

Чернівецький національний університет імені Юрія Федьковича

(повне найменування закладу вищої освіти)

Навчально-науковий інститут фізико-технічних і комп'ютерних наук

(назва інституту/факультету)

Кафедра комп'ютерних систем та мереж

(назва кафедри)

СИЛАБУС

навчальної дисципліни

OK15. Програмування. ЧЗ.

(вказіть назву навчальної дисципліни (іноземною, якщо дисципліна викладається іноземною мовою))

Об'єктно-орієнтоване програмування на C#

обов'язкова

(обов'язкова чи вибіркова)

Освітньо-професійна програма – “Програмування мобільних і вбудованих комп'ютерних систем та засобів Інтернету речей”

Спеціальність 123 – Комп'ютерна інженерія

(шифр і назва спеціальності)

Галузь знань 12 – Інформаційні технології

(шифр і назва галузі знань)

Рівень вищої освіти – перший (бакалаврський)

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

Навчально-науковий інститут фізико-технічних і комп'ютерних наук

(назва факультету / інституту, на якому здійснюється підготовка фахівців за вказаною освітньо-професійною програмою)

Мова навчання – українська

(мова, на якій читається дисципліна)

Кількість кредитів: 6

Форми навчальної діяльності: лекції, лабораторні роботи, самостійна робота, курсова робота

Форма підсумкового контролю: іспит, захист курсової роботи

Розробники: Двірничук Костянтин Васильович, асистент кафедри КСМ, канд. фіз.-мат. наук

(вказати авторів (викладач (ів)), їхні посади, наукові ступені, вчені звання)

Профайл викладача (-ів) <https://csn.chnu.edu.ua/>,
<https://csn.chnu.edu.ua/employees/dvirnychuk-kostyantyn-vasylovych/>

Контактний тел. + (38) 0372 50 94 32 (кафедра КСМ) – Двірничук К.В.

E-mail: k.dvirnychuk@chnu.edu.ua

Сторінка курсу в Moodle <https://moodle.chnu.edu.ua/enrol/instances.php?id=3456>

Консультації
on-line: понеділок з 17.00 до 18.00
очні консультації: четвер з 17.00 до 18.00

1. Анотація дисципліни

Курс «Програмування Ч3. Об'єктно-орієнтоване програмування на С#» призначений для надання навиків створення програмних додатків на мові програмування С#, розробки графічних додатків з використанням технологій Windows Forms та .NET MAUI, розробки додатків з використанням реляційних баз даних.

2. Мета навчальної дисципліни: надати студентам систематизовані знання про мету, завдання та принципи об'єктно-зорієнтованого програмування. Навчити студентів створювати програми на мові С# з використанням операторів розгалуження, циклів, масивів, колекцій, кортежів, функцій, навчити їх створювати класи, абстрактні класи, інтерфейси, статичні методи та поля, абстрактні та віртуальні методи, надати їм основні принципи технологій Windows Forms, Entity Framework (ADO.NET), .NET MAUI.

Завдання – на основі отриманих теоретичних знань виробити у студентів уміння розробляти власні програмні системи, використовувати в них всі вивчені програмні конструкції, реляційні та не реляційні БД, виробити, також, у студентів вміння працювати з технологіями Windows Forms, Entity Framework (ADO.NET), .NET MAUI.

3. Пререквізити. Для коректного розуміння і засвоєння матеріалу даного курсу слухачі повинні попередньо пройти курси з: «Програмування Ч1. Основи алгоритмізації та програмування на Python і JavaScript», «Програмування Ч2. Програмування мовою С++».

4. Результати навчання

У результаті вивчення навчальної дисципліни студент повинен

4.1. Знати: мову С#, будувати програми на мові С# з допомогою різних програмних конструкцій, технології програмування Windows Forms, Entity Framework (ADO.NET), .NET MAUI.;

4.2. Вміти: будувати програмні додатки, використовувати для цього різні вивчені до цього різні конструкції та особливості ООП, використовувати технологію Windows Forms, Entity Framework (ADO.NET), .NET MAUI.

4.3. Набути компетентностей:

ЗК – загальних:

ЗК1. Здатність до абстрактного мислення, аналізу і синтезу.

ЗК2. Здатність вчитися і оволодівати сучасними знаннями.

ЗК3. Здатність застосовувати знання у практичних ситуаціях.

ЗК4. Здатність спілкуватися державною мовою як усно, так і письмово.

ЗК6. Навички міжособистісної взаємодії.

ЗК7. Вміння виявляти, ставити та вирішувати проблеми.

ФК – фахових (спеціальних):

ФК2. Здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення.

ФК3. Здатність створювати системне та прикладне програмне забезпечення комп'ютерних систем та мереж.

ФК11. Здатність оформляти отримані робочі результати у вигляді презентацій, науково-технічних звітів.

ФК13. Здатність вирішувати проблеми у галузі комп'ютерних та інформаційних технологій, визначати обмеження цих технологій.

ФК14. Здатність проектувати системи та їхні компоненти з урахуванням усіх аспектів їх життєвого циклу та поставленої задачі, включаючи створення, налаштування, експлуатацію, технічне обслуговування та утилізацію.

ФК15. Здатність аргументувати вибір методів розв'язування спеціалізованих задач, критично оцінювати отримані результати, обґрунтовувати та захищати прийняті рішення.

ПРН – програмних результатів навчання:

ПРН1. Знати і розуміти наукові положення, що лежать в основі функціонування комп'ютерних засобів, систем та мереж.

ПРН3. Знати новітні технології в галузі комп'ютерної інженерії.

ПРН4. Знати та розуміти вплив технічних рішень в суспільному, економічному, соціальному і екологічному контексті.

ПРН5. Мати знання основ економіки та управління проектами.

ПРН6. Вміти застосовувати знання для ідентифікації, формулювання і розв'язування технічних задач спеціальності, використовуючи методи, що є найбільш придатними для досягнення поставлених цілей.

ПРН8. Вміти системно мислити та застосовувати творчі здібності до формування нових ідей.

ПРН10. Вміти розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем, розраховувати, експлуатувати, типове для спеціальності обладнання.

ПРН11. Вміти здійснювати пошук інформації в різних джерелах для розв'язання задач комп'ютерної інженерії.

ПРН12. Вміти ефективно працювати як індивідуально, так і у складі команди.

ПРН13. Вміти ідентифікувати, класифікувати та описувати роботу комп'ютерних систем та їх компонентів.

ПРН14. Вміти поєднувати теорію і практику, а також приймати рішення та виробляти стратегію діяльності для вирішення завдань спеціальності з урахуванням загальнолюдських цінностей, суспільних, державних та виробничих інтересів.

ПРН16. Вміти оцінювати отримані результати та аргументовано захищати прийняті рішення

ПРН18. Використовувати інформаційні технології та для ефективного спілкування на професійному та соціальному рівнях.

ПРН19. Здатність адаптуватись до нових ситуацій, обґрунтовувати, приймати та реалізовувати у межах компетенції рішення.

ПРН20. Усвідомлювати необхідність навчання впродовж усього життя з метою поглиблення набутих та здобуття нових фахових знань, удосконалення креативного мислення.

ПРН21. Якісно виконувати роботу та досягати поставленої мети з дотриманням вимог професійної етики.

5. Опис навчальної дисципліни

5.1. Загальна інформація

Назва навчальної дисципліни <i>OK15 Програмування. ЧЗ. Об'єктно-орієнтоване програмування на C#</i>												
Форма навчання	Рік підготовки	Семестр	Кількість			Кількість годин						Вид підсумкового контролю
			кредитів	годин	змістових модулів	лекції	практичні	семінарські	лабораторні	самостійна робота	індивідуальні завдання	
Денна	2	3	6	180	2	30	-	-	30	120	-	іспит
Заочна												

Примітка. Співвідношення кількості годин аудиторних занять до самостійної і індивідуальної роботи становить: для денної форми навчання – 0,5 (60/120).

5.2. Дидактична карта навчальної дисципліни

Назви змістових модулів і тем	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		
Змістовий модуль 1.														
Тема 1. Основи C#.	7	4		3										
Тема 2. Вступ в ООП.	23	11		12										
Разом за змістовим модулем 1	30	15		15										
Змістовий модуль 2.														
Тема 3. Особливості ООП	64	9		15		40								
Тема 4. Розробка графічних інтерфейсів. Робота з БД	86	6				80								
Разом за змістовим модулем 2	150	15		15		120								
Усього годин	180	30		30		120								

5.3. Теми лабораторних занять

№ з/п	Назва теми
1	Основи C#
2	Методи
3	Класи та об'єкти
4	Доступ до членів класу
5	Успадкування
6	Узагальнення
7	Делегати, лямбди, події
8	Інтерфейси
9	Графічні додатки та бази даних

5.4. Самостійна робота

№ з/п	Назва теми
1	Колекція «Словник»
2	Ітератори
3	Створення моделей в Entity framework
4	Успадкування в Entity framework
5	Відношення між моделями в Entity framework
7	Контейнери в WinForms
8	Панель інструментів та меню в WinForms
9	Створення графічного інтерфейсу в .NET MAUI
10	Компоновка в .NET MAUI
11	Елементи управління в .NET MAUI
12	Курсова робота

6. Форми і методи навчання

Форми навчання – це проблемні й оглядові лекції, лабораторні заняття, заняття із застосуванням комп'ютерної та телекомунікаційної техніки, інтерактивні заняття з навчанням одних студентів іншими, інтегровані заняття, проблемні заняття, відеолекції, відеозаняття і відеоконференції засобами Google Meet, Zoom, заняття з використанням системи електронного навчання Moodle.

Підходи до навчання: використовуються студентоцентрований, проблемно-орієнтований, діяльнісний, комунікативний, професійно-орієнтований, міждисциплінарний підходи.

Для викладання матеріалів з навчальної дисципліни «Комп'ютерні системи» використовуються наступні методи навчання.

6.1. Словесні методи навчання. Навчальна лекція

За допомогою даного методу забезпечується усне викладення матеріалу великими ємністю й складністю логічних побудов, доказів і узагальнень. В ході лекції використовуються прийоми усного викладення інформації, підтримання уваги протягом тривалого часу, активізації мислення студентів, прийоми забезпечення логічного запам'ятовування, переконання, аргументації, доказів, класифікації, систематизації і узагальнення. В залежності від специфіки лекційного матеріалу іноді використовується лекція-діалог.

6.2. Індуктивний метод навчання

Даний метод навчання використовується в рамках лекційних занять, коли матеріал носить, здебільшого, фактичний характер. В рамках лабораторних занять метод застосовується при виконанні технічних задач, коли студенти використовують раніше здобуті теоретичні знання при роботі з конкретними пристроями (комп'ютерами) та програмними продуктами.

6.3. Репродуктивний метод навчання

Даний метод навчання використовується в рамках лекційних і лабораторних занять, а також під час самостійної роботи студентів. Метод передбачає роботу студентів за визначеним алгоритмом. Згідно з методом для виконання завдань студентам надаються методичні вказівки, правила і навчальні приклади.

6.4. Проблемно-пошукові методи навчання

Проблемно-пошукові методи застосовуються в ході проблемного навчання, а саме в процесі виконання лабораторних робіт та індивідуальних науково-дослідних завдань, де під проблемною ситуацією треба вважати невідповідність між тим, що вивчається і вже вивченим. При використанні проблемно-пошукових методів навчання викладач використовує такі прийоми: створює проблемну ситуацію (ставить питання, пропонує задачу, експериментальне завдання), організує колективне обговорення можливих підходів до рішення проблемної ситуації, стимулює висування гіпотез, тощо. Студенти роблять припущення про шляхи вирішення проблемної ситуації, узагальнюють раніше набуті знання, виявляють причини явищ, пояснюють їхнє походження, вибирають найбільш раціональний варіант вирішення проблемної ситуації. Викладач обов'язково керує цим процесом на всіх етапах, а також за допомогою запитань-підказок. Також даний метод використовується при опрацюванні матеріалів в системі дистанційної освіти «Moodle».

6.5. Наочний метод навчання

Наочний метод достатньо важливий для студентів, оскільки забезпечує візуальне подання навчального матеріалу, зокрема, з використанням інформаційно-комунікаційних технологій. При викладанні дисципліни наочний метод навчання поєднується зі словесними методами для представлення інформації у вигляді таблиць, рисунків, схем та діаграм.

7. Система контролю та оцінювання

Засобами оцінювання та демонстрування результатів навчання є

- контрольні роботи;
- стандартизовані тести;
- презентації результатів виконаних завдань та досліджень;
- завдання на лабораторному обладнанні.

Формами поточного контролю рівня знань є усна та письмова відповідь студента при захисті виконаних лабораторних робіт, кількість отриманих балів при виконанні тестового завдання, а також письмова відповідь при написанні модульних контрольних робіт. Формами підсумкового контролю рівня знань є усна та письмова відповідь студента при здачі іспиту.

7.1. Критерії оцінювання результатів навчання з навчальної дисципліни

Критерієм успішного проходження здобувачем освіти підсумкового оцінювання є досягнення ним мінімальних порогових рівнів оцінок за кожним запланованим результатом навчання навчальної дисципліни.

У залежності від характеру відповіді студента кількість балів за кожний вид діяльності може бути визначена за наступними критеріями:

К-ть балів	Критерії оцінки
Мах	Студент дає вичерпну відповідь на поставлене запитання
0,8 · Мах	Студент при відповіді на поставлене запитання припустився незначних неточностей, які не впливають на суть відповіді
0,6 · Мах	Студент при відповіді на поставлене запитання припустився помилок, які виправляє за допомогою викладача; в середньому може дати правильні відповіді на 50% питань теми
0,4 · Мах	Студент при відповіді на поставлене запитання припустився суттєвих помилок, які все ж таки виправляє за допомогою викладача; дає правильні відповіді на 30% питань теми
0,2 · Мах	Студент за допомогою викладача фрагментарно відповідає на запитання, проте не в повній мірі володіє мінімальним рівнем знань з даного питання
0	Характер відповідей дає підставу стверджувати, що студент неправильно зрозумів суть питання чи не знав правильної відповіді, а тому відповідав, припускаючись грубих помилок.

Примітка: за Мах прийнято максимальну оцінку для даного виду діяльності; заокруглення проводиться до одиниць балу.

Шкала та критерії оцінювання: національна та ЄКТС (Європейська кредитна трансферно-накопичувальна система, ECTS)

Оцінка за шкалою ЄКТС	Пояснення	Оцінка за 100-бальною шкалою	Оцінка за національною шкалою
A	відмінно	90 – 100	відмінно
B	дуже добре	80-89	добре
C	добре	70-79	
D	задовільно	60-69	задовільно
E	достатньо	50-59	
FX	(незадовільно) з можливістю повторного складання	35-49	незадовільно
F	(незадовільно) з обов'язковим повторним курсом	1-34	

Розподіл балів, які отримують студенти

Поточне оцінювання (аудиторна та самостійна робота)						Підсумковий контроль (іспит)	Сумарна кількість балів
Змістовий модуль 1			Змістовий модуль 2				
T1	T2	M1	T3	T4	M2		
10	10	10	10	10	10	40	100

T1, T2 ... T4 – теми змістових модулів; M1, M2 – модульні контрольні роботи

7.2. Розподіл максимально можливої кількості балів, які отримують студенти за виконання всіх видів навчальної діяльності

Змістовий модуль 1.

T1. Основи мови C # (Лабораторні роботи 1-3 - 10 балів).

T2. Об'єктно-орієнтоване програмування на C# (Лабораторні роботи 3-6 - 10 балів)

M1. Модульна контрольна робота №1 (10 балів)

Змістовий модуль 2.

T3. Особливості ООП (Лабораторні роботи 7, 8 - 10 балів)

T4. Графічні додатки та бази даних (Лабораторна робота 9 - 10 балів)

M2. Модульна контрольна робота №2 (10 балів)

7.3. Умови зарахування результатів неформальної освіти

Студент, згідно Положення ЧНУ «Про неформальну освіту» може отримати додаткові бали, або бути звільненим від окремих видів роботи з окремих тем, якщо у нього наявні сертифікати про неформальну освіту з проблем, які вивчаються на дисципліні «Комп'ютерні системи».

Також, як виконані види роботи з відповідних тем зараховуються студенту бали за наукові публікації у матеріалах науково-практичних конференцій та фахових чи апробаційних виданнях.

7.4. Політика курсу

Самостійне виконання навчальних завдань, завдань поточного та підсумкового контролю результатів навчання (для осіб з особливими освітніми потребами ця вимога застосовується з урахуванням їхніх індивідуальних потреб і можливостей).

Академічна доброчесність: посилення на джерела інформації у разі використання ідей, розробок, тверджень, відомостей; надання достовірної інформації про результати власної навчальної (наукової, творчої) діяльності, використані методики досліджень і джерела інформації.

Відвідування: Відвідування занять є обов'язковим. Засвоєння пропущеної теми лекції з поважної причини перевіряється під час складання підсумкового контролю. Пропуск лекції з неповажної причини відпрацьовується студентом (співбесіда, реферат тощо). Пропущені лабораторні заняття, незалежно від причини пропуску, студент відпрацьовує згідно з графіком консультацій.

8. Рекомендована література

Базова (основна)

1. Двірничук К.В. Конспект лекцій з навчальної дисципліни «Програмування ЧЗ. Основи ООП» (електронне видання) : навч. посіб. Чернівці, 2022. 200с.
2. Мова програмування С#. Сайт про програмування : веб-сайт. URL: <https://abitap.com/category/c/> (дата звернення: 01.09.2022).
3. Мова програмування С#. Сайт про програмування : веб-сайт. URL: <https://metanit.com/sharp/tutorial/> (дата звернення: 01.09.2022).
4. Посібник по Entity Framework Core. Сайт про програмування : веб-сайт. URL: <https://metanit.com/sharp/entityframeworkcore/> (дата звернення: 01.09.2022).
5. Двірничук К. В., Вацек Д. О. Мова програмування Java: навч. Посібник. Чернівці : Чернівець. нац. ун-т ім. Ю. Федьковича, 2021. 152 с.
6. Двірничук К.В. Конспект лекцій з навчальної дисципліни «Технології проектування програмних систем» (електронне видання) : навч. посіб. Чернівці, 2022. 100с
7. Будай А. Дизайн-патерни – просто, як двері (електронний видання) : навч. посіб. 2019, Львів. 90 с.
8. Книга Head First. Патерни проектування / Ерік Фрімен, Елізабет Робсон, Берт Бейтс, Кеті Сієрра. Харків : Фабула, 2020. 672 с.
9. Бородкіна І., Бородкін Г. Інженерія програмного забезпечення. Посібник для студентів вищих навчальних закладів : навч. посіб. Київ : Центр навчальної літератури, 2018. 204 с.
10. Грицюк Ю. Аналіз вимог до програмного забезпечення : монографія. Львів Львівська політехніка, 2018. 456 с.
11. Мельник Н., Левус Є. Вступ до інженерії програмного забезпечення : навч. посіб. Львів : Львівська політехніка, 2018. 248 с.
12. Патерни проектування. Сайт про програмування : веб-сайт. URL: <https://abitap.com/category/paterny-proektuvannya/> (дата звернення: 01.09.2022).

Допоміжна

1. Патерни проектування. Сайт про програмування : веб-сайт. URL: <https://metanit.com/sharp/patterns/> (дата звернення: 01.09.2022).
2. Троелсен Е., Джепикс Ф. Мова Програмування С# 6.0 і платформа .NET 4.6 : монографія. Книжка лавка, 2019. 800 с.
3. Роберт С. Мартін. Чиста архітектура : монографія. Харків : Фабула, 2019. 368 с.
4. Роберт С. Мартін. Чистий код : монографія. Харків : Фабула, 2019. 416 с.
5. Мельник Р.А. Програмування веб-застосувань (фронт-енд та бек-енд) : навч. посіб. Львів : Львівська політехніка, 2018. 248 с.
6. Посібник по ADO.NET і роботі з базами даних. Сайт про програмування : веб-сайт. URL: <https://metanit.com/sharp/adonet/> (дата звернення: 01.09.2022).
7. Посібник по програмуванню в Windows Forms. Сайт про програмування : веб-сайт. URL: <https://metanit.com/sharp/windowsforms/> (дата звернення: 01.09.2022).

9. Інформаційні ресурси

1. <https://csn.chnu.edu.ua/about-us/ok-rivni/>