

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ЗАТВЕРДЖУЮ

Декан факультету інформаційних технологій



РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Реінжиніринг та оптимізація програмних систем

Назва дисципліни

Галузь знань – 12 Інформаційні технології

Спеціальність – 121 Інженерія програмного забезпечення

Рівень вищої освіти – Другий (магістерський)

Освітньо-професійна програма – Інженерія програмного забезпечення

Обсяг дисципліни – 4 кредита ЄКТС, Шифр дисципліни – ОФП.06

Мова навчання – українська

Статус дисципліни: фахової підготовки

Факультет – Інформаційних технологій

Кафедра – Інженерії програмного забезпечення

Кафедра – Інженерії програмного забезпечення														
Форма здобуття освіти	Курс	Семестр	Загальний обсяг		Кількість годин						Курсовий проєкт*	Курсова робота	Форма семестрового контролю	
					Аудиторні заняття					Самостійна робота, у т.ч. ІРС			Залік	Іспит
			Кредити ЄКТС	Години	Разом	Лекції	Лабораторні роботи	Практичні заняття	Семінарські заняття					
Д	1	1	4	120	34	16	18			86				+
Разом ДФН			4	120	34	16	18			86				1

Робоча програма складена на основі освітньо-професійної програми «Інженерія програмного забезпечення» за спеціальністю 121 «Інженерія програмного забезпечення»

Робоча програма складена  канд. техн. наук, доцент Юрій ФОРКУН
Підпис Науковий ступінь, вчене звання, Ім'я, ПРІЗВИЩЕ

Схвалена на засіданні кафедри Інженерії програмного забезпечення
Назва

Протокол від 28.08 2025 № 1.

Зав. кафедри інженерії програмного забезпечення  Леонід БЕДРАТЮК
Назва Підпис Ім'я, ПРІЗВИЩЕ

Робоча програма розглянута та схвалена вченою радою факультету інформаційних технологій

Голова вченої ради факультету  Тетяна ГОВОРУЩЕНКО

Хмельницький 2025

2 ЛИСТ ПОГОДЖЕННЯ

Посада	Назва факультету	Підпис	Ініціали, прізвище
Завідувач кафедри інженерії програмного забезпечення, доктор, фіз.-мат. наук, проф.	Факультет інформаційних технологій		Леонід БЕДРАТЮК
Гарант освітньо-професійної програми, к.т.н., доцент	Факультет інформаційних технологій		Оксана ЯШИНА

3. Пояснювальна записка

Дисципліна «Реінжиніринг та оптимізація програмних систем» є однією із дисциплін фахової підготовки і займає провідне місце у підготовці здобувачів першого (бакалаврського) рівня вищої освіти, очної денної (далі – денної) форми здобуття вищої освіти, які навчаються за освітньо-професійною програмою «Інженерія програмного забезпечення» в межах спеціальності 121 «Інженерія програмного забезпечення».

Пререквізити: вихідна

Кореквізити: переддипломна практика(ОФП.07); Кваліфікаційна робота (ОФП.08).

Відповідно до освітньої програми дисципліна сприяє забезпеченню:

компетентностей: здатність особи розв'язувати складні задачі і проблеми у певній галузі професійної діяльності або у процесі навчання, що передбачає проведення досліджень та/або здійснення інновацій та характеризується невизначеністю умов і вимог (ІК); Здатність до абстрактного мислення, аналізу та синтезу (ЗК.01); Здатність проводити дослідження на відповідному рівні (ЗК.03); Здатність аналізувати предметні області, формувати, класифікувати вимоги до програмного забезпечення (ФК.01); Здатність розробляти і реалізовувати наукові та/або прикладні проекти у сфері інженерії програмного забезпечення (ФК.02); Здатність проектувати архітектуру програмного забезпечення, моделювати процеси функціонування окремих підсистем і модулів. (ФК.03); Здатність розробляти, аналізувати та застосовувати специфікації, стандарти, правила і рекомендації в сфері інженерії програмного забезпечення. (ФК.05); Здатність ефективно керувати фінансовими, людськими, технічними та іншими проектними ресурсами у сфері інженерії програмного забезпечення. (ФК.06); Здатність розробляти і координувати процеси, етапи та ітерації життєвого циклу програмного забезпечення на основі застосування сучасних моделей, методів та технологій розроблення програмного забезпечення. (ФК.08); Здатність проектувати програмні системи та їх компоненти із використанням поглиблених знань з оптимізації, реінжинірингу програмних систем, управління проектами та якістю з вибором парадигм програмування та застосування машинного навчання. (УК).

програмних результатів навчання: Розробляти і модифікувати архітектуру програмного забезпечення для реалізації вимог замовника (ПРН08); Модифікувати існуючі та розробляти нові алгоритмічні рішення детального проектування програмного забезпечення. (ПРН10); Приймати ефективні організаційно-управлінські рішення в умовах невизначеності та зміни вимог, порівнювати альтернативи, оцінювати ризики. (ПРН12); Конфігурувати програмне забезпечення, керувати його змінами та розробленням програмної документації на всіх етапах життєвого циклу. (ПРН13); Прогнозувати розвиток програмних систем та інформаційних технологій (ПРН14); Прогнозувати розвиток програмних систем та інформаційних технологій (ПРН15); Планувати, організовувати та здійснювати тестування, верифікацію та валідацію програмного забезпечення (ПРН16); Проектувати архітектуру та розробляти програмні компоненти систем із використанням поглиблених знань з оптимізації, реінжинірингу програмних систем, управління проектами та якістю з вибором парадигм програмування із застосуванням машинного навчання, а також їх використання у практичній, експертній та консультативній діяльності (ПРН18).

Мета дисципліни. формування компетентностей, необхідних для абстрактного мислення, аналізу та синтезу на відповідних рівнях оптимізації проектування та розробки програмних систем та їх компонентів, опанування студентами компонентних технологій та інтегрованих середовищ розробки для ефективної оптимізації програмного коду та програмних систем, конкретних виробничих задач інженерії програмного забезпечення.

Предмет дисципліни. Предметом вивчення дисципліни є реінжиніринг та оптимізація програмних систем.

Завдання дисципліни. Формування студентами знань і практичних навичок практичного вирішення прикладних задач з обґрунтуванням реінжинірингу та оптимізації програмних систем

Результати навчання. Студент, який успішно завершив вивчення освітньої компоненти, має: уміло застосовувати інструментарій реінжинірингу та оптимізації програмних систем; визначати, які властивості мають ті чи інші об'єкти разом із заданими для них функцій та операцій; розробляти і модифікувати архітектуру програмного забезпечення для реалізації вимог замовника, модифікуючи існуючі та створюючи нові алгоритмічні рішення детального проектування програмного забезпечення; приймати ефективні організаційно-управлінські рішення в умовах невизначеності та зміни вимог, порівнювати альтернативи, оцінювати ризики; налаштовувати програмне забезпечення, керувати його змінами та розробкою програмної документації на всіх етапах життєвого циклу; прогнозувати розвиток програмних систем та інформаційних технологій; виконувати реінжиніринг програмного забезпечення відповідно до вимог замовника; планувати, організовувати та здійснювати тестування, верифікацію та валідацію програмного забезпечення; застосовувати методи фундаментальних і прикладних дисциплін інженерії програмного забезпечення при проектуванні архітектури та розробці програмних систем з використанням поглиблених знань про оптимізацію, реінжиніринг програмних систем, управління проектами та якістю, вибором парадигм програмування та застосуванням машинного навчання.

4. Структура залікових кредитів дисципліни

Назва теми (теми)	Кількість годин, відведених на:		
	лекції	лабораторні роботи	СРС
Тема 1. Продуктивність, якість та оптимізація програмних систем	2		17
Тема 2. Реінжинірінг програмного забезпечення.	2	4	17
Тема 3. Рефакторинг та оптимізаційні перетворення програмного забезпечення.	4	4	17
Тема 4. Реінжинірінг та рефакторинг баз даних.	4	6	17
Тема 5. Методи оптимізації процесу реінжинірінгу програмних продуктів.	4	4	17
Разом за п'ятий семестр:	16	18	86

5. ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

5.1. Зміст лекційного курсу

Номер лекції	Перелік тем лекцій, їх анотації	Кількість годин
<i>Тема 1. Продуктивність, якість та оптимізація програмних систем</i>		
1	Продуктивність, якість та оптимізація програмних систем. Програмні та організаційні заходи визначення продуктивності та оптимізації ПЗ. Оптимізація етапів процесу розробки, їх закони, правила та умови проектування. Літ.: [1-6]	2
<i>Тема 2. Реінжиніринг програмного забезпечення</i>		
2	Основи реінжинірингу програмного забезпечення. Етапи та процеси реінжинірингу програмного забезпечення. Основні проблеми при плануванні задач реінжинірингу. Оптимізація робіт на етапі проектування програмної системи. Оцінка об'єму робіт на етапі проектування. Зворотня розробка програмного забезпечення. Основні задачі та етапи зворотної розробки. Ризики повторного використання коду програмного забезпечення. Літ.: [1-6]	2
3	Аналіз архітектури програмного забезпечення. Вимоги до проектів. Специфікації завдань. Аналіз знань та алгоритмів. Специфікації завдань. Типи архітектур та їх моделі. Архітектури, засновані на потоках даних. Незалежні компоненти. Віртуальні машини. Репозиторні архітектури. Рівневі архітектури. Додатки зі змішаною архітектурою. Процедура вибору архітектури. Типи кодування. Літ.: [4,12]	2
<i>Тема 3 Рефакторинг та оптимізаційні перетворення програмного забезпечення.</i>		
4	Рефакторинг та оптимізаційні перетворення програмного забезпечення. Принципи рефакторингу. Формат методів рефакторингу. Організація та оптимізація програмного коду. Організація та оптимізація даних. [1,6]	2
5	Організація та оптимізація алгоритмів. Інструментальні засоби проведення рефакторингу. Літ.: [1,6]	2
<i>Тема 4. Реінжиніринг та рефакторинг баз даних.</i>		
6	Реінжиніринг та рефакторинг баз даних. Категорії операцій реінжинірингу та рефакторингу бази даних. Процеси реінжинірингу та рефакторингу бази даних. Операції реінжинірингу та рефакторингу бази даних. Операції реінжинірингу та рефакторингу структури бази даних. Літ.: [2,5,11]	2
<i>Тема 5. Методи оптимізації процесу реінжинірингу програмних продуктів.</i>		
7	Аналіз критеріїв ефективності програмних систем. Методи і моделі визначення ефективності використання програмних систем. Нечіткомножинні підходи й критерії оцінки програмних систем. [1-6,9]	2
8	Оптимізаційні моделі розробки програмного забезпечення з показниками величини проекту. Оптимізація на основі моделі заснованої на кількості помилок. Функція інтенсивності виявлення несправностей. Модель з індексом величини проекту. Процедури стохастичної оптимізації. Літ.: [1-6, 9,13]	2
Разом		16

5.2. Зміст лабораторних робіт

№	Тема лабораторного заняття	Кількість годин
1	Лабораторна робота №1. Вимоги до проектів. Специфікації завдань. Літ.: [1]	2
2	Лабораторна робота №2. Умови оптимізації. Літ.: [1]	2
3	Лабораторна робота №3. Реінженерія архітектури програмного забезпечення. Літ.: [4,12]	2
4	Лабораторна робота №4. Реінженерія архітектури програмного забезпечення. Літ.: [4,12]	2
5	Лабораторна робота № 5. Рефакторинг програмного коду. Літ.: [1,6]	2
6	Лабораторна робота № 6. Оптимізація програмного коду. Літ.: [1,6]	2
7	Лабораторна робота №7. Реінженерія та оптимізація БД. Літ.: [2,5]	2
8	Лабораторна робота №8. Оптимізація взаємодії програмної системи і СКБД. Літ.: [2,5]	2
9	Узагальнююче заняття	2
Разом		18

5.3 Зміст самостійної роботи студентів денної форми навчання

Самостійна робота студентів усіх форм здобуття освіти полягає у систематичному опрацюванні програмного матеріалу з відповідних джерел інформації, підготовці до лабораторних робіт та тестування. Крім цього до послуг студентів сторінка навчальної дисципліни у Модульному середовищі для навчання, де розміщені Робоча програма дисципліни та необхідні документи з її навчально-методичного забезпечення.

Номер тижня	Вид самостійної роботи	Кількість годин
П'ятий семестр		
1	Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи №1.	5
2	Опрацювання лекційного матеріалу. Опрацювання результатів лабораторної роботи №1.	5
3	Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи №2.	5
4	Опрацювання лекційного матеріалу. Опрацювання результатів лабораторної роботи №2.	5
5	Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи №3.	5
6	Опрацювання лекційного матеріалу. Опрацювання результатів лабораторної роботи №3. Підготовка до тестового контролю.	5
7	Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи №4.	5
8	Опрацювання лекційного матеріалу. Опрацювання результатів лабораторної роботи №4.	5
9	Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи №5.	5
10	Опрацювання лекційного матеріалу. Опрацювання результатів лабораторної роботи №5.	5
11	Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи №6.	5
12	Опрацювання лекційного матеріалу. Опрацювання результатів лабораторної роботи №6. Підготовка до контрольної роботи.	5
13	Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи №7.	5
14	Опрацювання лекційного матеріалу. Опрацювання результатів лабораторної роботи №7.	5
15	Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи №8.	5
16	Опрацювання лекційного матеріалу. Підготовка до тесту	6
17	Опрацювання лекційного матеріалу. Опрацювання результатів лабораторної роботи №8.	5
Разом		86

Технології та методи навчання

Процес навчання з дисципліни ґрунтується на використанні традиційних та сучасних методів. Зокрема, лекції проводяться словесними методами, а лабораторні заняття проводяться (з використанням методів комп'ютерного моделювання, методів проєктної діяльності, тренінгових вправ, аналіз проблемних ситуацій, пояснення, з використанням інформаційних технологій та сучасних засобів їх реалізації і мають за мету набуття студентами практичних навиків з реалізації реінжинірингу програмних застосунків та їх програмування, засвоєння студентами основних понять реінжинірингу та використання засобів для їх програмування та оптимізації.

7. Методи контролю

Поточний контроль здійснюється під час аудиторних лабораторних занять, а також у дні проведення контрольних заходів, встановлених робочою програмою і графіком освітнього процесу, в т.ч. з використанням Модульного середовища для навчання. При цьому використовуються такі методи поточного контролю:

- оцінювання результатів захисту лабораторних робіт;
- тестовий контроль засвоєння теоретичного та практичного матеріалу з теми.

При виведенні підсумкової семестрової оцінки враховуються результати як поточного контролю, так і підсумкового контролю, який проводиться з усього матеріалу дисципліни за білетами, попередньо розробленими і затвердженими на засіданні кафедри. Здобувач вищої освіти, який набрав з будь-якого виду навчальної роботи, суму балів нижчу за 60 відсотків від максимального балу, **не допускається** до семестрового контролю, поки не виконає обсяг роботи, передбачений Робочою програмою. Здобувач вищої освіти, який набрав позитивний середньозважений бал (60 відсотків і більше від максимального балу) з усіх видів поточного контролю і не склав іспит, вважається таким, який **має** академічну заборгованість. Ліквідація академічної заборгованості із семестрового контролю здійснюється у період екзаменаційної сесії або за графіком, встановленим деканатом відповідно до «Положення про контроль і оцінювання результатів навчання здобувачів вищої освіти у ХНУ».

8. Політика дисципліни

Політика навчальної дисципліни загалом визначається системою вимог до здобувача вищої освіти, що передбачені чинними положеннями Університету про організацію і навчально-методичне забезпечення освітнього процесу. Зокрема, проходження інструктажу з техніки безпеки; відвідування занять з дисципліни є обов'язковим. За об'єктивних причин (підтверджених документально) теоретичне навчання за погодженням із лектором може відбуватись в он-лайн режимі. Успішне опанування дисципліни і формування фахових компетентностей і програмних результатів навчання передбачає необхідність підготовки до лабораторного заняття (вивчення теоретичного матеріалу з теми роботи, попередню підготовку протоколу роботи, підготовку до усного опитування для допуску до заняття (наведені у Методичних рекомендаціях до лабораторних занять)), активно працювати на занятті, якісно підготувати звіт (відповідно до теми), захистити результати виконаної роботи, брати участь у дискусіях щодо прийнятих конструктивних рішень при виконанні здобувачами лабораторних робіт тощо.

Здобувачі вищої освіти мають дотримуватися встановлених термінів виконання всіх видів навчальної роботи відповідно до робочої програми навчальної дисципліни. Термін захисту лабораторної роботи вважається своєчасним, якщо студент захистив її на наступному після виконання роботи занятті. Пропущене лабораторне заняття студент зобов'язаний відпрацювати у встановлений викладачем термін, але не пізніше, ніж за два тижні до кінця теоретичних занять у семестрі.

Засвоєння студентом теоретичного матеріалу з дисципліни оцінюється за результатами тестування.

У разі виявлення порушення політики академічної доброчесності в будь-яких видах навчальної роботи здобувач вищої освіти отримує незадовільну оцінку і має повторно виконати завдання з відповідної теми (виду роботи), що передбачені робочою програмою. Будь-які форми порушення академічної доброчесності **не допускаються**.

У межах вивчення навчальної дисципліни здобувачам вищої освіти передбачено визнання і зарахування результатів навчання, набутих шляхом неформальної освіти, що розміщені на доступних платформах, які сприяють формування компетентностей і поглибленню результатів навчання, визначених робочою програмою дисципліни, або забезпечують вивчення відповідної теми та/або виду робіт з програми навчальної дисципліни (детальніше у Положенні про порядок визнання та зарахування результатів навчання здобувачів вищої освіти у ХНУ).

9. Оцінювання результатів навчання студентів у семестрі

Оцінювання академічних досягнень здобувача вищої освіти здійснюється відповідно до «Положення про контроль і оцінювання результатів навчання здобувачів вищої освіти у ХНУ». При поточному оцінюванні виконаної здобувачем роботи з кожної структурної одиниці і отриманих ним результатів викладач виставляє йому певну кількість балів із встановлених Робочою програмою для цього виду роботи. При цьому кожна структурна одиниця навчальної роботи може бути зарахована, якщо здобувач набрав не менше 60 відсотків (мінімальний рівень для позитивної оцінки) від максимально можливої суми балів, призначеної структурній одиниці.

При оцінюванні результатів навчання здобувачів вищої освіти з будь-якого виду навчальної роботи

(структурної одиниці) рекомендується використовувати наведені нижче узагальнені критерії:

Таблиця – Критерії оцінювання навчальних досягнень здобувача вищої освіти

Оцінка та рівень досягнення здобувачем запланованих ПРН та сформованих компетентностей	Узагальнений зміст критерія оцінювання
Відмінно (високий)	Здобувач вищої освіти глибоко і у повному обсязі опанував зміст навчального матеріалу, легко в ньому орієнтується і вміло використовує понятійний апарат; уміє пов'язувати теорію з практикою, вирішувати практичні завдання, впевнено висловлювати і обґрунтовувати свої судження. Відмінна оцінка передбачає логічний виклад відповіді мовою викладання (в усній або у письмовій формі), демонструє якісне оформлення роботи і володіння спеціальними приладами та інструментами, прикладними програмами. Здобувач не вагається при видозміні запитання, вміє робити детальні та узагальнюючі висновки, демонструє практичні навички з вирішення фахових завдань. При відповіді допустив дві–три несуттєві <i>похибки</i> .
Добре (середній)	Здобувач вищої освіти виявив повне засвоєння навчального матеріалу, володіє понятійним апаратом, орієнтується у вивченому матеріалі; свідомо використовує теоретичні знання для вирішення практичних задач; виклад відповіді грамотний, але у змісті і формі відповіді можуть мати місце окремі неточності, нечіткі формулювання правил, закономірностей тощо. Відповідь здобувача вищої освіти будується на основі самостійного мислення. Здобувач вищої освіти у відповіді допустив дві–три <i>несуттєві помилки</i> .
Задовільно (достатній)	Здобувач вищої освіти виявив знання основного програмного матеріалу в обсязі, необхідному для подальшого навчання та практичної діяльності за професією, справляється з виконанням практичних завдань, передбачених програмою. Як правило, відповідь здобувача вищої освіти будується на рівні репродуктивного мислення, здобувач вищої освіти має слабкі знання структури навчальної дисципліни, допускає неточності і <i>суттєві помилки</i> у відповіді, вагається при відповіді на видозмінене запитання. Разом з тим, набув навичок, необхідних для виконання нескладних практичних завдань, які відповідають мінімальним критеріям оцінювання і володіє знаннями, що дозволяють йому під керівництвом викладача усунути неточності у відповіді.
Незадовільно (недостатній)	Здобувач вищої освіти виявив розрізнені, безсистемні знання, не вміє виділяти головне і другорядне, допускається помилок у визначенні понять, перекидає їх зміст, хаотично і невпевнено викладає матеріал, не може використовувати знання при вирішенні практичних завдань. Як правило, оцінка «незадовільно» виставляється здобувачеві вищої освіти, який не може продовжити навчання без додаткової роботи з вивчення навчальної дисципліни.

На основі результатів поточного контролю виставляється підсумкова семестрова оцінка.

Структурування дисципліни за видами навчальної роботи і оцінювання результатів навчання студентів денної форми здобуття освіти у семестрі

Аудиторна робота								Контрольні заходи		Семестровий контроль	
_____ семестр											
Лабораторні роботи №:								Тестовий контроль:	Іспит	Разом балів	
1	2	3	4	5	6	7	8				
Кількість балів за вид навчальної роботи (мінімум-максимум)											
3-5	3-5	3-5	3-5	3-5	3-5	3-5	3-5	12-20	24-40	60-100*	
24-40								12-20	24-40		

Виконана й оформлена відповідно до встановлених Методичними рекомендаціями вимог лабораторна робота комплексно оцінюється викладачем при її захисті з урахуванням таких критеріїв: самостійність та правильність виконання; повнота відповіді та знання методик реінжинірингу та оптимізації програмних систем

Результат виконання і захисту здобувачем вищої освіти кожної лабораторної роботи оцінюється відповідно до таблиці Критеріїв оцінювання навчальних досягнень здобувача вищої освіти.

У випадку виявлення здобувачем рівня знань, нижчого ніж 60 відсотків від максимального балу, встановленого Робочою програмою для кожної структурної одиниці, лабораторна робота йому *не зараховується* і для її захисту він має детальніше опрацювати матеріал з теми роботи, методику її виконання, виправити грубі помилки та повторно вийти на її захист у призначений для цього викладачем час.

Оцінювання результатів тестового контролю

Тест, передбачений Робочою програмою, складається із 10 тестових завдань, кожне з яких є рівнозначним.

Відповідно до таблиці структурування видів робіт за тестовий контроль здобувач залежно від кількості правильних відповідей може отримати від 3 до 5 балів.

Розподіл балів в залежності від наданих правильних відповідей на тестові завдання

Кількість правильних відповідей	0-5	6	7	8	9	10
Відсоток правильних відповідей	0-59	60	70	80	90	100
Кількість отриманих балів	0	3	4	5		

На тестування відводиться 30-40 хвилин. Студент проходить тестування в он-лайн режимі у Модульному середовищі для навчання. Також, студент може проходити тестування письмово, записуючи правильні відповіді у талоні відповідей. При отриманні негативної оцінки тест слід перездати до терміну *наступного* контролю.

Оцінювання результатів підсумкового семестрового контролю (іспит)

Освітня програма передбачає підсумковий семестровий контроль з дисципліни у формі іспиту, завданням якого є системне й об'єктивне оцінювання як теоретичної, так і практичної підготовки здобувача з навчальної дисципліни. Складання іспиту відбувається за попередньо розробленими і затвердженими на засіданні кафедри білетами. Відповідно до цього в екзаменаційному білеті пропонується поєднання питань як теоретичного (в т.ч. у тестовій формі), так і практичного характеру.

Таблиця – Оцінювання результатів підсумкового семестрового контролю здобувачів денної форми навчання (40 балів для підсумкового контролю)

Види завдань	Для кожного окремого виду завдань		
	Мінімальний (достатній) бал (задовільно)	Потенційні позитивні бали* (середній бал) (добре)	Максимальний (високий) бал (відмінно)
Теоретичне питання № 1	3	4	5
Теоретичне питання № 2	3	4	5
Практичне завдання (задача 18 балів)	18	24	30
Разом:	24		40

Примітка. *Позитивний бал за іспит, відмінний від мінімального (24 бали) та максимального (40 балів), знаходиться в межах 25-39 балів та розраховується як сума балів за усі структурні елементи (завдання) іспиту.

Для кожного окремого виду завдань підсумкового семестрового контролю застосовуються критерії оцінювання навчальних досягнень здобувача вищої освіти, наведені вище (**Таблиця – Критерії оцінювання навчальних досягнень здобувача вищої освіти**).

Підсумкова семестрова оцінка за інституційною шкалою і шкалою ЄКТС визначається в автоматизованому режимі після внесення викладачем результатів оцінювання у балах з усіх видів навчальної роботи до електронного журналу. Співвідношення інституційної шкали оцінювання і шкали оцінювання ЄКТС наведені нижче у таблиці «Співвідношення».

Семестровий іспит виставляється, якщо загальна сума балів, яку набрав студент з дисципліни за результатами поточного контролю, знаходиться у межах від 60 до 100 балів. При цьому за інституційною шкалою ставиться оцінка «відмінно/добре/задовільно», а за шкалою ЄКТС – буквене позначення оцінки, що відповідає набраній студентом сумі балів відповідно до таблиці «Співвідношення»

Таблиця – Співвідношення інституційної шкали оцінювання і шкали оцінювання ЄКТС

Оцінка ЄКТС	Рейтингова шкала балів	Інституційна оцінка (рівень досягнення здобувачем вищої освіти запланованих результатів навчання з навчальної дисципліни)	
		Залік	Іспит/диференційований залік
A	90-100	Зараховано	Відмінно/Excellent – високий рівень досягнення запланованих результатів навчання з навчальної дисципліни, що свідчить про безумовну готовність здобувача до подальшого навчання та/або професійної діяльності за фахом
B	83-89		Добре/Good – середній (максимально достатній) рівень досягнення запланованих результатів навчання з навчальної дисципліни та готовності до подальшого навчання та/або професійної діяльності за фахом
C	73-82		Задовільно/Satisfactory – Наявні мінімально достатні для подальшого навчання та/або професійної діяльності за фахом результати навчання з навчальної дисципліни
D	66-72		
E	60-65		
FX	40-59	Незараховано	Незадовільно/Fail – Низка запланованих результатів навчання з навчальної дисципліни відсутня. Рівень набутих результатів навчання є недостатнім для подальшого навчання та/або професійної діяльності за фахом
F	0-39		Незадовільно/Fail – Результати навчання відсутні

10. Питання для самоконтролю

1. Ознаки продуктивності програмного забезпечення.
2. Надійність програмних систем.
3. Програмні та організаційні заходи для визначення продуктивності ПЗ.
4. Процеси розробки, їх закони, правила та умови проектування.
5. Реінжиніринг програмних систем
6. Інтеграція програмного забезпечення
7. Етапи та процеси реінженірингу програмного забезпечення
8. Аналіз вимог до проєктів.
9. Етапи та процеси реінжинірингу програмного забезпечення.
10. Основні проблеми при плануванні задач реінжинірингу
10. Специфікації завдань.
11. Типи архітектур. Оптимізація робіт на етапі проектування програмної системи
13. Типи кодування.
12. Мови, типи платформ, кросплатформеність.
13. Оптимальне конструювання модулів програмних систем
14. Продуктивність у суміжних областях ІТ.
15. Оптимізація робіт на етапі проектування програмної системи.
16. Зворотня розробка програмного забезпечення
19. Аналіз архітектури програмного забезпечення
20. Віртуальні машини.
17. Репозиторні архітектури.
18. Рівневі архітектури.
19. Додатки зі змішаною архітектурою.
20. Рефакторинг програмних систем
21. Цілі рефакторингу
22. Рефакторинг та оптимізаційні перетворення програмного забезпечення.
23. Ризики повторного використання коду
24. Інструментальні засоби проведення рефакторингу
25. Організація та оптимізація даних.
26. Організація та оптимізація програмного коду.
27. Організація та оптимізація алгоритмів.
28. Частота оновлення.
29. Частота фіксації помилок.
30. Оптимізація алгебраїчних виразів.
31. Виконання алгебраїчних спрощень.
32. Оптимізація викликів функцій.
33. Оптимізація передачі аргументів.
34. Оптимізація функцій. Оптимізація розподілу змінних.
35. Оптимізація "мертвого" коду. Оптимізація константних умов.
36. Реінжиніринг та оптимізація даних у СКБД.
37. Рефакторинг баз даних
38. Операції реінжинірингу та рефакторингу структури бази даних
43. Старіння процедурного кешу.
39. Неоптимальні індекси.
40. Неоптимальні статистики.
41. Методи визначення ефективності використання програмних систем.
42. моделі визначення ефективності використання програмних систем.
43. Оптимальне конструювання модулів програмних систем.
44. Критерії оцінки якості машинної оптимізації.
45. Методики оцінки якості машинної оптимізації.

Навчально-методичне забезпечення

Навчальний процес з дисципліни повністю і в достатній кількості забезпечений необхідною навчально-методичною літературою. . Зокрема, викладачами кафедри підготовлені і видані такі роботи:

Людино-машинний взаємодія : методичні вказівки до лабораторних робіт для студентів за освітньо-професійною програмою 121 Інженерія програмного забезпечення / Реінжиніринг та оптимізація програмних систем: лабораторний практикум. О.В.Бармак, Ю.В. Форкун, О.М. Яшина. – Хмельницький : ХНУ, 2021. – 58 с.

12. Матеріально-технічне та програмне забезпечення дисципліни (за потреби)

Інформаційна та комп'ютерна підтримка: ПК, планшет, смартфон або інший мобільний пристрій, проектор. Програмне забезпечення: програми Microsoft Office або аналогічні, доступ до мережі Інтернет, робота з презентаціями, VS Code або аналоги , MySQL або аналоги.

. Рекомендована література

Основна:

1. Martin Fowler. Refactoring: Improving the Design of Existing Code (2nd Edition). Addison-Wesley Professional, 2020. 424p. ISBN 978-0134757599

2. Scott Ambler. Refactoring Databases: Evolutionary Database Design. Addison-Wesley Signature Series, 2022. 384 p. ISBN 978-0-321-77451-4
3. Con Mike User Stories Applied for Agile Software Development. Addison-Wesley Professional, 2021. 268 p. ISBN 0-321-20568-5
4. Len Bass, Paul Clements, Rick Kazman. Architektura oprogramowania w praktyce. Wydanie II. Helion. 2021, 464 p. ISBN 978-83-246-3302-9
5. Scott W. Ambler and Pramod J. Sadalage. Refactoring Databases. Addison-Wesley, 2022. 388p. ISBN: 0321293533
6. Martin Robert C. Clean Code Collection. Pearson Education, Inc, 2022. 703 p. ISBN 10 978-013-292847-2, ISBN 13 0-13-292847-7.
7. Bugl Daniel. Learn React Hooks: Build and refactor modern React.js applications using Hooks. Packt Publishing, 2019. 426 p. ISBN 978-1838641443, 1838641440.
8. Alls J. Clean Code in C#: Refactor your legacy C# code base and improve application performance by applying best practices. Packt, 2020. 500 p. ISBN 9781838982973.
9. Vasilik Sylvia Moestl. SQL Practice Problems. CreateSpace Independent Publishing, 2021. 118 p. ISBN 1540422658.
10. Alls J. Clean Code in C#: Refactor your legacy C# code base and improve application performance by applying best practices. Packt, 2020. 500 p. ISBN 9781838982973.
11. Groff J., Weinberg P. SQL: The Complete Reference. Third Edition. McGraw-Hill, 2021. 911 p. ISBN: 978-0071592550. Vasilik Sylvia Moestl. SQL Practice Problems. CreateSpace Independent Publishing, 2021. 118 p. ISBN 1540422658.
12. Sangwan Raghvinder. Software and Systems Architecture in Action. Auerbach Publications, 2024. 232 p.
13. Mitra T. Practical Software Architecture Moving from System Context to Deployment. IBM Press, 2022. 303 p. ISBN: 013376303X, 9780133763034
14. Lemaire M. Refactoring at Scale: Regaining Control of Your Codebase. O'Reilly Media, 2020. 245 p. ISBN 978-1-492-07553-0.
15. McDonough J.E. Automated Unit Testing with ABAP: A Practical Approach. Apress, 2021. 408 p. ISBN 978-1484269503.
16. Довгань Л.С., Мохонько Г.А., Малик І.П. Управління проектами: навчальний посібник до вивчення дисципліни для магістрів галузі знань 07 Управління та адміністрування К.: КПІ ім. Ігоря Сікорського, 420 с. <http://ela.kpi.ua/handle/123456789/19481>.
17. 3. О.В. Бармак, В.В. Кудрявцев, Ю.В. Форкун, О.М. Яшина. Підхід до аналізу програмного коду з використанням метрик Холстеда // Вісник Хмельницького національного університету. Технічні науки, 2021. № 3. с. 25-29.

Додаткова:

1. Sridhar B. Model Driven Software Engineering in the Mobile Era with an Emphasis on Security / International Journal of Electronics Communication and Computer Engineering. Vol. 6, Iss. 6. 2022. P. 619-623.
2. Сазерленд Д. Scrum. Навчись робити вдвічі більше за менший час / Д. Сазерленд; пер. з англ. Я. Лебеденко. – Х.: Книжковий Клуб «Клуб Сімейного Дозвілля», 2021.– 281с.
3. Рач В.А. Кількісні методи ресурсно-часового оцінювання в гнучкій методології розробки програмного забезпечення. Управління проектами та розвиток виробництва. http://nbuv.gov.ua/UJRN/Uprv_2021_1_7

14. Інформаційні ресурси:

1. Модульне середовище для навчання. URL: <https://msn.khmnmu.edu.ua/course/view.php?id=6767>
2. Електронна бібліотека університету. URL: http://lib.khmnmu.edu.ua/asp/php_f/p1age_lib.php
3. Репозитарій ХНУ. URL: <http://elar.khmnmu.edu.ua/home/>

РЕІНЖІНІРИНГ ТА ОПТИМІЗАЦІЯ ПРОГРАМНИХ СИСТЕМ

Тип дисципліни	Обов'язкова
Рівень вищої освіти	Другий (магістерський)
Мова навчання	Українська
Семестр	перший
Кількість встановлених кредитів ЄКТС	5,0
Форми навчання, для яких викладається дисципліна	Денна

Результати навчання. Студент, який успішно завершив вивчення освітньої компоненти, має: уміло застосовувати інструментарій реінжинірингу та оптимізації програмних систем; визначати, які властивості мають ті чи інші об'єкти разом із заданими для них функцій та операцій; розробляти і модифікувати архітектуру програмного забезпечення для реалізації вимог замовника, модифікуючи існуючі та створюючи нові алгоритмічні рішення детального проектування програмного забезпечення; приймати ефективні організаційно-управлінські рішення в умовах невизначеності та зміни вимог, порівнювати альтернативи, оцінювати ризики; налаштовувати програмне забезпечення, керувати його змінами та розробкою програмної документації на всіх етапах життєвого циклу; прогнозувати розвиток програмних систем та інформаційних технологій; виконувати реінжиніринг програмного забезпечення відповідно до вимог замовника; планувати, організовувати та здійснювати тестування, верифікацію та валідацію програмного забезпечення; застосовувати методи фундаментальних і прикладних дисциплін інженерії програмного забезпечення при проектуванні архітектури та розробці програмних систем з використанням поглиблених знань про оптимізацію, реінжиніринг програмних систем, управління проектами та якістю, вибором парадигм програмування та застосуванням машинного навчання.

Зміст навчальної дисципліни. Продуктивність та якість програмного забезпечення. Реінженерія програмного забезпечення. Рефакторинг та оптимізаційні перетворення програмного забезпечення. Реінжинерія та рефакторинг баз даних. Методи оптимізації процесу реінжинерії програмних продуктів.

Пререквізити: вихідна

Кореквізити: переддипломна практика (ОФП.07); Кваліфікаційна робота (ОФП.08).

Запланована навчальна діяльність: Мінімальний обсяг навчальних занять в одному кредиті ЄКТС навчальної дисципліни для другого (магістерського) рівня вищої освіти за денною формою здобуття освіти становить 8 годин на 1 кредит ЄКТС.

Форми (методи) навчання: Процес навчання з дисципліни ґрунтується на використанні традиційних та сучасних методів. Зокрема, лекції проводяться словесними методами, а лабораторні заняття проводяться (з використанням методів комп'ютерного моделювання, методів проектної діяльності, тренінгових вправ, аналіз проблемних ситуацій, пояснення, з використанням інформаційних технологій та сучасних засобів їх реалізації і мають за мету набуття студентами практичних навиків з реалізації реінжинірингу програмних застосунків та їх програмування, засвоєння студентами основних понять реінжинірингу та використання засобів для їх програмування та оптимізації.)

Форми оцінювання результатів навчання: оцінювання лабораторних робіт; тестування.

Вид семестрового контролю: іспит – 1 семестр.

Навчальні ресурси:

1. Martin Fowler. Refactoring: Improving the Design of Existing Code (2nd Edition). Addison-Wesley Professional, 2020. 424p. ISBN 978-0134757599
2. Scott Ambler. Refactoring Databases: Evolutionary Database Design. Addison-Wesley Signature Series, 2022. 384 p. ISBN 978-0-321-77451-4
3. Con Mike User Stories Applied for Agile Software Development. Addison-Wesley Professional, 2021. 268 p. ISBN 0-321-20568-5
4. Len Bass, Paul Clements, Rick Kazman. Architektura oprogramowania w praktyce. Wydanie II. Helion. 2021, 464 p. ISBN 978-83-246-3302-9
5. Модульне середовище для навчання. URL: <https://msn.khmnmu.edu.ua/course/view.php?id=6767>
6. Електронна бібліотека університету. URL: http://lib.khmnmu.edu.ua/asp/php_f/p1age_lib.php

Викладачі: канд. техн. наук, доцент кафедри ІІЗ Форкун Ю.В.