, розробка та затвердження викладачем діаграми прецедентів (Use-case діаграма) UML, яка демонструє особливості обраної предметної області та діаграми бази даних, на якій зображено всі таблиці з іменами їх стовпців, позначено ключові стовпці та усі зв’язки між таблицями (максимум 2 бали).

- 1.1. Створити базу даних в [SQL Server Management Studio](#) (SSMS), або в іншій, попередньо узгодженій з викладачем СКБД (максимум 3 бали).
- 1.2. Створити новий проєкт та під'єднати наявну БД (див. попередній етап) (максимум 2 бали)..
- 1.3. Налаштування, створення контролерів та представлень (максимум 3 бали).
- 1.4. Зміна майстер-сторінки та стилізація (максимум 2 бали).
- 1.5. Відображення даних на діаграмі (максимум 2 бали).
- 1.6. Робота з файлами (формування та збереження звітів) у форматі MS Excel (експорт та імпорт файлів) (максимум 3 бали).
- 1.7. Робота з автентифікацією, авторизацією та ролями (максимум 3 бали).

Лабораторна робота № 2. Розробка веб-додатку

Набір технологій:

- **Entity Framework Core (EF Core) Code First – ORM Framework**
- **Web API (ASP.NET)**
- **HTML, JavaScript**

Етапи:

- 2.0. Формулювання персонального завдання, розробка та затвердження викладачем діаграми прецедентів (Use-case діаграма) UML, яка демонструє особливості обраної предметної області (максимум 3 бали).
- 2.1. Створення додатку для Web API. Створення моделі та класу контексту даних (максимум 2 бали).
- 2.2. Реалізація контролерів та їх тестування (максимум 2 бали)..
- 2.3. Виклик веб-API за допомогою JavaScript.
- 2.4 Unit-тестування (максимум 1 додатковий бал).

З детальною інформацією можна ознайомитися в посібнику [1].

Запитання для підготовки до іспиту

1. База даних. Система керування БД.
2. Моделі даних.
3. Поняття база даних (БД). Система керування базами даних (СКБД). Моделі даних. Поняття реляційної бази даних. Реляційна алгебра. Первинний ключ, зовнішні ключі.
4. Реляційна алгебра.
5. Транзакція. Тригер. Представлення.
6. Технології доступу до даних. Призначення. Приклади.
7. Коротка характеристика ADO.
8. Коротка характеристика ODBC.
9. Коротка характеристика OLE DB
10. Коротка характеристика BDE
11. Коротка характеристика DAO, JDBC.
12. Рівні застосування бібліотеки ADO.NET (підключений (connected layer) автономний рівень (disconnected layer) та за допомогою технології Entity Framework).
13. Архітектура ADO.NET. Постачальники даних. Connection, Command, DataReader, DataAdapter, Parameter, Transaction. Підключення до бази даних. Команди. Виконання команди. Параметризовані об'єкти команд. Транзакції.
14. Архітектура ADO.NET.
15. Рівні застосування бібліотеки ADO.NET.
16. ORM (Object-relational Mapping).
17. Архітектура ADO.NET. DataSet.
18. Архітектура ADO.NET. DataAdapter.
19. Архітектура доступу до даних Entity Framework.
20. Модель EDM (Entity Data Model). edmx.
21. ADO.NET Entity Framework (EF). EntityClient. DbContext.
22. ADO.NET. Рівні застосування.

23. Рівні застосування бібліотеки ADO.NET (підключений (connected layer) автономний рівень (disconnected layer) та за допомогою технології Entity Framework). Автономний рівень.
24. ADO.NET. Постачальники даних.
25. ADO.NET. Команди. Виконання команд.
26. ADO.NET. Підключення до даних.
27. ADO.NET. Транзакції
28. ADO.NET. Архітектура.
29. Архітектура доступу до даних Entity Framework.
30. Підключення. Entity Data Model (EDM). Концептуальна та сутнісна моделі даних. Доступ до даних. Архітектура доступу до даних. Доступ до даних сутностей та їх зміна. Запити до концептуальної моделі даних. EntityClient, DbContext.
31. LINQ to Entities.
32. Селектори jQuery.
33. Шаблон проектування Model-view-controller (MVC, «Модель-представлення-контролер»).
34. Архітектура ASP.NET MVC 4.
35. ASP.NET MVC. Контролери.
36. ASP.NET MVC. Моделі.
37. ASP.NET MVC. Представлення.
38. Опишіть підхід до роботи з даними в Entity Framework Database First.
39. Опишіть підхід до роботи з даними в Entity Framework Model First.
40. Опишіть підхід до роботи з даними в Entity Framework Code First
41. Структура SWEBOOK.
42. Наведіть визначення програмної інженерії та інформатики. Опишіть їх зв'язок та відмінності.
43. Короткий огляд HTTP. HTML. CSS. Підключення CSS до HTML. Правила CSS. Селектори CSS. Класи. Псевдоелементи. Блоки. Списки.
44. Шаблон проектування Model-view-controller (MVC, «Модель-представлення-контролер»).
45. ASP.NET MVC. Контролери. Моделі.
46. ASP.NET MVC. Представлення. Двигун представлення Razor.
47. Програмна інженерія. Життєвий цикл програмного продукту та моделі процесу.
48. Agile методи. Принципи та основні ідеї.
49. Життєвий цикл програмного продукту та моделі процесу; моделі оцінки процесу; метрики процесу розробки програмного забезпечення.
50. Еволюція програм: супровід програм, характеристики супроводжуваного програмного забезпечення (ПЗ), реінженірінг, успадковані системи, повторне використання ПЗ
51. Масштабованість, продуктивність і ефективність інформаційних технологій та систем. Перспективи розвитку.
52. Розробка ПЗ: об'єктно-орієнтований аналіз і проектування; проектування для повторного використання; шаблони проектування; програмні оточення; програмні засоби тестування.

Студент не допускається до іспиту, якщо під час семестру набрав менше ніж 36 бали. Студент допускається до іспиту за умови виконання 70% передбачених планом лабораторних робіт.

7.2 Організація оцінювання:

Перший семестр:

Терміни проведення форм оцінювання:

1. Контрольна робота 1 (тест): до 7 тижня семестру.
2. Контрольна робота 2 (тест): до 14 тижня семестру.
3. Домашні завдання 1: до 3 тижня семестру.
4. Домашні завдання 2: до 4 тижня семестру.
5. Домашні завдання 3: до 5 тижня семестру.
6. Домашні завдання 4: до 6 тижня семестру.
7. Домашні завдання 5: до 9 тижня семестру.

8. Домашні завдання 6: до 11 тижня семестру.
9. Домашні завдання 7: до 12 тижня семестру.
10. Захист лабораторної роботи 1 (проекту): до 7 тижня семестру.
11. Захист лабораторної роботи 2 (проекту): до 10 тижня семестру.
12. Захист лабораторної роботи 3 (проекту): до 13 тижня семестру.

Другий семестр:

Терміни проведення форм оцінювання:

1. Контрольна робота (тест): до 7 тижня семестру.
2. Контрольна робота (тест): до 15 тижня семестру.
3. Лабораторна робота 1 (проект): до 13 тижня семестру.

Етап 1.0: до 2 тижня семестру;

Етап 1.1: до 3 тижня семестру;

Етап 1.2: до 6 тижня семестру;

Етап 1.3: до 6 тижня семестру;

Етап 1.4: до 7 тижня семестру;

Етап 1.5: до 8 тижня семестру;

Етап 1.6: до 10 тижня семестру;

Етап 1.7: до 13 тижня семестру;

4. Лабораторна робота 2 (проект): до 10 тижня семестру.

Етап 2.0: до 13 тижня семестру;

Етап 2.1: до 15 тижня семестру;

Етап 2.2: до 16 тижня семестру;

Етап 2.3: до 18 тижня семестру;

Етап 2.4: до 18 тижня семестру;

Студент має право на одне перескладання кожної контрольної роботи із можливістю отримання максимально 8 балів за кожну. Термін перескладання визначається викладачем.

У разі неякісного виконання лабораторної роботи, викладач має право не зарахувати лабораторну роботу, або знизити за неї бали.

Студент має право здавати лабораторні роботи після закінчення визначеного для них терміну, але з втратою одного балу за кожен тиждень, який пройшов з моменту закінчення терміну її здачі.

7.3 Шкала відповідності оцінок

Відмінно / Excellent	90-100
Добре / Good	75-89
Задовільно / Satisfactory	60-74
Незадовільно / Fail	0-59
Зараховано / Passed	60-100
Не зараховано / Fail	0-59

**8. Структура навчальної дисципліни. Тематичний план лекцій і лабораторних занять
СЕМЕСТР 3**

№ лекції	Назва лекції	Кількість годин		
		Лекції	Лаборат. заняття	Сам. р-та
Частина 1. Основи об'єктно-орієнтованої методології. Знайомство з мовою C#				
1.	Тема 1. Основні поняття ООП. Порівняння ООП з парадигмою процедурного програмування. Втілення ООП в мовах програмування. Самостійна робота: логічне програмування. Методи і алгоритми обчислень. Структури даних і механізми управління. Виконання лабораторної роботи 1.	2	2	4
2.	Тема 2. Основні поняття об'єктно-орієнтованого програмування. Основні етапи життєвого циклу програмних систем. Самостійна робота: функціональне програмування, опрацювання лекційного матеріалу. Виконання домашнього завдання 1, або виконання лабораторної роботи 1.	2	2	4
3.	Тема 3. Основні принципи моделювання програмних систем. Огляд мови UML. Самостійна робота: застосування шаблонів проектування та UML для об'єктно-орієнтованого моделювання ПС. Виконання домашнього завдання 2, виконання лабораторної роботи 1.	2	2	4
4.	Тема 4. .NET Framework. Порівняльний аналіз мов C++ та C#. Самостійна робота: опрацювання лекційного матеріалу. Виконання домашнього завдання 3, виконання лабораторної роботи 1.	2	2	4
5.	Тема 5. Порівняльний аналіз мов C++ та C#. Класи, перетворення типів, оператори та вирази, оператори обробки виключень, неявні, анонімні та динамічні типи даних. Самостійна робота: опрацювання лекційного матеріалу. Виконання домашнього завдання 4, виконання лабораторної роботи 1.	2	2	4
6.	Тема 6. Огляд технології .NET Framework. Середовище CLR. Бібліотека класів Framework Class Library (FCL). Самостійна робота: Основи синтаксичного аналізу та обчислення виразів. Використання ANTLR для виконання лабораторної роботи №1. Виконання домашнього завдання 5. Виконання лабораторної роботи 2.	2	2	4
7.	Тема 7. Вимоги і рекомендації з написання коду. Рефакторинг. Мова запитів XPath. Трансформація XML. Трансформація XML з використанням XSLT. Самостійна робота: Основні компоненти XML документа. Застосування XML документа. Моделі DOM та SAX для обробки XML. Виконання домашнього завдання 5. Виконання лабораторної роботи 1.	4	1	4

	<i>Контрольна робота 1</i>		1	
Всього по частині 1		16	14	28
Частина 2. Функціональні можливості C#. Шаблони проектування. Огляд технології .NET Core				
8.	Тема 8. Функціональні можливості C#. LINQ to Object, LINQ to XML. Лямбда-вирази. Застосування лямбда-виразів. Самостійна робота: опрацювання лекційного матеріалу. Виконання лабораторної роботи 2. Виконання домашнього завдання 5.	2	2	4
9.	Тема 9. Шаблони проектування. Призначення. Класифікація. Самостійна робота: опрацювання лекційного матеріалу. Виконання лабораторної роботи 2. Виконання домашнього завдання 6.	2	2	4
10.	Тема 10. Породжуючі патерни (шаблони) проектування. Самостійна робота: опрацювання лекційного матеріалу. Виконання лабораторної роботи 3. Виконання домашнього завдання 6.	2	2	4
11.	Тема 11. Структурні патерни (шаблони) проектування. Самостійна робота: опрацювання лекційного матеріалу. Виконання лабораторної роботи 3. Виконання домашнього завдання 7.	2	2	4
12.	Тема 12. Патерни поведінки. Антишаблони.. Самостійна робота: опрацювання лекційного матеріалу. Виконання лабораторної роботи 3. Виконання домашнього завдання 7, підготовка доповідей за патернами проектування.	2	2	4
13.	Тема 13. Принципи проектування S.O.L.I.D. та інші. Самостійна робота: опрацювання лекційного матеріалу. Виконання лабораторної роботи 3.	4	2	4
14.	Тема 14. Інструментальні середовища та технології програмування. Огляд. Самостійна робота: Опрацювання лекційного матеріалу. Виконання лабораторної роботи 3.	2		2
15.	Тема 15. Методи тестування. Програмні засоби тестування. Поняття про верифікацію та валідацію. Самостійна робота: Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етап 2.4).	2		2
	<i>Контрольна робота 2</i>		2	
Контроль за підсумками лабораторних робіт 2 та 3				
Всього по частині 2		18	14	28
Консультація			2	
ВСЬОГО		34	28	56

Загальний обсяг 120 год., в тому числі:

Лекцій – **34 год.**

Лабораторні заняття - **28 год.**

Консультації – **2 год.**

Самостійна робота - **56 год.**

СЕМЕСТР 4

№ лекції	Назва лекції	Кількість годин		
		Лекції	Лабора- т. занять	Сам. р-та
Частина 3. Огляд та застосування ORM (Object-relational mapping, Об'єктно-реляційне відображення)-технологій				
1.				
16.	Тема 16. Засоби для роботи з реляційними базами даних. Огляд ORM-технологій. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етапи 1.2, 1.3).	2	2	6
17.	Тема 17. Бібліотека ADO.NET. Архітектура ADO.NET. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етапи 1.2, 1.3).	2	2	6
18.	Тема 18. Загальна характеристика Entity Framework Core (EF Core). Модель даних. Архітектура доступу до даних Entity Framework. Застосування LINQ to Entities. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етап 1.4).	2	2	4
19.	Тема 19 SQL в EF Core. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етап 1.6).	2	1	4
	<i>Контрольна робота 1</i>		1	
Всього по частині 3		8	8	20
Частина 4. Розробка web-застосунків				
20.	Тема 20. Основи HTML, CSS, JavaScript. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етапи 1.7, 2.0).	2	2	4
21.	Тема 21. ASP.NET Core. Основи. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етапи 1.7, 2.0).	2	2	4
22.	Тема 22. ASP.NET Core. Dependency Injection. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етапи 1.7, 2.0).	2	2	6
23.	Тема 23. Обробник представлень Razor, ASP.NET Core MVC. Контролери. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етапи 2.1, 3).	2	2	6
24.	Тема 24. ASP.NET Core MVC. Представлення. Моделі. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етапи 2.1, 3.0).	2	2	4
25.	Тема 25. ASP.NET Core (Web API). Async/await в C#. Короткий огляд.	2	2	6

	<i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етап 2.2).			
	<i>Контрольна робота 2</i>		1	
Всього по частині 2		12	12	30
ВСЬОГО		20	20	50

Загальний обсяг 90 год., в тому числі:

Лекцій – **20 год.**

Лабораторні заняття - **20 год.**

Самостійна робота - **50 год.**

9. Рекомендовані джерела:

Основна

1. Омельчук Л.Л. Інструментальні середовища та технології програмування. Лабораторний практикум / Л.Л. Омельчук, Н.Г. Русіна. - Одеса: Айс Прінт, 2020. - 175 с.
2. Омельчук Л.Л. [Об'єктно-орієнтоване програмування. Лабораторний практикум: навчальний посібник](#) / Л.Л. Омельчук. – Київ, 2021. - 264 с.
2. В.В. Зубенко, Л.Л. Омельчук. Програмування : навчальний посібник (гриф МОН України) / - К. : ВПЦ "Київський університет", 2011. - 623 с.
3. <http://msdn.microsoft.com/ru-RU/>
5. Лаврищева Е.М. Современные методы программирования: возможности и инструменты // Проблемы програмування. – 2006.– № 2-3. – С.60-74.
6. Буч Г. Объектно-ориентированное программирование. – Киев, Диалектика, 1993
7. Буч Г., Якобсон А., Рамбо Дж. UML. Классика CS. 2-е изд. / Пер. с англ.; Под общей редакцией проф. С. Орлова — СПб.: Питер, 2006. — 736 с. ISBN 5-469-00599-2
8. Крэг Ларман. Применение UML 2.0 и шаблонов проектирования = Applying UML and Patterns : An Introduction to Object-Oriented Analysis and Design and Iterative Development. — 3-е изд. — М.: Вильямс, 2006. — 736 с. — ISBN 0-13-148906-2

Додаткова:

9. Ставровский А.Б, Карнаух Т.О. Перші кроки програмування.-К.:Диалектика.-2005, с.389.
10. Кнут Д. Искусство программирования.Т.1,2,3. Изд 3-е, испр. -М.СбП.К.:Вильямс, 2001.
11. Вирт Н. Алгоритмы и структуры данных. - М.:Мир,1989.с.263.
12. Джеймс Р. Грофф, Пол Н. Вайнберг, Эндрю Дж. Оппель. SQL: полный справочник = SQL: The Complete Reference. — 3-е изд. — М.: Вильямс, 0. — 960 с. — ISBN 978-5-8459-1654-9
13. Алекс Кригель, Борис Трухнов. SQL. Библия пользователя. Язык запросов SQL = SQL Bible. — 2-е изд. — М.: Диалектика, 2009. — 752 с. — ISBN 978-5-8459-1546-7
14. К. Дж. Дейт. Введение в системы баз данных / Пер. с англ. — 8-е изд. — М.: Вильямс, 2005. — 1328 с.

10. Додаткові ресурси:

<https://metanit.com/sharp/entityframeworkcore/>