Національний технічний університет України
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Кафедра програмного забезпечення комп'ютерних систем

МЕТОДИ ПОБУДОВИ ПРОГРАМНИХ ЗАСОБІВ ДЛЯ МАТРИЧНИХ ОБЧИСЛЕНЬ

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	***Перший (бакалаврський)***
Галузь знань	*12 Інформаційні технології*
Спеціальність	*121 Інженерія програмного забезпечення*
Освітня програма	*Інженерія програмного забезпечення мультимедійних та інформаційно-пошукових систем*
Статус дисципліни	*Вибіркова*
Форма навчання	*Очна (денна)*
Рік підготовки, семестр	*4 рік навчання, 8 семестр*
Обсяг дисципліни	*Лекції: 36 год., лабораторні заняття: 18 год, самостійна робота: 66 год.*
Семестровий контроль/ контрольні заходи	*Залік, модульна контрольна робота, календарний контроль*
Розклад занять	*Згідно розкладу на весняний семестр поточного навчального року (rozklad.kpi.ua)*
Мова викладання	*Українська*
Інформація про керівника курсу / викладачів	Лектор: *к.т.н., доцент, Онай Микола Володимирович, onay@pzks.fpm.kpi.ua* Комп'ютерний практикум: *к.т.н., доцент, Онай Микола Володимирович, onay@pzks.fpm.kpi.ua*

Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчання та результати навчання

Вивчення дисципліни «Методи побудови програмних засобів для матричних обчислень» дозволяє сформувати у здобувачів вищої освіти компетенції, необхідні для розв'язання складних задач професійної діяльності, пов'язаної із розробленням програмних систем для розв'язання типових задач, що виникають при математичному моделюванні природних процесів та явищ.

Метою *вивчення дисципліни «Методи побудови програмних засобів для матричних обчислень» є формування у здобувачів освіти здатності проводити науково-інноваційну діяльність, пов'язану із розробленням програмних систем для виконання матричних обчислень.*

Предметом *дисципліни «Методи побудови програмних засобів для матричних обчислень» є методи виконання програмних обчислень над матрицями, що мають спеціальну структуру та велику розмірність.*

Вивчення дисципліни «Методи побудови програмних засобів для матричних обчислень» підсилює формування у здобувачів освіти ***фахових компетентностей (ФК)****, необхідних для розв'язання практичних задач професійної діяльності:*
ФК18 *Здатність розробляти методи чисельного розв'язання математичних задач з використанням програмних засобів.*
ФК20 *Здатність застосовувати набуті фундаментальні математичні знання для розроблення методів обчислень при створенні мультимедійних та інформаційно-пошукових систем.*

*Вивчення дисципліни «Методи побудови програмних засобів для матричних обчислень» сприяє формуванню у студентів наступних **програмних результатів навчання** (ПРН) за освітньою програмою:*
ПРН01 *Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.*

ПРН25 *Знати і вміти використовувати фундаментальний математичний інструментарій при побудові алгоритмів та розробленні сучасного програмного забезпечення.*

ПРН26 *Вміти розробляти та використовувати методи і алгоритми наближеного розв'язання математичних задач при проєктуванні мультимедійних та інформаційно-пошукових систем.*

2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

Успішному вивченню дисципліни «Методи побудови програмних засобів для матричних обчислень» передує вивчення дисципліни «Алгоритмічне забезпечення мультимедійних та інформаційно-пошукових систем» навчального плану підготовки бакалаврів за спеціальністю 121 Інженерія програмного забезпечення.

Отримані в результаті засвоєння дисципліни «Методи побудови програмних засобів для матричних обчислень» теоретичні знання та практичні уміння можуть бути корисні для проведення наукових досліджень та при виконанні бакалаврської кваліфікаційної роботи.

3. Зміст навчальної дисципліни

Дисципліна «Методи побудови програмних засобів для матричних обчислень» передбачає вивчення тем:

Тема 1. Методи розв'язання матричних проблем власних значень

Тема 2. Сингулярне розкладення прямокутних матриць

Тема 3. Програмні методи розв'язання систем нелінійних рівнянь

Тема 4. Апроксимація функцій

Модульна контрольна робота

Залік

4. Навчальні матеріали та ресурси

Базова література:

1. Андруник В.А. Чисельні методи в комп'ютерних науках: навчальний посібник / Андруник В.А., Висоцька В.А., Пасічник В.В., Чирун Л.Б., Чирун Л.В. Ч // Том 2 за ред. В.В. Пасічника – Львів: Видавництво «Новий Світ», 2020. – 536 с
Використати для вивчення принципів розв'язання математичних задач, що виникають при побудові математичних моделей. Матеріали знаходяться у вільному доступі в Інтернеті.

Додаткова література:

2. William H. Press Numerical Recipes in C. The Art of Scientific Computing / William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery // Cambridge University Press, 2002. – 1018 p.
Використати для опанування практичних навичок з дисципліни.
3. Walter Gautschi Numerical Analysis [Електронний ресурс], 2012. Режим доступу: http://www.ikiu.ac.ir/public-files/profiles/items/090ad_1410599906.pdf
Використати для опанування теоретичним матеріалом з дисципліни.

4. Singular Value Decomposition [Електронний ресурс] Режим доступу: https://www.cs.cmu.edu/~venkatg/teaching/CStheory-infoage/book-chapter-4.pdf
Використати для вивчення загальної структури сингулярного розкладення матриці. Матеріали знаходяться у вільному доступі в Інтернеті.

5. Courtney Remani Numerical Methods for Solving Systems of Nonlinear Equations [Електронний ресурс], 2013.
Режим доступу: https://www.lakeheadu.ca/sites/default/files/uploads/77/docs/RemaniFinal.pdf
Використати для вивчення принципів розв'язання нелінійних рівнянь. Матеріали знаходяться у вільному доступі в Інтернеті.

6. McDonough J. M. Computational Numerical Analysis [Електронний ресурс], 2007. Режим доступу: http://web.engr.uky.edu/~acfd/egr537-lctrs.pdf

Використати для опанування практичних навичок з дисципліни.

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

№ з/п	Тип навчального заняття	Опис навчального заняття
		Тема 1. Методи розв'язання матричних проблем власних значень
1	*Лекція 1. Найпростіші методи вирішення часткової проблеми власних значень*	*Власні пари матриць та їх найпростіші властивості. Задачі, що зводяться до алгебраїчної проблеми власних значень. Відношення Релея. Степеневий метод вирішення часткової проблеми власних значень. Метод слідів.* Δ^2*-процес Ейткена.* *Завдання на СРС: розглянути приклади розв'язання часткової проблеми власних значень степеневим методом та його модифікаціями та п. 6, № 1.*
2	*Лекція 2. Поліпшені методи вирішення часткової проблеми власних значень*	*Метод скалярних добутків вирішення часткової проблеми власних значень. Метод часток Релея. Метод зворотних ітерацій. Метод зворотних ітерацій зі зсувом. Метод зворотних ітерацій з відношеннями Релея. Визначення порядку збіжності поліпшених методів вирішення часткової проблеми валсних значень.* *Завдання на СРС: розглянути приклади розв'язання часткової проблеми власних значень методом скалярних добутків, методом зворотних ітерацій та їх модифікаціями та п. 6, № 2.*
3	*Комп'ютерний практикум 1*	*Завдання: Розробити програмну систему для розв'язання матричних проблем власних значень.*
4	*Лекція 3. Метод обертань розв'язання повної проблеми власних значень або метод Якобі*	*Перетворення подібності. Класичний метод Якобі. Циклічний метод Якобі з бар'єром. Аналіз стратегій вибору ключового елемента в методі Якобі. Критерій зупинки ітераційного процесу в методі Якобі та його модифікаціях.*

		Завдання на СРС: розглянути приклади розв'язання повної проблеми власних значень різноманітними алгоритмами, що реалізують методом Якобі та п. 6, № 3.
5	Лекція 4. LU-метод розв'язання повної несиметричної проблеми власних значень та його модифікації	Застосування LU-розкладення матриці для задачі знаходження всіх власних пар матриці. Матриці Хесенберга. Завдання на СРС: побудувати LU-алгоритм для знаходження власних пар симетричних додатно визначених матриць на базі U^TU - та LL^T-розкладення Холецького та п. 6, № 4.
6	Лекція 5. QR-метод розв'язання повної несиметричної проблеми власних значень та його модифікації	Застосування QR-розкладення матриці для розв'язання несиметричних спектральних алгебраїчних задач. Перетворення Гівенса. Модифікації класичного QR-алгоритму. Поняття QB-факторизації. Зсуви за Уілкінсоном. Теорема про неявне Q. Двокроковий QR-алгоритм Френсіса. Завдання на СРС: вивести формули для побудови матриці $\mathbf{H}$, що використовується у QB-факторизації та п. 6, № 5.
7	Комп'ютерний практикум 2	Завдання: Розробити програмну систему для розв'язання матричних проблем власних значень.
	Тема 2. Сингулярне розкладення прямокутних матриць	
8	Лекція 6. Основні терміни та поняття, загальний підхід до побудови сингулярного розкладення	Прикладні задачі в яких виникає необхідність виконувати сингулярне розкладення прямокутної матриці. Узагальнення поняття власного числа – сингулярне число. Праві та ліві сингулярні вектори. Етапи знаходження сингулярного розкладення прямокутної матриці. Завдання на СРС: дослідити взаємозв'язок між правими та лівими сингулярними векторами та п. 6, № 6.
9	Лекція 7. Дводіагоналізація квадратної матриці	Застосування QR-розкладення та перетворень Гівенса для дводіагоналізації квадратної матриці. Процес переслідування. Процедура вичерпування. Завдання на СРС: вивести формули для побудови матриць Хаусхолдера $\mathbf{H}_1^{(i)}$ та $\mathbf{H}_2^{(j)}$, що використовуються для приведення квадратної матриці $\mathbf{A}$ до верхнього та нижнього дводіагонального вигляду та п. 6, №7.
10	Комп'ютерний практикум 3	Завдання: Розробити програмні компоненти для виконання сингулярного розкладення.
11	Лекція 8. Сингулярне розкладення дводіагональної матриці	Модифікація QR-алгоритму з неявними зсувами вирішення повної проблеми власних значень для виконання сингулярного розкладення дводіагональної матриці. Особливості застосування перетворення Гівесна та Хаусхолдера для отримання сингулярного розкладення.

		Завдання на СРС: вивести формули для обчислення параметра зсуву τ та формули для процесу переслідування у нижній дводіагональній матриці та п. 6, № 8.
12	*Лекція 9. Приклади застосування сингулярного розкладення в інженерних задачах дводіагональної матриці*	*Обчислення абсолютної величини визначника квадратної матриці. Спектральне число обумовленості. Знаходження розв'язку однорідної СЛАР. Псевдорозв'язок СЛАР. Нормальний псевдорозв'язок СЛАР. Матриця Мура-Пенроуза. Псевдообернена матриця.* *Завдання на СРС: розглянути приклади знаходження рангу матриці, псевдооберненої матриці та числа обумовленості прямокутної матриці та п. 6, № 9.*
13	*Комп'ютерний практикум 4*	*Завдання: Розробити програмні компоненти для демонстрації прикладів застосування сингулярного розкладення прямокутних матриць.*
14	*Лекція 10*	*Модульна контрольна робота. Частина 1*
	Тема 3. Методи розв'язання систем нелінійних рівнянь	
15	*Лекція 11. Метод простих ітерацій*	*Геометричний зміст системи нелінійних рівнянь. Основні етапи розв'язання системи нелінійних рівнянь. Задача про нерухому точку нелінійного відображення. Метод протсих ітерацій розв'язання системи нелінійних рівнянь. Достатня умова збіжності метода простих ітерацій.* *Завдання на СРС: проаналізувати умови теореми про збіжність методу простих ітерацій для розв'язання систем нелінійних рівнянь та п. 6, № 10.*
16	*Лекція 12. Метод Ньютона розв'язання систем нелінійних рівнянь та його спрощений варіант*	*Метод Ньютона розв'язання системи нелінійних рівнянь в неявному вигляді. Явний метод Ньютона розв'язання системи нелінійних рівнянь. Спрощений метод Ньютона розв'язання системи нелінійних рівнянь.* *Завдання на СРС: розглянути приклади знаходження розв'язку СНР методом Ньютона та спрощеним методом Ньютона та п. 6, № 11.*
17	*Комп'ютерний практикум 5*	*Завдання: Розробити програмні модулі для розв'язання систем нелінійних рівнянь.*
18	*Лекція 13. Модифікації метода Ньютона розв'язання систем нелінійних рівнянь*	*Двоступеневий метод Ньютона розв'язання системи нелінійних рівнянь. Методи Ньютона з послідовною апроксимацією обернених матриць. Класичний метод січних.* *Завдання на СРС: розглянути приклади розв'язання СНР за допомогою модифікацій метода Ньютона та п. 6, № 12.*
19	*Лекція 14. Метод січних Бройдена, метод Брауна та зв'язок задачі розв'язання системи*	*Співвідношення січних. Метод січних Бройдена. Метод Брауна. Загальний вигляд оптимізаційних задач. Формулювання задачі розв'язання системи нелінійних рівнянь як оптимізаційної задачі.*

	нелінійних рівнянь із задачею оптимізації	Завдання на СРС: розглянути приклади розв'язання СНР методом січних Бройдена та методом Брауна та п. 6, № 13.
20	Комп'ютерний практикум 6	Завдання: Розробити програмні модулі для розв'язання систем нелінійних рівнянь.
	Тема 4. Апроксимація функцій	
21	Лекція 15. Скінченно-різницеві інтерполяційні формули	Поняття скінченної різниці. Порядок скінченної різниці. Зв'язок скінченних різниць з похідними. Діагональна таблиця скінченних різниць. Перший інтерполяційний многочлен Ньютона. Другий інтерполяційний многочлен Ньютона. Інтерполяційні формули Ньютона. Центральні інтерполяційні формули. Перший інтерполяційний многочлен Гаусса. Другий інтерполяційний многочлен Гаусса. Інтерполяційна формула Стірлінга. Інтерполяційна формула Бесселя. Завдання на СРС: п. 6, № 14.
22	Комп'ютерний практикум 7	Завдання: Розробити програмні компоненти для знаходження наближених значень табличних функцій.
23	Лекція 16. Принципи побудови інтерполяційних формул для нерівновіддалених вузлів	Поняття розділеної різниці. Порядок розділеної різниці. Таблиця розділених різниць. Перша інтерполяційна формула Ньютона для нерівновіддалених вузлів. Друга інтерполяційна формула Ньютона для нерівновіддалених вузлів. Завдання на СРС: п. 6, № 15.
24	Комп'ютерний практикум 8	Завдання: Розробити програмні компоненти для візуалізації графіка апроксимованої функції.
25	Лекція 17. Принципи побудови інтерполяційних формул для нерівновіддалених вузлів	Поняття розділеної різниці. Порядок розділеної різниці. Таблиця розділених різниць. Перша інтерполяційна формула Ньютона для нерівновіддалених вузлів. Друга інтерполяційна формула Ньютона для нерівновіддалених вузлів. Завдання на СРС: п. 6, № 16.
26	Комп'ютерний практикум 9	Підсумки
27	Лекція 18	Модульна контрольна робота. Частина 2

6. Самостійна робота студента/аспіранта

Дисципліна «Методи побудови програмних засобів для матричних обчислень» ґрунтується на самостійній підготовці до аудиторних занять на теоретичні та практичні теми.

№ з/п	Назва теми, що виноситься на самостійне опрацювання	Кількість годин	Література
1	Підготовка до лекцій	16	1-6
2	Підготовка до комп'ютерного практикуму	27	1-6
3	Підготовка до модульної контрольної роботи. Частина 1	9	1-6
4	Підготовка до модульної контрольної роботи. Частина 2	9	1-6

5	*Підготовка до заліку*	*5*	*1-6*

Політика та контроль

7. Політика навчальної дисципліни (освітнього компонента)

Відвідування занять. *Відсутність на аудиторному занятті не передбачає нарахування штрафних балів, оскільки фінальний рейтинговий бал студента формується виключно на основі оцінювання результатів навчання. Разом з тим, обговорення результатів виконання тематичних завдань, а також презентація / публічний виступ та участь у обговореннях та доповнення на семінарах оцінюватимуться під час аудиторних занять. Для активної участі у роботі семінару студент готується за рекомендованою викладачем до певного семінарського заняття літературою. Участь у роботі семінару також передбачає підготування доповідей та співдоповідей у межах усіх занять.*

Пропущені контрольні заходи оцінювання. *Кожен студент має право відпрацювати пропущені з поважної причини (лікарняний, мобільність тощо) заняття за рахунок самостійної роботи. Детальніше за посиланням: https://kpi.ua/files/n3277.pdf.*

Процедура оскарження результатів контрольних заходів оцінювання. *Студент може підняти будь-яке питання, яке стосується процедури контрольних заходів та очікувати, що воно буде розглянуто згідно із наперед визначеними процедурами. Студенти мають право аргументовано оскаржити результати контрольних заходів, пояснивши з яким критерієм не погоджуються відповідно до оціночного. Календарний контроль проводиться з метою підвищення якості навчання студентів та моніторингу виконання студентом вимог силабусу.*

Академічна доброчесність. *Політика та принципи академічної доброчесності визначені у розділі 3 Кодексу честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». Детальніше: https://kpi.ua/code .*

Норми етичної поведінки. *Норми етичної поведінки студентів і працівників визначені у розділі 2 Кодексу честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». Детальніше: https://kpi.ua/code .*

Інклюзивне навчання. *Засвоєння знань та умінь в ході вивчення дисципліни «Науково-дослідна діяльність у комп'ютерній інженерії» може бути доступним для більшості осіб з особливими освітніми потребами, окрім здобувачів з серйозними вадами зору, які не дозволяють виконувати завдання за допомогою персональних комп'ютерів, ноутбуків та/або інших технічних засобів.*

Навчання іноземною мовою. *У ході виконання завдань студентам може бути рекомендовано звернутися до англомовних джерел. Призначення заохочувальних та штрафних балів Відповідно до Положення про систему оцінювання результатів навчання сума всіх заохочувальних балів не може перевищувати 10% рейтингової шкали оцінювання.*

Всі студенти повинні відвідувати лекційні та практичні заняття, на яких потрібно активно працювати над засвоєнням навчального матеріалу. За об'єктивних причин (наприклад - хвороба, міжнародне стажування) навчання може відбуватись в он-лайн формі індивідуально за погодженням із керівником курсу.

Політика щодо дедлайнів та перескладання:

Роботи, які здаються із порушенням термінів без поважних причин, оцінюються на нижчу оцінку. Перескладання модулів відбувається із дозволу деканату за наявності поважних причин (наприклад, лікарняний).

Політика щодо академічної доброчесності:

Усі письмові роботи перевіряються на наявність плагіату і допускаються до захисту із коректними текстовими запозиченнями не більше 20%. Списування під час контрольних робіт заборонені (в т. ч. із використанням мобільних пристроїв).

8. Види контролю та рейтингова система оцінювання результатів навчання (РСО)

Протягом семестру студентами виконують 8 комп'ютерних практикумів. Максимальна кількість балів за кожний комп'ютерний практикум: 6 балів.

Бали нараховуються за:
- якість виконання комп'ютерного практикуму: 0-2 бали;
- відповідь на теоретичні запитання під час захисту комп'ютерного практикуму: 0-2 бали;
- своєчасне представлення роботи до захисту: 0-2 бали.

Критерії оцінювання якості виконання:
2 бали – робота виконана якісно, в повному обсязі;
1 бал – робота виконана в повному обсязі, але містить незначні помилки;
0 балів – робота виконана не в повному обсязі, або містить суттєві помилки.

Критерії оцінювання відповіді:
2 бали – відповідь повна, добре аргументована;
1 бал – в цілому відповідь правильна, але має недоліки або незначні помилки;
0 балів – немає відповіді або відповідь неправильна.

Критерії оцінювання своєчасності представлення роботи до захисту:
2 бали – робота представлена до захисту не пізніше вказаного терміну;
0 балів – робота представлена до захисту пізніше вказаного терміну.

Максимальна кількість балів за виконання та захист комп'ютерних практикумів:
6 балів × 8 комп. практ. = 48 балів.

Завдання на ***модульну контрольну роботу*** *складається з 3 запитань – 1 теоретичного та 2 практичних. Відповідь на теоретичне запитання оцінюється 6 балами, а відповідь на практичне запитання оцінюється 10 балами.*

Критерії оцінювання кожного теоретичного запитання контрольної роботи:
6 балів – відповідь правильна, повна, добре аргументована;
5 балів – відповідь правильна, розгорнута, але не дуже добре аргументована;
4 балів – в цілому відповідь правильна, але має недоліки;
3 балів – у відповіді є незначні помилки;
1-2 бали – у відповіді є суттєві помилки;
0 балів – немає відповіді або відповідь неправильна.

Критерії оцінювання практичного запитання контрольної роботи:
9-10 балів – відповідь правильна, розрахунки виконані у повному обсязі;
7-8 балів – відповідь правильна, але не дуже добре підкріплена розрахунками;
5-6 балів – в цілому відповідь правильна, але має недоліки;
3-4 балів – у відповіді є незначні помилки;
1-2 бали – у відповіді є суттєві помилки;
0 балів – немає відповіді або відповідь неправильна.

Максимальна кількість балів за модульну контрольну роботу:
*2 роботи * (6 балів × 1 теоретичне запитання + 10 бали × 2 практичних запитання) = 52 бали.*

Рейтингова шкала з дисципліни дорівнює:
$R_c = R_{ком.практ} + R_{МКР}$ *= 48 балів + 52 балів = 100 балів.*

Календарний контроль: провадиться двічі на семестр як моніторинг поточного стану виконання вимог силабусу.
На першій атестації (8-й тиждень) студент отримує «зараховано», якщо його поточний рейтинг не менше 50 % від максимальної кількості балів (20 балів), яку може отримати студент до першої атестації.
На другій атестації (14-й тиждень) студент отримує «зараховано», якщо його поточний рейтинг не менше 50 % від максимальної кількості балів (35 балів), яку може отримати студент до другої атестації.

Семестровий контроль: ***залік***

Умови допуску до семестрового контролю:
За семестрового рейтингу (R_C) не менше 60 балів та зарахуванні усіх робіт комп'ютерного практикуму аспірант отримує залік «автоматом» відповідно до таблиці (Таблиця відповідності рейтингових балів оцінкам за університетською шкалою). В іншому разі він має виконати залікову контрольну роботу.

Необхідною умовою допуску до виконання залікової контрольної роботи є виконання і захист комп'ютерного практикуму.

Аспірант може спробувати підвищити свою оцінку шляхом написання залікової контрольної роботи, при цьому його бали, отримані за семестр, анулюються («жорстка» система оцінювання).

Склад та критерії оцінювання залікової контрольної роботи:

Завдання на ***залікову контрольну роботу*** *складається з 4 запитань – 2 теоретичних та 2 практичних. Відповідь на кожне теоретичне та практичне запитання оцінюється 25 балами.*

Критерії оцінювання кожного теоретичного запитання контрольної роботи:
24-25 балів – відповідь правильна, повна, добре аргументована;
21-23 бали – відповідь правильна, розгорнута, але не дуже добре аргументована;
17-20 балів – в цілому відповідь правильна, але має недоліки;
12-16 балів – у відповіді є незначні помилки;
1-11 бали – у відповіді є суттєві помилки;
0 балів – немає відповіді або відповідь неправильна.

Критерії оцінювання практичного запитання контрольної роботи:
24-25 балів – відповідь правильна, розрахунки виконані у повному обсязі;
21-23 бали – відповідь правильна, але не дуже добре підкріплена розрахунками;
17-20 балів – в цілому відповідь правильна, але має недоліки;
12-16 балів – у відповіді є незначні помилки;
1-11 бали – у відповіді є суттєві помилки;
0 балів – немає відповіді або відповідь неправильна.

Максимальна кількість балів за модульну контрольну роботу:
25 балів × 2 теоретичних запитання + 25 балів × 2 практичних запитання = 100 балів.

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

Кількість балів	*Оцінка*
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно
Не виконані умови допуску	Не допущено

9. Додаткова інформація з дисципліни (освітнього компонента)

Перелік запитань, які виносяться на модульний та семестровий контроль у Додатку 1

Робочу програму навчальної дисципліни (силабус):

Складено к.т.н., доц., Онай М.В.

Ухвалено кафедрою ПЗКС (протокол №8 від 25.01.2023)

Погоджено Методичною комісією факультету прикладної математики (протокол №6 від 27.01.2023)

Додаток 1. Перелік запитань, які виносяться на модульний та семестровий контроль

Модульна контрольна робота №1 включає такі типи задач:

1. *Виконати сингулярне розкладення матриці (на першому етапі звести матрицю до верхньої дводіагональної форми).*
2. *Виконати сингулярне розкладення матриці (на першому етапі звести матрицю до нижньої дводіагональної форми)*
3. *З точністю* $\varepsilon \le 10^{-2}$ *алгоритмом №0 реалізації методу Якобі знайти всі власні числа та власні вектори матриці. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
4. *З точністю* $\varepsilon \le 10^{-2}$ *алгоритмом №1 реалізації методу Якобі знайти всі власні числа та власні вектори матриці. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
5. *З точністю* $\varepsilon \le 10^{-2}$ *алгоритмом №2 реалізації методу Якобі знайти всі власні числа та власні вектори матриці. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
6. *LU-методом (одинична діагональ у матриці L), з точністю* $\varepsilon \le 10^{-2}$*, знайти всі власні числа матриці. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
7. *LLT-методом, з точністю* $\varepsilon \le 10^{-2}$*, знайти всі власні числа матриці. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
8. *UTU -методом, з точністю* $\varepsilon \le 10^{-2}$*, знайти всі власні числа матриці. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
9. *З точністю* $\varepsilon \le 10^{-2}$ *алгоритмом №0 реалізації QR-метода знайти всі власні числа матриці. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
10. *З точністю* $\varepsilon \le 10^{-2}$ *алгоритмом №1 реалізації QR-метода знайти всі власні числа матриці. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
11. *З точністю* $\varepsilon \le 10^{-2}$ *алгоритмом №2 реалізації QR-метода знайти всі власні числа матриці. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
12. *З точністю* $\varepsilon \le 5 \cdot 10^{-2}$ *алгоритмом №3 реалізації QR-метода знайти всі власні числа матриці. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
13. *LU-методом (одинична діагональ у матриці U), з точністю* $\varepsilon \le 10^{-2}$*, знайти всі власні числа матриці. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
14. *З точністю* $\varepsilon \le 5 \cdot 10^{-2}$ *алгоритмом №4 реалізації QR-метода знайти всі власні числа матриці. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
15. *З точністю* $\varepsilon \le 5 \cdot 10^{-2}$ *алгоритмом №5 реалізації QR-метода знайти всі власні числа матриці. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*

16. *Привести задану матрицю до форми Хесенберга та методом Якобі №0 знайти всі власні числа та власні вектори, що їм відповідають з точністю $\varepsilon \le 10^{-2}$. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
17. *Привести задану матрицю до форми Хесенберга та методом Якобі №1 знайти всі власні числа та власні вектори, що їм відповідають з точністю $\varepsilon \le 10^{-2}$. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
18. *Привести задану матрицю до форми Хесенберга та методом Якобі №2 знайти всі власні числа та власні вектори, що їм відповідають з точністю $\varepsilon \le 3 \cdot 10^{-2}$. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
19. *Привести задану матрицю до форми Хесенберга та методом Якобі №3 знайти всі власні числа та власні вектори, що їм відповідають з точністю $\varepsilon \le 5 \cdot 10^{-2}$. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
20. *З точністю $\varepsilon \le 10^{-2}$ алгоритмом №4 реалізації методу Якобі знайти всі власні числа та власні вектори матриці. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
21. *З точністю $\varepsilon \le 7 \cdot 10^{-2}$ алгоритмом №5 реалізації методу Якобі знайти всі власні числа та власні вектори матриці. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
22. *З точністю $\varepsilon \le 10^{-2}$ будь-яким методом (вказати назву обраного метода) знайти всі власні числа матриці. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*

Модульна контрольна робота №2 включає такі типи задач:

1. *Виконати локалізацію всіх коренів системи нелінійних рівнянь. Знайти один з локалізованих коренів будь-яким ітераційним методом уточнення коренів системи нелінійних рівнянь (вказати назву обраного ітераційного методу) з точністю $\varepsilon \le 5 \cdot 10^{-2}$. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
2. *Виконати локалізацію всіх коренів системи нелінійних рівнянь. Знайти один з локалізованих коренів явним методом Ньютона розв'язання систем нелінійних рівнянь з точністю $\varepsilon \le 10^{-2}$. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
3. *Виконати локалізацію всіх коренів системи нелінійних рівнянь. Знайти один з локалізованих коренів неявним методом Ньютона розв'язання систем нелінійних рівнянь з точністю $\varepsilon \le 10^{-1}$. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
4. *Виконати локалізацію всіх коренів системи нелінійних рівнянь. Знайти один з локалізованих коренів спрощеним методом Ньютона розв'язання систем нелінійних рівнянь з точністю $\varepsilon \le 10^{-2}$. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
5. *Виконати локалізацію всіх коренів системи нелінійних рівнянь. Знайти один з локалізованих коренів двоступеневим методом Ньютона розв'язання систем*

нелінійних рівнянь з точністю $\varepsilon \le 7 \cdot 10^{-2}$. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.

6. *Виконати локалізацію всіх коренів системи нелінійних рівнянь. Знайти один з локалізованих коренів апроксимаційним аналогом №1 метода Ньютона розв'язання систем нелінійних рівнянь з точністю $\varepsilon \le 10^{-2}$. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
7. *Виконати локалізацію всіх коренів системи нелінійних рівнянь. Знайти один з локалізованих коренів апроксимаційним аналогом №2 метода Ньютона розв'язання систем нелінійних рівнянь з точністю $\varepsilon \le 10^{-2}$. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
8. *Виконати локалізацію всіх коренів системи нелінійних рівнянь. Знайти один з локалізованих коренів апроксимаційним аналогом №1 двоступеневого метода Ньютона розв'язання систем нелінійних рівнянь з точністю $\varepsilon \le 10^{-2}$. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
9. *Виконати локалізацію всіх коренів системи нелінійних рівнянь. Знайти один з локалізованих коренів апроксимаційним аналогом №2 двоступеневого метода Ньютона розв'язання систем нелінійних рівнянь з точністю $\varepsilon \le 10^{-2}$. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
10. *Виконати локалізацію всіх коренів системи нелінійних рівнянь. Знайти один з локалізованих коренів дискретним (крок дискретизації обрати сталим для всіх ітерацій) явним методом Ньютона розв'язання систем нелінійних рівнянь з точністю $\varepsilon \le 3 \cdot 10^{-2}$. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*
11. *Виконати локалізацію всіх коренів системи нелінійних рівнянь. Знайти один з локалізованих коренів дискретним (крок дискретизації обрати сталим для всіх ітерацій) неявним методом Ньютона розв'язання систем нелінійних рівнянь з точністю $\varepsilon \le 7 \cdot 10^{-2}$. Якщо необхідна кількість ітерацій перевищує значення 3, то дозволяється обмежитись трьома ітераціями.*