

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА

ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК ТА КІБЕРНЕТИКИ

Кафедра теорії та технології програмування

«ЗАТВЕРДЖУЮ»

Заступник декана
з навчальної роботи



Кашпур О.Ф.

« 28 » серпня 2020 року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
ІНСТРУМЕНТАЛЬНІ СЕРЕДОВИЩА
ТА ТЕХНОЛОГІЇ ПРОГРАМУВАННЯ

для студентів

галузь знань **12 «Інформаційні технології»**
(шифр і назва)
спеціальність **122 «Комп'ютерні науки»**
(шифр і назва спеціальності)
освітній рівень **бакалавр**
(молодший бакалавр, бакалавр, магістр)
освітня програма **«Інформатика»**
(назва освітньої програми)

вид дисципліни **обов'язкова**

Форма навчання	денна
Навчальний рік	2020/2021
Семестр	4
Кількість кредитів ECTS	5
Мова викладання, навчання та оцінювання	українська
Форма заключного контролю	іспит

Викладачі: к.ф.-м.н., доц. Омельчук Л.Л. (лекції, лабораторні заняття),
к.п.н., асистент Русіна Н.Г. (лабораторні заняття)

Пролонговано: на 20 __/20__ н.р. _____ (_____) «__» __ 20__ р.
(підпис, ПІБ, дата)

на 20 __/20__ н.р. _____ (_____) «__» __ 20__ р.
(підпис, ПІБ, дата)

КИЇВ – 2020

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

ЗАТВЕРДЖЕНО

Зав. кафедри «Теорії та технології програмування»

Нікітченко М.С.

(підпис)

(прізвище та ініціали)

Протокол № 1 від «28» серпня 2020 р.

Схвалено гарантом освітньо-професійної програми «Інформатика»

Омельчук Л.Л. «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

(підпис)

Омельчук Л.Л.

(прізвище та ініціали)

1. Мета дисципліни – засвоєння знань з інструментальних середовищ та технологій програмування. Оволодіння базовими навичками проектування програмних систем, набуття навичок використання інструментальних середовищ програмування, та використання технологій роботи з даними та технологій створення веб-додатків.

2. Попередні вимоги до опанування або вибору навчальної дисципліни:

1. *Знати:* основні поняття об'єктно-орієнтованого програмування, основні етапи життєвого циклу ПС, шаблони, антишаблони та принципи об'єктно-орієнтованого проектування програмного забезпечення.

2. *Вміти:* застосовувати на практиці інструментальні програмні засоби проектування та розробки програмного забезпечення.

3. *Володіти елементарними навичками:* програмування мовою C#.

3. Анотація навчальної дисципліни:

Навчальна дисципліна “Інструментальні середовища та технології програмування” є складовою освітньо-професійної програми підготовки фахівців за першим (бакалаврським) рівнем вищої освіти *галузі знань 12 „Інформаційні технології” зі спеціальності 122 „Комп’ютерні науки”, освітньо-професійної програми – „Інформатика”.*

Дана дисципліна є обов’язковою навчальною дисципліною за *програмою “Інформатика”.*

Викладається у 4 семестрі 2 курсу в **обсязі – 150 год.**

(5 кредитів ECTS) зокрема: *лекції – 34 год., лабораторні – 38 год., консультації – 2 год., самостійна робота – 76 год.* У курсі передбачено **2 частини** та **2 контрольні роботи.** Завершується дисципліна – **іспитом в 4 семестрі.**

В результаті вивчення навчальної дисципліни студент повинен

знати 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 «Інформатика» студент повинен опанувати компетентності та результати навчання, які надає дисципліна „Об’єктно-орієнтоване програмування” освітньо-професійної програми «Інформатика». Дисципліна „Інструментальні середовища та технології програмування” пов’язана з дисципліною „Бази даних та інформаційні системи”. Дисципліна „Інструментальні середовища та технології програмування” є базовою для засвоєння дисципліни «Системного програмування».

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 проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління;
- здатність реалізувати багаторівневу обчислювальну модель на основі архітектури клієнт-сервер, включаючи бази даних, сховища даних і бази знань, для забезпечення обчислювальних потреб багатьох користувачів, обробки транзакцій, у тому числі на хмарних сервісах;
- здатність застосовувати методології, технології та інструментальні засоби для управління процесами життєвого циклу інформаційних і програмних систем, продуктів і сервісів інформаційних технологій відповідно до вимог замовника;
- здатність використовувати інтелектуальні інформаційні технології;
- здатність реалізовувати фази та ітерації життєвого циклу створення програмних систем на основі моделей та методів розробки програмного забезпечення.

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 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	Лекція		
PH1.5	Знати принципи роботи технологій створення веб-застосунків на прикладі ASP.Net	Лекція		
PH2.1	Вміти працювати з технологією ADO.Net на автономному рівні	лабораторна робота, самостійна робота	Захист лабораторної роботи, іспит	33%
PH2.2	Вміти працювати з технологією ADO.Net Entity Framework	лабораторна робота, самостійна робота	Захист лабораторної роботи, іспит	33%
PH2.3	Вміти працювати з технологією ASP.Net	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 Програмні результати навчання	РН 1.1	РН 1.2	РН 1.3	РН 1.4	РН 1.5	РН 2.1	РН 2.2	РН 2.3	РН 4.1
<i>(з опису освітньої програми)</i>									
ПРН9. Використовувати інструментальні засоби розробки клієнт-серверних застосунків, проектувати концептуальні, логічні та фізичні моделі баз даних, розробляти та оптимізувати запити до них.	+	+				+		+	
ПРН11. Володіти навичками використання методології управління життєвим циклом програмного забезпечення, продуктів і сервісів інформаційних технологій відповідно до вимог і обмежень замовника, вміти готувати проектну документацію (техніко-економічне обґрунтування, технічне завдання, бізнес-план, креативний бриф, угоду, договір, контракт та ін.).		+	+	+	+		+		+

7. Схема формування оцінки.

7.1 Форми оцінювання студентів:

- семестрове оцінювання:

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 – 10 балів/6 балів.
5. Звіт (етап 3.0): РН4.1 – 10 балів/6 балів.

- підсумкове оцінювання (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)

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).

Звіт до першої лабораторної роботи:

Звіт до лабораторної роботи повинен відповідати усім вимогам, які висуваються до курсових робіт

http://csc.knu.ua/media/filer_public/4f/74/4f7459c9-9e5a-4a77-

З детальною інформацією можна ознайомитися за посиланням:

<https://classroom.google.com/c/MjI0MTAxMTAxMTU0?cjc=ljl46u7>

За бажанням студента та опанування ним дисципліни «Об'єктно-орієнтоване програмування» з оцінкою не нижче 75 балів лабораторна робота 1 та звіт можуть бути замінені колективним проєктом в межах міждисциплінарного проєкту зі студентами 4-го року навчання ОПП «Інформатика».

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:

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;
5. *Звіт у форматі курсової роботи (етап 3.0): до 15 тижня семестру.*

Лабораторна робота № 1. Розробка веб-додатку

Лабораторна робота № 2. Розробка веб-додатку

Звіт до першої лабораторної роботи:

Звіт до лабораторної роботи повинен відповідати усім вимогам, які висуваються до курсових робіт http://csc.knu.ua/media/filer_public/4f/74/4f7459c9-9e5a-4a77-

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

8. Структура навчальної дисципліни. Тематичний план лекцій і лабораторних занять

№ лекції	Назва лекції	Кількість годин		
		Лекції	Лаборат. заняття	Сам. р-та
Частина 1. Огляд та застосування ORM (Object-relational mapping, Об'єктно-реляційне відображення)-технологій				
1.	Тема 1. Інструментальні середовища та технології програмування. Огляд.. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етап 1.0).	2	4	6
2.	Тема 2. Поняття реляційної бази даних. Огляд сучасних СУБД <i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етап 1.1).	2	2	4
3.	Тема 3. 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	4
4.	Тема 4. Бібліотека ADO.NET. Архітектура ADO.NET. Рівні застосування бібліотеки ADO.NET. Підключений (connected layer) рівень. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етапи 1.2, 1.3).	2	2	5
5.	Тема 5. Технологія ADO.NET. Автономний рівень (disconnected layer). Застосування LINQ to DataSet та SQL. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етапи 1.2, 1.3).	2	2	5
6.	Тема 6. Загальна характеристика Entity Framework Core (EF Core). Модель даних. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етап 1.4).	2	2	5
7.	Тема 7. Архітектура доступу до даних Entity Framework. Застосування LINQ to Entities. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етап 1.5).	2	2	5
8.	Тема 8. SQL в EF Core. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етап 1.6).	4	2	6
	<i>Контрольна робота 1</i>		2	
Всього по частині 1		18	20	40
Частина 2. Розробка web-застосувань. Елементи інженерії програмних систем				
9.	Тема 9. Основи HTML, CSS, JavaScript. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етапи 1.7, 2.0).	2	2	4
10.	Тема 10. ASP.NET Core. Основи. <i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етапи 1.7, 2.0).	2	2	4
11.	Тема 11. 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

12.	Тема 12. <i>Обробник представлень Razor, ASP.NET Core MVC. Контролери.</i> <i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етапи 2.1, 3).	2	2	6
13.	Тема 13. <i>ASP.NET Core MVC. Представлення. Моделі.</i> <i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етапи 2.1, 3.0).	2	2	4
14.	Тема 14. <i>ASP.NET Core (Web API). Async/await в C#. Короткий огляд.</i> <i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етап 2.2).	2	2	4
15.	Тема 15. <i>Async/await в C#. Короткий огляд.</i> <i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етап 2.3).	2	2	4
16.	Тема 16. <i>Методи тестування. Програмні засоби тестування. Поняття про верифікацію та валідацію.</i> <i>Самостійна робота:</i> Опрацювання лекційного матеріалу та власного проекту лабораторного практикуму (етап 2.4).	2	2	4
	<i>Контрольна робота 2</i>		2	
	Всього по частині 2	16	18	36
	ВСЬОГО	34	38	76

Загальний обсяг 150 год., в тому числі:

Лекцій – **34 год.**

Лабораторні заняття - **38 год.**

Консультації – **2 год.**

Самостійна робота - **76 год.**

9. Рекомендовані джерела:

Основна

1. Омельчук Л.Л. Інструментальні середовища та технології програмування. Лабораторний практикум / Л.Л. Омельчук, Н.Г. Русіна. - Одеса: Айс Принт, 2020. - 175 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/>

<https://classroom.google.com/c/MjI0MTAxMTAxMTU0?cjc=lj146u7>