

**ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД УКООПСПІЛКИ
«ПОЛТАВСЬКИЙ УНІВЕРСИТЕТ ЕКОНОМІКИ І ТОРГІВЛІ»**

**ІНСТИТУТ ЕКОНОМІКИ, УПРАВЛІННЯ ТА ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ
ФОРМА НАВЧАННЯ ДЕННА
КАФЕДРА МАТЕМАТИЧНОГО МОДЕЛЮВАННЯ ТА СОЦІАЛЬНОЇ
ІНФОРМАТИКИ**

Допускається до захисту

Завідувач кафедри _____ О.О. Ємець
(підпис)

« _____ » _____ 2020 р.

**ПОЯСНЮВАЛЬНА ЗАПИСКА
ДО БАКАЛАВСЬКОЇ РОБОТИ**

на тему

**РОЗРОБКА ТРЕНАЖЕРУ З ТЕМИ «МЕТОД КВАЗІМІНОРИВ ДЛЯ
ВІДШУКАННЯ ШЛЯХІВ У ГРАФІ» ДИСЦИПЛІНИ «ТЕОРІЯ СИСТЕМ ТА
МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ»**

зі спеціальності 122 «Комп'ютерні науки та інформаційні технології»

Виконавець роботи Денисенко Валерій Вікторович

_____ « ____ » _____ 2020р.
(підпис)

Науковий керівник к.ф.-м.н., доц. Олексійчук Юрій Федорович

_____ « ____ » _____ 2020р.
(підпис)

ПОЛТАВА 2020р.

ЗМІСТ

ВСТУП.....	3
1. ПОСТАНОВКА ЗАДАЧІ	5
2. ІНФОРМАЦІЙНИЙ ОГЛЯД	7
2.1. Сутність та переваги дистанційного навчання	7
2.2. Огляд відомого програмного забезпечення для дистанційного навчання.....	12
2.3. Огляд матеріалу за темою роботи	15
3. ТЕОРЕТИЧНА ЧАСТИНА.....	18
3.1. Алгоритмізація задачі за темою роботи.....	18
3.2. Розробка блок-схеми, яка підлягає програмуванню.....	23
3.3. Обґрунтування вибору програмних засобів для реалізації завдання роботи	24
4. ПРАКТИЧНА ЧАСТИНА.....	28
4.1. Опис процесу програмної реалізації	28
4.2. Опис програми.....	32
4.3. Необхідна користувачу програми інструкція	35
ВИСНОВКИ	37
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	39

ВСТУП

Одним із наслідків інформатизації суспільства та постійного зниження вартості послуг на використання глобальної мережі Internet стало суттєве поглиблення процесів упровадження інформаційних технологій в освітню практику.

Для того, щоб студент зміг успішно виконувати свої функції у професійній діяльності, йому необхідно під час навчання набути певних практичних навичок та теоретичних знань. Однією із важливих умов підвищення розвитку теорії і практики є створення системи завдань, їх послідовність та різноманітність.

Важливим завданням при викладанні, зокрема, математичних дисциплін, є наочність демонстрації розв'язування тих чи інших завдань. Вирішити це завдання можливо через розробку та впровадження в дистанційний курс програм–тренажерів.

Навчаючись на дистанційних курсах студент підвищує свій інтелектуальний і творчий потенціал, навчається самостійно приймати рішення.

Метою роботи є програмна реалізація тренажера з теми «Метод квазімінорів для відшукування шляхів у графі» для дистанційного курсу «Теорія систем та математичне моделювання».

Об'єктом розробки в даній роботі є процес дистанційного навчання математичним дисциплінам.

Предметом розробки є програмний продукт, що реалізує тренажер для закріплення знань з методу квазімінорів для відшукування шляхів у графі на мові програмування Java.

Головне завдання – розробити алгоритм роботи тренажеру відповідно до теми роботи та створити його програмну реалізацію.

Перелік використаних методів полягає у застосуванні методу квазімінорів для відшукування шляхів у графі та програмних засобів мови Java.

При реалізації тренажеру використано середовище візуальної розробки програм NetBeans IDE та об'єктно-орієнтована мова програмування Java.

Тренажер готовий до використання в дистанційному курсі «Теорія систем та математичне моделювання».

Робота складається з чотирьох розділів. У першому розділі розглянуто постановку задачі тренажеру. У другому розділі описано сутність та переваги дистанційного навчання, огляд відомого програмного забезпечення для дистанційного навчання та матеріалу за темою роботи. У третьому розділі представлено алгоритмізацію задачі за темою роботи, блок-схему, яка підлягає програмуванню, обґрунтування вибору програмних засобів для реалізації завдання роботи. У четвертому розділі – описано процесу програмної реалізації, роботу програми і необхідну користувачу програми інструкцію.

Обсяг пояснювальної записки: 52 стор., в т.ч. основна частина - 39 стор., джерела - 12 назв.

1. ПОСТАНОВКА ЗАДАЧІ

Оскільки потрібно розробити навчальний тренажер з теми «Метод квазімінорів для відшукування шляхів у графі», як складову дистанційного курсу «Теорія систем та математичне моделювання», то слід виконати наступні завдання роботи:

- розглянути сутність та переваги дистанційного навчання;
- розглянути відоме програмне забезпечення для дистанційного навчання;
- розглянути матеріал за темою роботи;
- розглянути специфіку застосування методу квазімінорів для відшукування шляхів у графі;
- розробити алгоритм тренажеру, що дозволить закріпити знання з теми;
- обґрунтувати вибір програмного забезпечення;
- описати процес програмної реалізації;
- описати роботу тренажеру.

Щоб допомогти студенту у проходженні тренінгу потрібно розробити можливість звернутися до теоретичного матеріалу з теми. В алгоритмі слід покроково описати всі дії при розв'язанні прикладів, де необхідно застосувати метод квазімінорів для відшукування шляхів у графі.

Розроблений тренажер повинен мати зрозумілий для користувача інтерфейс.

Пропонується, щоб він містив:

- стартову сторінку;
- сторінку з теоретичним матеріалом або окремий файл;
- сторінку з умовою прикладу, завданням, варіантами відповідей;
- кінцеву сторінку.

Сторінка з теоретичним матеріалом має бути доступною користувачеві постійно, щоб він міг в будь-який момент до неї звернутися.

Якщо користувач невірно відповість на завдання, то повинна виводитися підказка.

При завершенні проходження тренажера необхідно реалізувати можливість розпочати проходження знову або завершити його.

2. ІНФОРМАЦІЙНИЙ ОГЛЯД

2.1. Сутність та переваги дистанційного навчання

Дистанційне навчання (ДН) — освітній процес, у якому значна частина викладання здійснюється викладачем (або групою викладачів), віддаленим у просторі або часі від учня (або групи учнів).

Сучасне дистанційне навчання здійснюється в основному за допомогою технологій і ресурсів мережі Інтернет.

Термін «дистанційне навчання» іноді вживається для позначення форм навчання, що існували задовго до появи комп'ютерів: заочне, кореспондентське, домашнє навчання, екстернат, для яких характерно навчання на відстані, дистанції.

Серед тлумачень змісту дистанційного навчання виділяють два підходи, які істотно розрізняються з дидактичної точки зору.

Перший досить розповсюджений сьогодні підхід такий: дистанційне навчання — обмін інформацією між учителем і учнем (групою учнів) за допомогою електронних мереж чи інших засобів телекомунікацій. Учень розглядається як одержувач деякого інформаційного змісту і системи завдань для його засвоєння. Результати самостійної роботи повертаються знову вчителю, який оцінює якість і рівень засвоєння матеріалу. Під знаннями розуміється трансльована інформація, а особистий досвід учнів і їх діяльність щодо конструювання знань майже не організуються.

Другий підхід принципово відрізняється від попереднього. Основою дистанційного навчання виступає особистісна продуктивна діяльність учнів, яка будується за допомогою сучасних засобів телекомунікацій. Цей підхід припускає інтеграцію інформаційних і педагогічних технологій, що забезпечують інтерактивність взаємодії суб'єктів освіти і продуктивність навчального процесу. Обмін і пересилання інформації відіграють у цьому випадку роль допоміжного середовища організації продуктивної освітньої

діяльності учнів. Навчання відбувається в реальному часі (чат, відео зв'язок, спільні для віддалених учнів і вчителя «віртуальні дошки» із графікою тощо), а також асинхронно (телеконференції на основі електронної пошти, форуми). Паралельно зі створенням учнями освітніх продуктів відбувається їх внутрішнє освітнє зростання. Особистісний, креативний і телекомунікативний характер освіти — основні риси дистанційного навчання цього типу. В процесі дистанційного навчання передбачається наявність викладача і учнів, їх спілкування, спілкування між собою, а також наявність у системі підручника, необхідного комплексу засобів навчання.

Розвиток системи дистанційного навчання обумовлений сукупністю переваг і можливостей. Це насамперед більш гнучкі умови навчання для дітей, які не змогли чи не можуть здобути її традиційним шляхом через віддаленість від кваліфікованих навчальних закладів, фізичних недоліків, індивідуальних особливостей і потреб.

Дистанційне навчання спроможне задовольнити додаткові освітні потреби учнів, особливо з навчальних дисциплін, змістовна частина яких дуже швидко змінюється, до яких слід віднести інформатику. Талановитий учень сільської школи може, наприклад, одночасно навчатися дистанційно у висококваліфікованих фахівців, які знаходяться в будь-якому куточку країни і світу, не залишаючи свого місця проживання. За допомогою електронних мереж учень з будь-якого міста чи села має доступ до світових культурних і наукових скарбів, вчиться в престижних університетах світу.

Дистанційне навчання має такі переваги перед очним

- оперативні (подолання бар'єрів у просторі та часі, одержання актуальної «свіжої» інформації, швидкий зворотний зв'язок),
- інформаційні (зростає доступність освітньої інформації, що знаходиться на спеціалізованих серверах, постачається учневі за

допомогою інтерактивних веб-каналів, публікується в телеконференціях, списках розсилання й інших засобах мережі Інтернет),

— комунікаційні (збільшується кількість потенційних учасників навчання — школярів, учителів, фахівців, які оперативно взаємодіють один з одним за допомогою електронних мереж, ліквідуються територіальні обмеження для проведення Інтернет-уроків, проектів, олімпіад),

— педагогічні (внаслідок специфіки дистанційних телекомунікацій навчання стає більш мотивованим, інтерактивним, технологічним індивідуалізованим, спрощується публікація учнівських робіт у мережі, їх експертиза та оцінка),

— психологічні (створення більш комфортних, порівняно з традиційними, емоційно-психологічних умов для самовираження учня, зняття психологічних бар'єрів і проблем, усунення помилок усного спілкування),

— економічні (загальні витрати на навчання зменшуються через економію транспортних витрат, витрат на оренду приміщень, скорочення «паперового» діловодства і тиражування посібників),

— ергономічні (учні і вчителі мають можливість розподіляти час занять за зручним для себе графіком і темпом, вибирати і використовувати для занять найбільш придатну техніку і комп'ютерне устаткування)

У таблиці 2.1 наведено порівняльні ознаки традиційної освіти і ДН.

Таблиця 2.1 – Порівняльні ознаки традиційної освіти і ДН

№ з/п	Ознаки порівняння	Традиційна освіта	Дистанційна освіта
1	"Центр" освітньої парадигми	Школа, вчитель, його професійні знання, жорстко визначені освітні технології	Учень та вибрані ним методи, технології і засоби навчання
2	Сутність процесу навчання	Репродукція знання учителя в учнях з використанням спрощених проблем	Постановка учнями реальних проблем і детальне вивчення кроків, методів і засобів розв'язування цих задач
3	Роль учителя	Практично єдиний доступний експерт в даній галузі та єдине джерело інформації	Радник і помічник учню
4	Роль учня	Пасивно вбирає знання від учителя	Активна (учень сам ставить реальні задачі, або одержує їх від компаній, і шукає підходи до їх розв'язання)
5	Навчальна програма і навчальний курс	Жорстка, статична	Гнучка, динамічна
6	Час навчання	Жорстко регламентується	Визначається самим учнем
7	Підручник з курсу	Друкований, пасивний, некольоровий, статичний	Постійно поновлюється, електронний, мультимедійний, активний, кольоровий з можливістю виклику зовнішніх програм; необмежені джерела глобального інформаційного простору
8	Технічні засоби навчання	Лабораторії і комп'ютерні мережі, що надаються школою	Домашній комп'ютер і/або підключення до інтернету. віртуальні наукові лабораторії
9	Додаткові інформаційні джерела з курсу	Локальні, лімітовані	Глобальні найкращі світові інформаційні джерела, необмежені

Дистанційна форма навчання передбачає створення і використання єдиного інформаційно-освітнього середовища, яке містить різні електронні джерела інформації, а саме:

курси дистанційного навчання, електронні підручники, розташовані на вітчизняних освітніх серверах (для різних моделей дистанційного навчання):

- віртуальні бібліотеки;
- бази даних освітніх ресурсів;
- веб-квести, призначені для цілей навчання;
- телекомунікаційні проекти;
- віртуальні методичні об'єднання вчителів;
- телеконференції, форуми для вчителів і учнів;
- консультаційні віртуальні центри (для вчителів, школярів, батьків);
- наукові об'єднання школярів.

Навчальне середовище формується таким чином, щоб учень мав вільний доступ з будь-якого курсу до:

- інформаційного забезпечення (довідники з відповідних предметів, енциклопедії, консультаційний центр);
- необхідних розділів курсів із суміжних галузей знань;
- лабораторних робіт, практикумів;
- веб-квестів;
- проектів [3-5].

2.2. Огляд відомого програмного забезпечення для дистанційного навчання

Дистанційне навчання – це добре організована й контрольована самоосвіта з використанням комп'ютерної техніки й комунікаційних мереж.

Одною з систем дистанційного навчання є Moodle. Moodle (Modular Object-Oriented Dynamic Learning Environment, вимовляється «Мудл») – це модульне об'єктно-орієнтоване динамічне навчальне середовище, яке називають також системою управління навчанням (LMS), системою управління курсами (CMS), віртуальним навчальним середовищем (VLE) або просто платформою для навчання, яка надає викладачам, учням та адміністраторам дуже розвинутий набір інструментів для комп'ютеризованого навчання, в тому числі дистанційного. Moodle можна використовувати в навчанні школярів, студентів, при підвищенні кваліфікації, бізнес-навчанні, як в комп'ютерних класах навчального закладу, так і для самостійної роботи вдома.

Ця система призначена для організації навчання онлайн у мережевому середовищі з використанням технологій Інтернету. Система забезпечує різноманіття процедур навчання онлайн, комбінуванням яких може бути організоване ефективне навчання в навчальному закладі. Moodle надає можливість інсталяції освітніх ресурсів (навчальних матеріалів) і забезпечує засобами доступу до ресурсів та управління ними; забезпечує комунікаційну взаємодію учасників освітнього процесу, що реалізовується у формі інтернетконференцій, форумів, дискусій, а також обміну посланнями, що містять, зокрема, завдання для тих, хто навчається, виконання завдань і коментарі.

Використання мережевих технологій дозволяє студенту будувати свою стратегію вивчення навчальної дисципліни та сприяє не тільки засвоєнню студентами знань, умінь, навичок, форм професійної поведінки,

а й формуванню певної структури особистісних якостей. Дослідження умов формування професійної компетентності майбутніх фахівців засобами мережевих технологій дозволило установити, що впровадження навчально-методичного комплексу, який містить дистанційні курси, розроблені засобами Moodle з використанням модульного підходу, представлення знань як динамічної, мультимодальної структури, у формуванні якої беруть участь студенти, сприяє набуттю студентами досвіду самостійного поповнення та оновлення професійних знань, особистісної причетності до цього процесу та відповідальності за нього.

Модульне навчання базується на позиціях діяльнісного, гнучкого підходу до визначення послідовності засвоєння навчального матеріалу, а сам модуль визначається як цілісна конструкція, що поєднує навчальний зміст і технологію оволодіння ним. Важливим критерієм побудови модулю є структурування діяльності студента в логіку етапів засвоєння знань: сприйняття, розуміння, осмислення, запам'ятовування, застосування, узагальнення, систематизація.

У цілому, дотримуючись загально дидактичних принципів до визначення етапів процесу навчання та спираючись на наявні дослідження структури процесу професійного навчання, для вивчення кожного окремого навчального модулю визначено таку послідовність етапів модульного навчання:

- 1) оглядове навчання;
- 2) вивчення теоретичного матеріалу;
- 3) засвоєння, формування та закріплення професійних умінь і навичок;
- 4) розвиваючий тренінг;
- 5) модульний контроль.

На кожному з описаних етапів професійного навчання запропоновано найбільш оптимальні засоби, форми та методи навчання,

які значною мірою обумовлені специфікою інформаційного середовища Moodle.

Система дистанційного навчання Moodle має досить багато можливостей як для студентів так і для викладачів.

У середовищі Moodle студенти отримують:

1) доступ до навчальних матеріалів (тексти лекцій, завдання до практичних/лабораторних та самостійних робіт; додаткові матеріали (книги, довідники, посібники, методичні розробки) та засобів для спілкування і тестування;

2) засоби для групової роботи (Вікі, форум, чат, семінар, вебінар);

3) можливість перегляду результатів проходження дистанційного курсу студентом;

4) можливість перегляд результатів проходження тесту;

5) можливість спілкування з викладачем через особисті повідомлення, форум, чат;

6) можливість завантаження файлів з виконаними завданнями;

7) можливість використання нагадувань про події у курсі.

Викладачам надається можливість:

1) використання інструментів для розробки авторських дистанційних курсів;

2) розміщення навчальних матеріалів (тексти лекцій, завдання до практичних/лабораторних та самостійних робіт; додаткові матеріали (книги, довідники, посібники, методичні розробки) у форматах .doc, .odt, .html, .pdf, а також відео, аудіо і презентаційні матеріали у різних форматах та через додаткові плагіни;

3) додавання різноманітних елементів курсу;

4) проведення швидкої модифікації навчальних матеріалів;

5) використання різних типів тестів;

6) автоматичного формування тестів;

7) автоматизації процесу перевірки знань, звітів щодо проходження студентами курсу та звітів щодо проходження студентами тестів;

8) додавання різноманітних плагінів до курсу дозволяє викладачу використовувати різноманітні сторонні програмні засоби для дистанційного навчання.

Отже, Moodle – це безкоштовна, відкрита (Open Source) система управління навчанням. Вона орієнтована насамперед на організацію взаємодії між викладачем та учнями, хоча підходить і для організації традиційних дистанційних курсів, а також підтримки очного навчання. [6-9].

2.3. Огляд матеріалу за темою роботи

Розглянемо кілька методів відшукування шляхів у графі, які є зручними в застосуванні.

Перший метод - використання квазімінорів.

Означення. Квазімінором M_{lk} елемента a_{lk} матриці A називають матрицю, яку одержують, викресливши в A рядок l та стовбець k .

Позначимо

$$M_{lk} = \left| a_{ij \rightarrow lk} \right|_{kl}$$

1. Будуємо матрицю безпосередніх шляхів $A = \left\| a_{ij} \right\|$, тобто таку матрицю A , де $a_{ij} \neq 0$ показує, що є шлях з i в j , $a_{ij} = 0$ — в іншому разі. За нею обчислюємо шляхи з вершини k в l :

$$\text{у вигляді } \pi_{kl} = a_{ki_1} a_{i_1 i_2} \dots a_{i_{m-1} i_m} a_{i_m l} + \dots + a_{kj_1} a_{j_1 j_2} \dots a_{j_{n-1} j_n} a_{j_n l}$$

2. Викреслюємо в вихідній матриці стовбець з номером k , що відповідає номеру вершини, з якої починається шлях, та рядок з номером l , що відповідає номеру вершини, в якій цей шлях закінчується:

$$\pi_{kl} = \left| a_{ij \rightarrow lk} \right|_{kl} = M_{lk}$$

В [2] доведено, що при $k \neq l$ шляхи з k в l символічно представляються як

$$\pi_{kl} = \left| a_{ij \rightarrow lk} \right|_{kl} = \sum_{\forall j \neq k} a_{kj} A_{kj}, \quad (2.1)$$

де A_{kj} - квазімінор, що містить на один рядок та на один стовпець менше, ніж M_{lk} , та $A_{kj} = 1$, при $j=l$ (тобто коли він відповідає елементу, що стоїть в стовпці з номером первісно викресленого рядка).

Зауваження. Якщо в (2.1) маємо доданок $a_{ij} A_{ij}$, то квазімінор A_{ij} далі розкладається по рядку, що мав в вихідній матриці номер j .

Зауважимо також, що кількість шляхів a_{kl} з вершини k в l одержимо, поклавши в π_{kl} всі $a_{ij} = 1$.

Приклад 1. Нехай задано граф. Знайти кількість шляхів між вершинами. Знайдемо шляхи з 1 в 2.

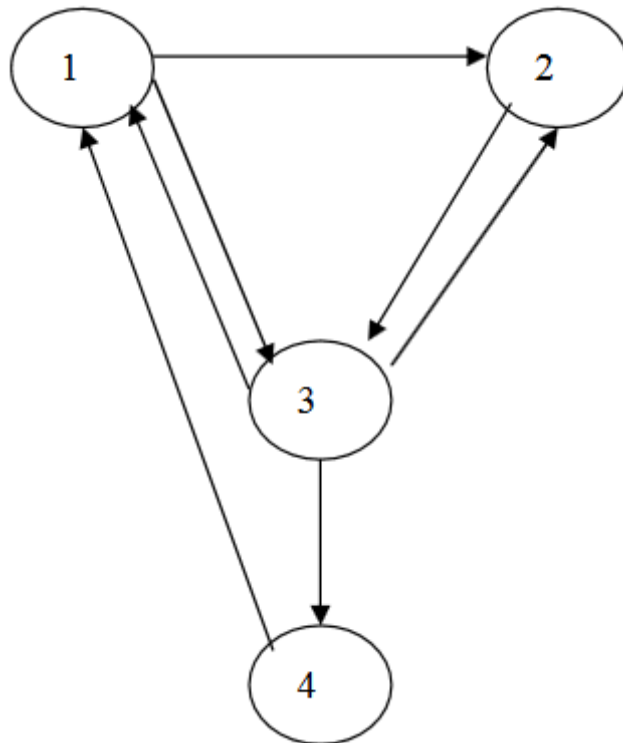


Рисунок 2.1 – Граф задачі (приклад 1) про кількість шляхів

$$A = \begin{array}{c|ccc|c} & 1 & 2 & 3 & 4 \\ \hline & & a_{12} & a_{13} & \\ \hline & & & a_{23} & \\ \hline a_{31} & & a_{32} & & a_{34} \\ \hline a_{41} & & & & \\ \hline \end{array} \begin{array}{l} 1 \\ 2 \\ 3 \\ 4 \end{array}$$

$$\pi_{12} = M_{21^*} = \begin{array}{c|cc|c} & 2^* & 3 & 4 \\ \hline & a_{12} & a_{13} & \\ \hline & a_{32} & & a_{34} \\ \hline & & & \\ \hline \end{array} \begin{array}{l} 1^* \\ 3 \\ 4 \end{array} =$$

$$= a_{12} \cdot 1 + a_{13} \cdot \begin{array}{c|c|c} & 2^* & 3 \\ \hline & a_{32} & a_{34} \\ \hline & & \\ \hline \end{array} \begin{array}{l} 3^* \\ 4 \end{array} = a_{12} + a_{13} \cdot (a_{32} \cdot 1) = a_{12} + a_{13}a_{32}$$

Отже маємо два шляхи: $1 \rightarrow 2$; $1 \rightarrow 3 \rightarrow 2$. Їх отримуємо за послідовністю індексів в доданках π_{12} .

Знайдемо шляхи з вершини 3 в 2.

$$\pi_{32} = M_{23^*} = \begin{array}{c|cc|c} & 1 & 2^* & 4 \\ \hline & 0 & a_{12} & 0 \\ \hline & a_{31} & a_{32} & a_{34} \\ \hline & a_{41} & 0 & 0 \\ \hline \end{array} \begin{array}{l} 1 \\ 3^* \\ 4 \end{array} = a_{31} \cdot a_{12} + a_{32} \cdot 1 +$$

$$+ a_{34} \cdot \begin{array}{c|c|c} & 1 & 2 \\ \hline & & a_{12} \\ \hline & a_{41} & \\ \hline \end{array} \begin{array}{l} 1 \\ 4^* \end{array} = a_{31}a_{12} + a_{32} + a_{34}a_{41}a_{12}$$

Отже знайдено 3 шляхи (по кількості доданків), а саме $3 \rightarrow 1 \rightarrow 2$; $3 \rightarrow 2$; $3 \rightarrow 4 \rightarrow 1 \rightarrow 2$ [10].

3. ТЕОРЕТИЧНА ЧАСТИНА

3.1. Алгоритмізація задачі за темою роботи

На головній сторінці тренажеру відображається тема, пропонується відкрити теоретичний матеріал або перейти до тестування.

Крок 1. Користувачу виводиться завдання: Квазімінором M_{lk} елемента a_{lk} матриці A називають матрицю, яку одержують:

- викресливши в M_{lk} рядок l та стовбець k ;
- викресливши в A рядок l та стовбець k ;
- викресливши в A рядок l ;
- викресливши в A стовбець k ;

Якщо вибрано другий варіант, то перехід на крок 2.

Якщо вибрано інший варіант, то виводиться підказка: Квазімінором M_{lk} елемента a_{lk} матриці A називають матрицю, яку одержують, викресливши в A рядок l та стовбець k .

Крок 2. Користувачу виводиться завдання: Будуємо матрицю безпосередніх шляхів $A = \|a_{ij}\|$, тобто таку матрицю A , де $a_{ij} \neq 0$ показує, що є шлях з i в j , $a_{ij} = 0$ — в іншому разі. За нею обчислюємо шляхи з вершини k в l у вигляді:

- $\pi_{kl} = a_{ki_1} a_{i_1 i_2} \dots + a_{i_{m-1} i_m} a_{i_m l} + \dots + a_{kj_1} a_{j_1 j_2} \dots a_{j_{n-1} j_n} a_{j_n l}$;
- $\pi_{kl} = a_{ki_1} a_{i_1 i_2} \dots a_{i_{m-1} i_m} a_{i_m l} + \dots + a_{kj_1} a_{j_1 j_2} + \dots + a_{j_{n-1} j_n} a_{j_n l}$;
- $\pi_{kl} = a_{ki_1} a_{i_1 i_2} \dots a_{i_{m-1} i_m} a_{i_m l} \dots a_{kj_1} a_{j_1 j_2} + \dots + a_{j_{n-1} j_n} a_{j_n l}$;
- $\pi_{kl} = a_{ki_1} a_{i_1 i_2} \dots a_{i_{m-1} i_m} a_{i_m l} + \dots + a_{kj_1} a_{j_1 j_2} \dots a_{j_{n-1} j_n} a_{j_n l}$;

Якщо вибрано четвертий варіант, то перехід на крок 3.

Якщо вибрано інший варіант, то виводиться підказка: Обчислюємо шляхи з вершини k в l у вигляді

$$\pi_{kl} = a_{ki_1} a_{i_1 i_2} \dots a_{i_{m-1} i_m} a_{i_m l} + \dots + a_{kj_1} a_{j_1 j_2} \dots a_{j_{n-1} j_n} a_{j_n l}.$$

Крок 3. Користувачу виводиться завдання: Викреслюємо в вихідній матриці стовбець з номером k , що відповідає номеру вершини, з якої починається шлях, та рядок з номером l , що відповідає номеру вершини, в якій цей шлях закінчується:

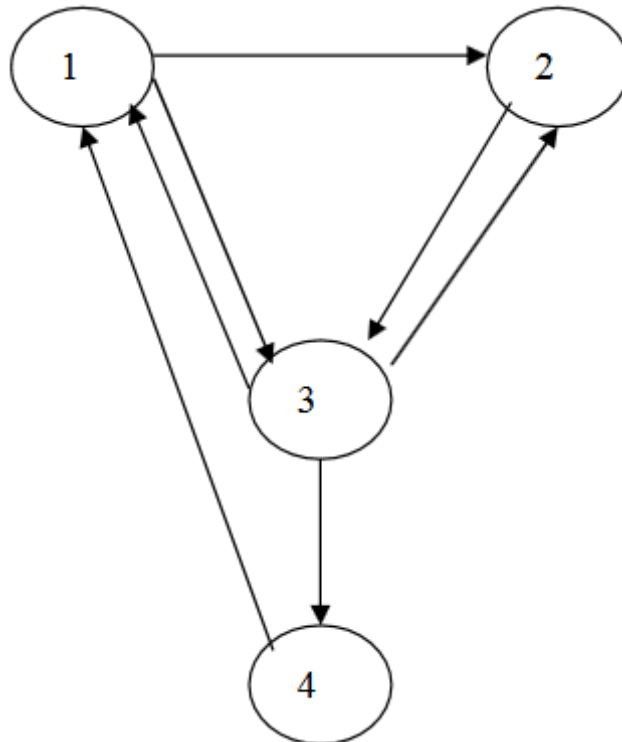
- $\pi_{kl} = \left| a_{ij \rightarrow lk} \right|_{kl} = M_{lk}$;
- $\pi_{kl} = \left| a_{lk \rightarrow ij} \right|_{kl} = M_{lk}$;
- $\pi_{kl} = \left| a_{ij \rightarrow lk} \right|_{ij} = M_{lk}$;
- $\pi_{kl} = \left| a_{ij \rightarrow lk} \right|_{kl} = M_{ij}$;

Якщо вибрано перший варіант, то перехід на крок 4.

Якщо вибрано інший варіант, то виводиться підказка: Викреслюємо в вихідній матриці стовбець з номером k та рядок з номером l :

$$\pi_{kl} = \left| a_{ij \rightarrow lk} \right|_{kl} = M_{lk}.$$

Крок 4. Користувачу виводиться завдання: Нехай задано граф. Знайти кількість шляхів між вершинами. Знайдемо шляхи з 1 в 2.



	1	2	3	4	
A=		a_{12}	a_{13}		1
			a_{23}		2
	a_{31}	a_{32}		a_{34}	3
	a_{41}				4

Встановити послідовність розв'язку:

$$(1) \quad \pi_{12} = M_{21^*} = \begin{array}{|c|c|c|} \hline & 2^* & 3 & 4 \\ \hline & a_{12} & a_{13} & \\ \hline & a_{32} & & a_{34} \\ \hline & & & \\ \hline \end{array} \begin{array}{l} 1^* \\ 3 \\ 4 \end{array}$$

$$(2) \quad a_{12} \cdot 1 + a_{13} \cdot \begin{array}{|c|c|} \hline & 2^* & 3 \\ \hline & a_{32} & a_{34} \\ \hline & & \\ \hline \end{array} \begin{array}{l} 3^* \\ 4 \end{array}$$

$$(3) \quad a_{12} + a_{13} \cdot (a_{32} \cdot 1) = a_{12} + a_{13}a_{32}$$

Якщо вказано вірно, то перехід на крок 5.

Якщо вказано невірно, то виводиться підказка:

$$\begin{aligned} \pi_{12} = M_{21^*} &= \begin{array}{|c|c|c|} \hline & 2^* & 3 & 4 \\ \hline & a_{12} & a_{13} & \\ \hline & a_{32} & & a_{34} \\ \hline & & & \\ \hline \end{array} \begin{array}{l} 1^* \\ 3 \\ 4 \end{array} = \\ &= a_{12} \cdot 1 + a_{13} \cdot \begin{array}{|c|c|} \hline & 2^* & 3 \\ \hline & a_{32} & a_{34} \\ \hline & & \\ \hline \end{array} \begin{array}{l} 3^* \\ 4 \end{array} = a_{12} + a_{13} \cdot (a_{32} \cdot 1) = a_{12} + a_{13}a_{32} \end{aligned}$$

Крок 5. Користувачу виводиться завдання: Отже знайдено шляхи.

Вказати їх кількість:

- 1;
- 2;
- 3;
- 4;

Якщо вибрано другий варіант, то перехід на крок 6.

Якщо вибрано інший варіант, то виводиться підказка: Отже маємо два шляхи.

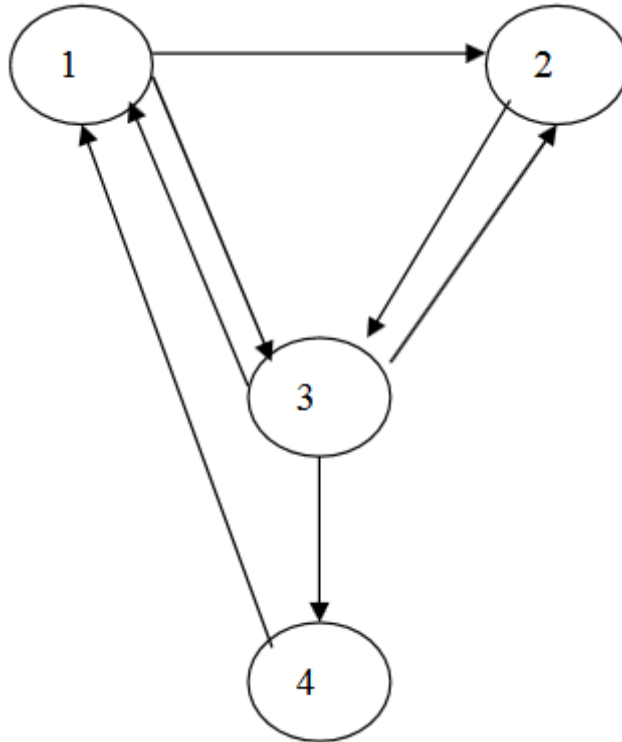
Крок 6. Користувачу виводиться завдання: Отже маємо два шляхи. Їх отримуємо за послідовністю індексів в доданках π_{12} .

Вказати шляхи: $_ \rightarrow _$; $_ \rightarrow _ \rightarrow _$

Якщо вказано $1 \rightarrow 2$; $1 \rightarrow 3 \rightarrow 2$, то перехід на крок 7.

Якщо вказано невірно, то виводиться підказка: Отже маємо два шляхи: $1 \rightarrow 2$; $1 \rightarrow 3 \rightarrow 2$.

Крок 7. Користувачу виводиться завдання: Нехай задано граф. Знайти кількість шляхів між вершинами. Знайдемо шляхи з вершини 3 в 2.



	1	2	3	4	
$A =$		a_{12}	a_{13}		1
			a_{23}		2
	a_{31}	a_{32}		a_{34}	3
	a_{41}				4

Встановити послідовність розв'язку:

$$(1) \quad \pi_{32} = M_{23^*} = \begin{array}{|c|c|c|} \hline 1 & 2^* & 4 \\ \hline 0 & a_{12} & 0 \\ \hline a_{31} & a_{32} & a_{34} \\ \hline a_{41} & 0 & 0 \\ \hline \end{array} \begin{array}{l} 1 \\ 3^* \\ 4 \end{array}$$

$$(2) \quad a_{31} \cdot a_{12} + a_{32} \cdot 1 + a_{34} \cdot \begin{array}{|c|c|} \hline 1 & 2 \\ \hline a_{41} & \\ \hline \end{array} \begin{array}{l} 1 \\ 4^* \end{array}$$

$$(3) \quad a_{31}a_{12} + a_{32} + a_{34}a_{41} a_{12}$$

Якщо вказано вірно, то перехід на крок 8.

Якщо вказано невірно, то виводиться підказка:

$$\begin{aligned} \pi_{32} = M_{23^*} = & \begin{array}{|c|c|c|} \hline 1 & 2^* & 4 \\ \hline 0 & a_{12} & 0 \\ \hline a_{31} & a_{32} & a_{34} \\ \hline a_{41} & 0 & 0 \\ \hline \end{array} \begin{array}{l} 1 \\ 3^* \\ 4 \end{array} = a_{31} \cdot a_{12} + a_{32} \cdot 1 + \\ & + a_{34} \cdot \begin{array}{|c|c|} \hline 1 & 2 \\ \hline a_{41} & \\ \hline \end{array} \begin{array}{l} 1 \\ 4^* \end{array} = a_{31}a_{12} + a_{32} + a_{34}a_{41} a_{12} \end{aligned}$$

Крок 8. Користувачу виводиться завдання: Отже знайдено шляхи.

Вказати їх кількість:

- 1;
- 2;
- 3;
- 4;

Якщо вибрано третій варіант, то перехід на крок 9.

Якщо вибрано інший варіант, то виводиться підказка: Отже знайдено 3 шляхи.

Крок 9. Користувачу виводиться завдання: Отже знайдено 3 шляхи (по кількості доданків).

Вказати шляхи: $_ \rightarrow _ \rightarrow _$; $_ \rightarrow _$; $_ \rightarrow _ \rightarrow _ \rightarrow _$

Якщо вказано $3 \rightarrow 1 \rightarrow 2$; $3 \rightarrow 2$; $3 \rightarrow 4 \rightarrow 1 \rightarrow 2$, то перехід на крок 10.

Якщо вказано невірно, то виводиться підказка: Отже знайдено 3 шляхи (по кількості доданків), а саме $3 \rightarrow 1 \rightarrow 2$; $3 \rightarrow 2$; $3 \rightarrow 4 \rightarrow 1 \rightarrow 2$.

Крок 10. Користувачу виводиться повідомлення про завершення тренажеру, пропонується розпочати знову або завершити роботу.

3.2. Розробка блок-схеми, яка підлягає програмуванню

На рисунку 3.1 зображено блок-схему алгоритму роботи тренажеру.

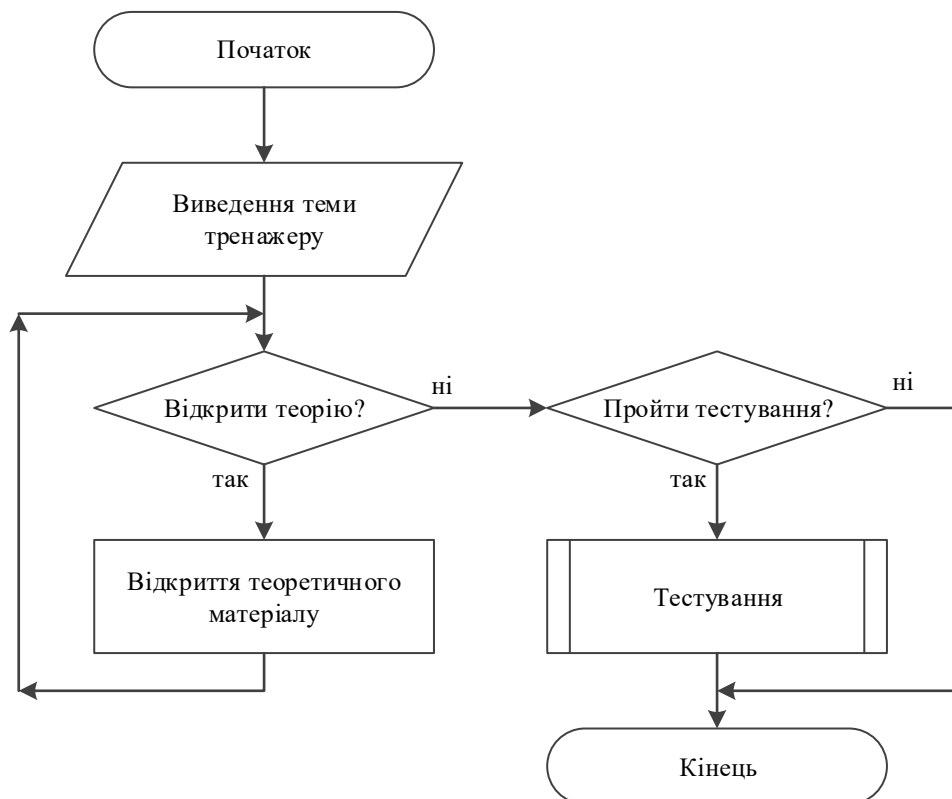


Рисунок 3.1 – Блок-схема алгоритму роботи тренажеру

На рисунку 3.2 зображено блок-схему алгоритму тестування.

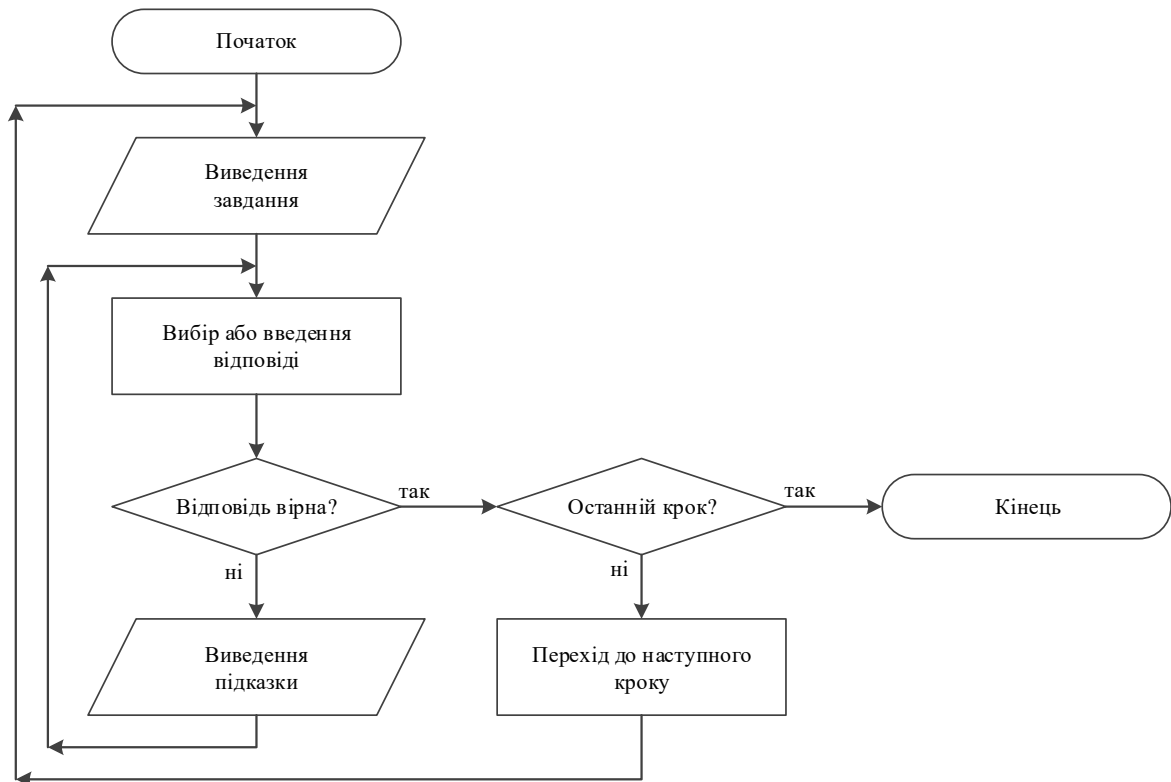


Рисунок 3.2 – Блок-схема алгоритму тестування

3.3. Обґрунтування вибору програмних засобів для реалізації завдання роботи

На сьогоднішній день Java займає одне з провідних місць у сфері розробки програмного забезпечення. Він займає лідируюче місце в банківському і телекомунікаційному секторах. Java об'єктно-орієнтована, сильно типізована мова програмування, розроблена компанією Sun Microsystems (в подальшому придбаній компанією Oracle). Програми Java зазвичай транслюються в спеціальний байт-код, тому вони можуть працювати на будь-якій комп'ютерній архітектурі, за допомогою віртуальної Java-машини.

Актуальність дослідження програмування на Java обумовлена зростанням популярності мови програмування, що представляє собою об'єктно-орієнтовану мову. Синтаксис Java схожий з синтаксисом мови C++. Так, наприклад, в Java немає покажчиків, які складні у використанні і потенційно можуть послужити причиною доступу програмного продукту до області пам'яті, що не належить їй. Разом з тим, в Java немає множинного спадкоємства і шаблонів.

Існуючі особливості зробили Java-середовище ідеальним засобом розробки клієнтських компонентів Web-систем.

Програми на Java, як і програми, написані на одному з мов високого рівня, не може бути відразу ж виконана і вимагає компіляції, тобто перекладу в послідовність машинних команд.

Інша особливість Java полягає в тому, що всі стандартні функції, що викликаються в програмі, підключаються до неї тільки на етапі виконання, а не включаються в байт-коди. Така динамічна компоновка значно зменшує скомпільований обсяг.

В даний час прийнято говорити про Java не як про окремих мовою програмування, а як про цілу родину технологій.

В даний час Java-технології знаходять широке застосування в різних сферах.

Мова Java використовується для розробки звичайних прикладних програм, які запускаються користувачем і мають доступ до всіх ресурсів комп'ютера нарівні з будь-якими іншими програмами.

Мова Java використовується для розробки спеціалізованих програм, призначених для використання на мобільних пристроях. Для їх запуску мобільний пристрій має підтримувати відповідну можливість.

Мова Java використовується для розробки спеціалізованих програм, зазвичай невеликого розміру, що запускаються браузером всередині web-документа для виведення динамічного вмісту та/або інтерактивної взаємодії з користувачем.

На мові Java розробляються спеціалізовані програми-сервлети, що функціонують в рамках Web-сервера і мають доступ до його ресурсів: файлам, баз даних і т.п. Для запуску сервлетів використовується Web-сервер зі спеціальним модулем (контейнером сервлетів) або спеціальний сервер додатків /

В процесі написання даної роботи були з'ясовані такі властивості мови Java:

- мова програмування об'єктно-орієнтована, оснащена багатою бібліотекою класів і в той же час досить проста для освоєння;
- цикл розробки додатків скорочений за рахунок того, що система побудована на основі інтерпретатора;
- додаток виходить автоматично стерпним між безліччю платформ і операційних систем;
- за рахунок вбудованої системи збірки сміття програміст звільняється від необхідності явного управління пам'яттю;
- додаток легко супроводжується і модифікується, тому що модулі можуть бути завантажені з мережі;
- в додатки вбудована система безпеки, що не допускає незаконного доступу і проникнення вірусів.

Аплети це програми Java, що включаються в сторінки HTML. Сторінка HTML повинна повідомляти браузеру, який аплет завантажити, і куди помістити кожен аплет на сторінці. Як ви і могли очікувати, для використання аплету необхідно спеціальний дескриптор, який повідомляє браузеру, де взяти файли класів, і як аплет позиціонується на Web-сторінці (розмір, місце розташування і т.п.). Потім браузер витягує необхідні файли класів з Internet (або з каталогу на користувальницької машині) і автоматично запускає аплет.

Коли з'явилися аплети, програмісти повинні були використовувати для перегляду Web-сторінок браузер HotJava, спеціально створений

компанією Sun для підтримки аплетів. Популярність аплетів істотно зросла, коли компанія Netscape включила віртуальну машину мови Java в свій браузер. Згодом така можливість з'явилася в Internet Explorer. На жаль, виникла проблема. Компанія Netscape не стала підтримувати більш сучасні версії мови Java, а компанія Microsoft схилилася до вирішення зовсім відмовлятися від Java.

Для того щоб вирішити дану проблему, компанія Sun розробила і розповсюдила інструмент під назвою Java Plug-In. Використовуючи різні механізми розширення можливостей Internet Explorer і Netscape, цей інструмент можна абсолютно вільно вбудувати в обидва браузера. Це дозволяє використовувати їх для запуску аплетів за допомогою середовища підтримки Java (Java Runtime Environment), розробленої компанією Sun. Передбачалося, що Java Plug-In дозволить скористатися перевагами останніх версій віртуальних машин. [11].

4. ПРАКТИЧНА ЧАСТИНА

4.1. Опис процесу програмної реалізації

Для роботи тренажеру було підключено бібліотеки:

```
import java.awt.CardLayout;
import java.awt.Desktop;
import java.io.BufferedInputStream;
import java.io.File;
import java.io.FileOutputStream;
import java.io.IOException;
import java.util.logging.Level;
import java.util.logging.Logger;
```

Запуск тренажеру відбувається завдяки void init().

```
@Override
public void init() {
    try {
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                initComponents();
            }
        });
    } catch (Exception ex) {
        ex.printStackTrace();
    }
}
```

Також було оголошено глобальну змінну int step для переходу по кроках відповідно алгоритму, присвоєно їй значення 1.

```
int step = 1;
```

Для відкриття файлу з матеріалом використовується подія `void jButton1ActionPerformed(java.awt.event.ActionEvent evt)`.

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
{
    BufferedInputStream in = null;
    FileOutputStream fout = null;
    try {
        in = new
BufferedInputStream(getClass().getResource("theory.pdf").openStream());
        fout = new FileOutputStream("theory.pdf");
        byte data[] = new byte[1024];
        int count;
        while ((count = in.read(data, 0, 1024)) != -1) {
            fout.write(data, 0, count);
        }
    } catch (IOException ex) {
        Logger.getLogger(JApplet.class.getName()).log(Level.SEVERE,
null, ex);
    } finally {
        try {
            if (in != null)
                in.close();
            if (fout != null) {
                fout.close();
                if (Desktop.isDesktopSupported()) {
```

```

File file = new
File(System.getProperty("user.dir")+"\\theory.pdf");
    Desktop.getDesktop().open(file);
    }
    }
    }
    catch (IOException ex) {
        Logger.getLogger(JApplet.class.getName()).log(Level.SEVERE,
null, ex);
    }
    }
    }
}

```

Щоб розпочати тестування було створено void
jButton2ActionPerformed(java.awt.event.ActionEvent evt).

```

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt)
{
    CardLayout cl=(CardLayout) Panel.getLayout();
    cl.show(Panel, "test");
    jLabel6.setVisible(false);
    jLabel8.setVisible(false);
    jLabel18.setVisible(false);
    jLabel28.setVisible(false);
    jLabel33.setVisible(false);
    jLabel35.setVisible(false);
    jLabel44.setVisible(false);
    jLabel49.setVisible(false);
    jLabel51.setVisible(false);
}

```

Кнопка «Далі» перевіряє відповідь і вибирає наступну дію за допомогою події `void jButton5ActionPerformed(java.awt.event.ActionEvent evt)` (див. Додаток А).

На кінцевій сторінці можна розпочати проходження знову завдяки `void jButton3ActionPerformed(java.awt.event.ActionEvent evt)` або закрити тренажер завдяки події `void jButton4ActionPerformed(java.awt.event.ActionEvent evt)`.

```
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt)
{
    buttonGroup1.clearSelection();
    step=1;
    jTextField1.setText("");
    ...

    jTextField20.setText("");
    jLabel6.setVisible(false);

    ...

    jLabel51.setVisible(false);
    CardLayout cl=(CardLayout) Panel.getLayout();
    cl.show(Panel, "test");
    cl =(CardLayout)jPanel1.getLayout();
    cl.show(jPanel1, "step1");
}
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt)
{
    System.exit(0);
}
```

4.2. Опис програми

Стартова сторінка містить тему тренажеру, пропонується відкрити теорію або розпочати тестування (рис.4.1).

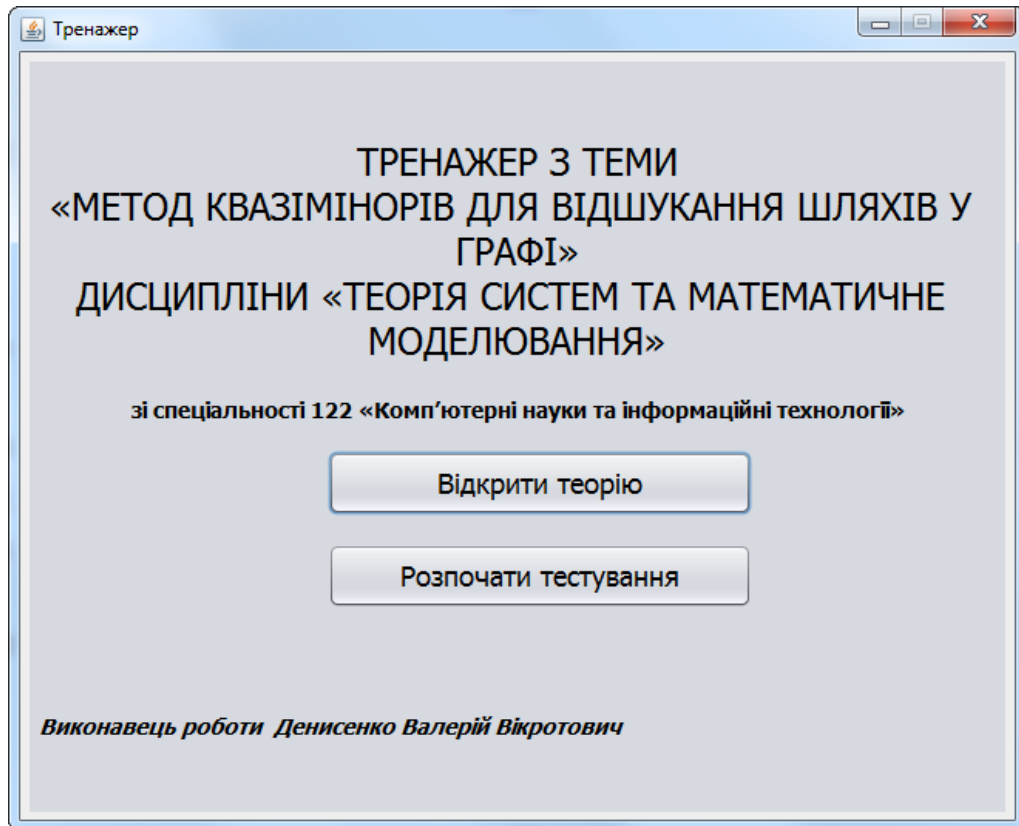


Рисунок 4.1 – Стартова сторінка

При тестуванні виводиться завдання і надається можливість вибрати відповідь із наданих або заповнити комірки (рис. 4.2, 4.3).

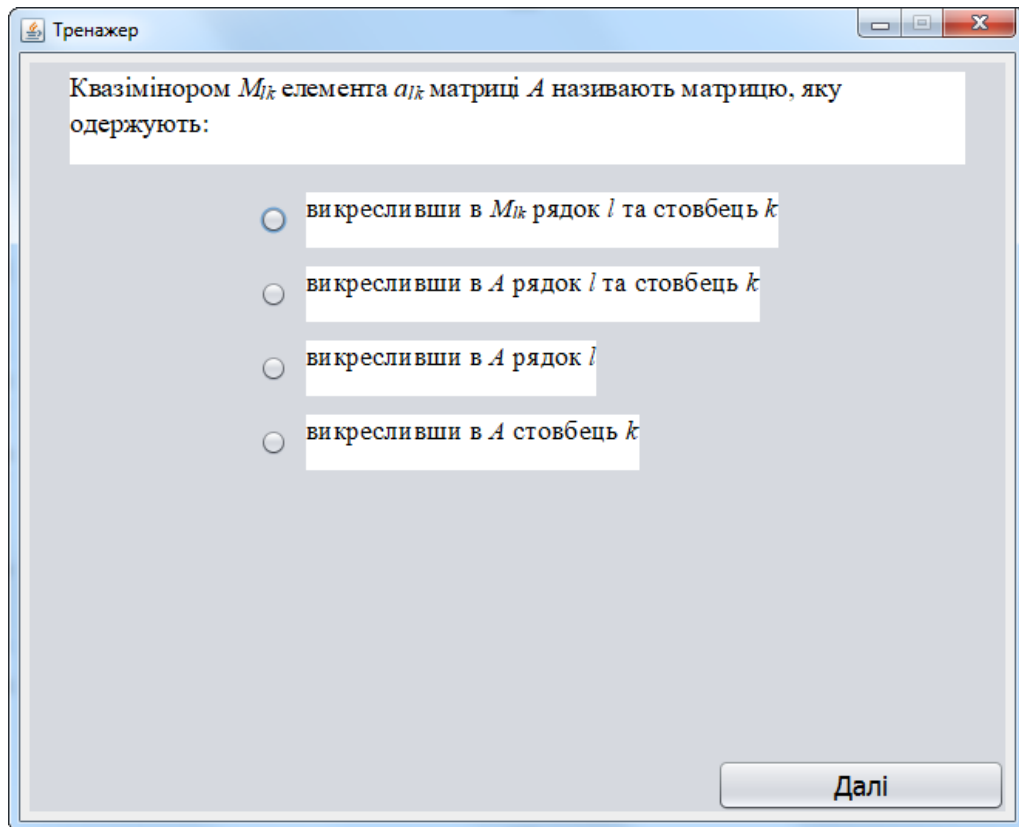


Рисунок 4.2 – Вибір відповіді

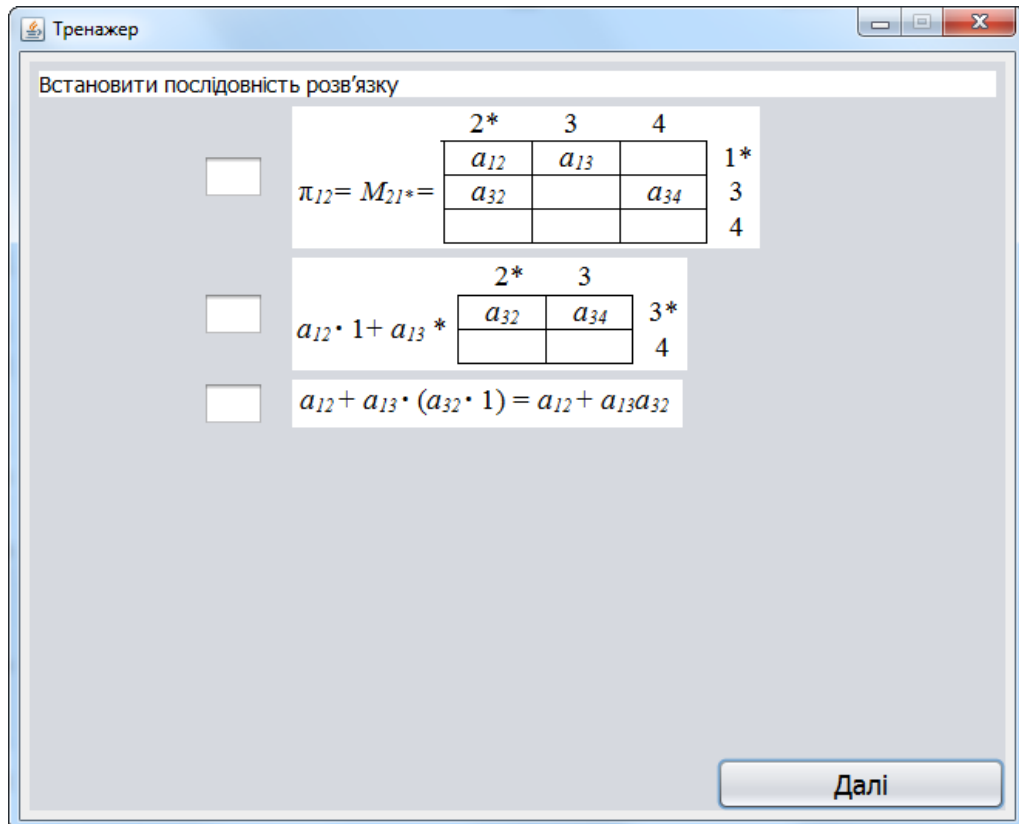


Рисунок 4.3 – Заповнення комірок

Якщо відповідь невірна, то відображається підказка (рис. 4.4).

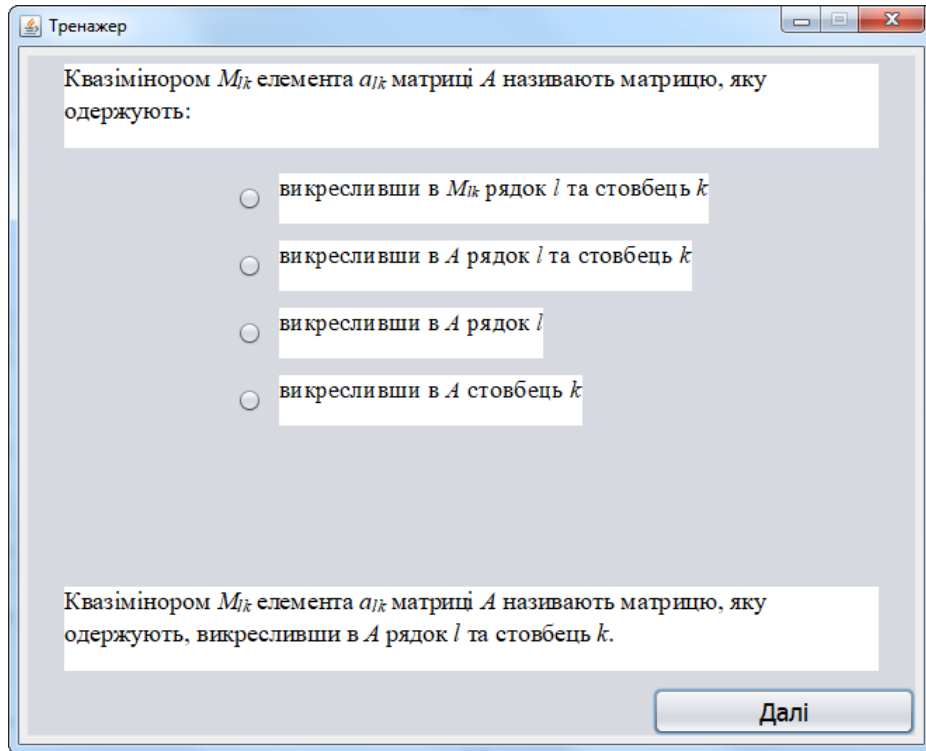


Рисунок 4.4 – Підказка

Кінцева сторінка містить повідомлення про завершення, пропонується розпочати тестування знову або завершити роботу (рис. 4.5).

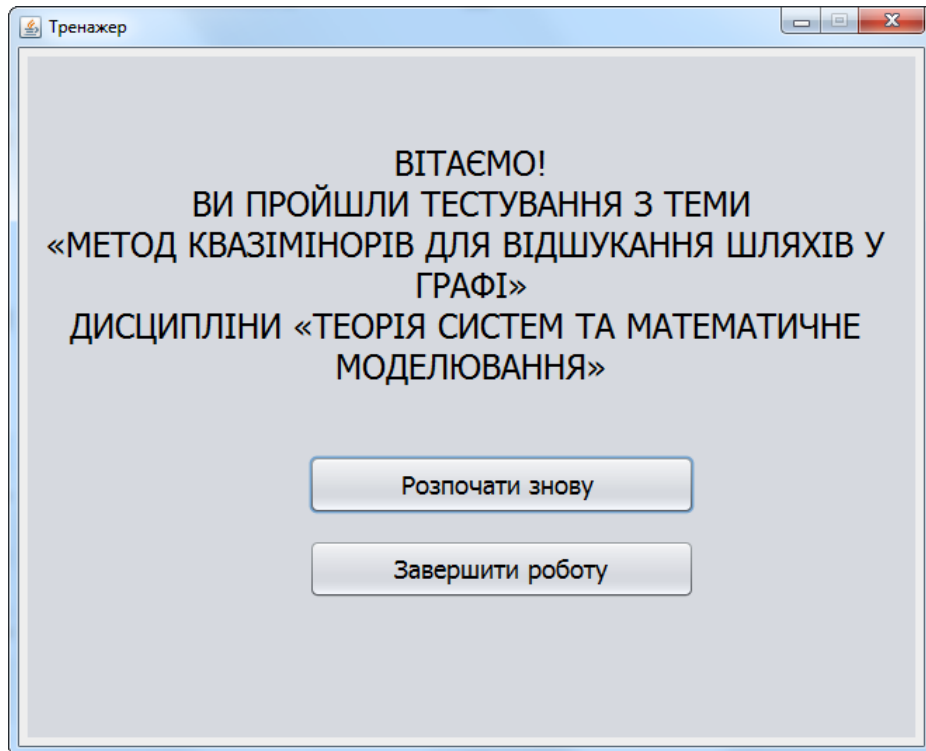


Рисунок 4.5 – Кінцева сторінка

4.3. Необхідна користувачу програми інструкція

Після запуску тренажеру можна відкрити теоретичний матеріал, натиснувши кнопку «Відкрити теорію» (рис. 4.6). Щоб розпочати тестування потрібно натиснути «Розпочати тестування».

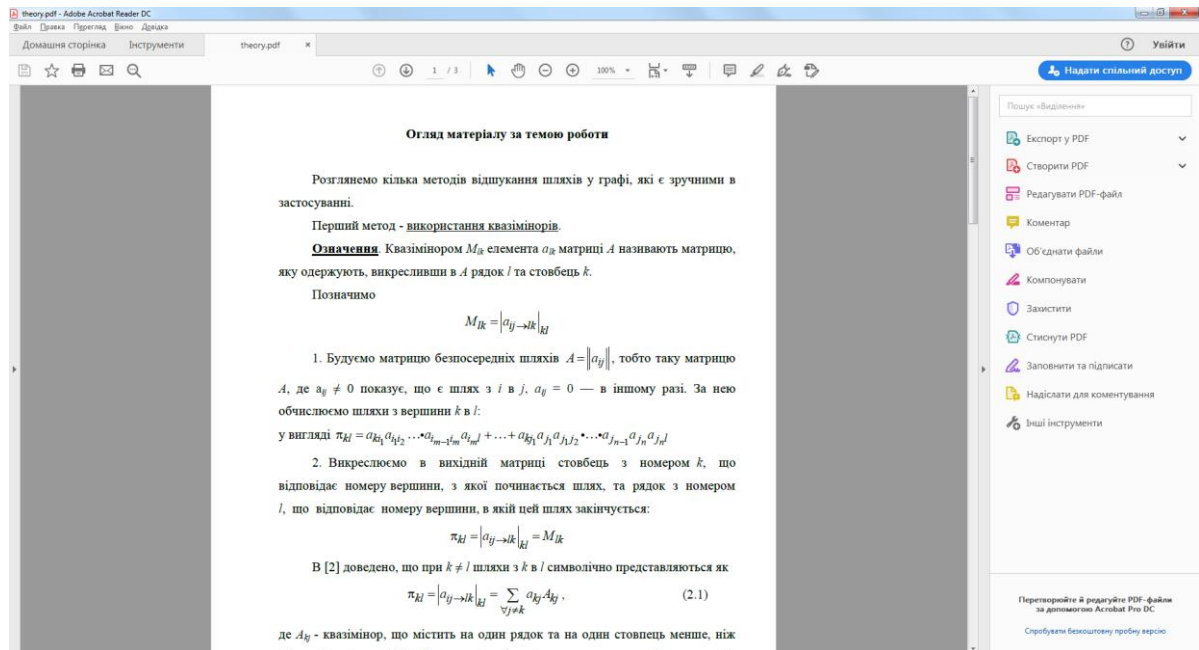


Рисунок 4.6 – Файл з теоретичним матеріалом

На першому кроці (рис. 4.2) необхідно вибрати один з варіантів відповіді. Якщо відповідь невірна, то відображається підказка (рис. 4.4). Якщо вірна – виведеться наступне завдання (рис. 4.7).

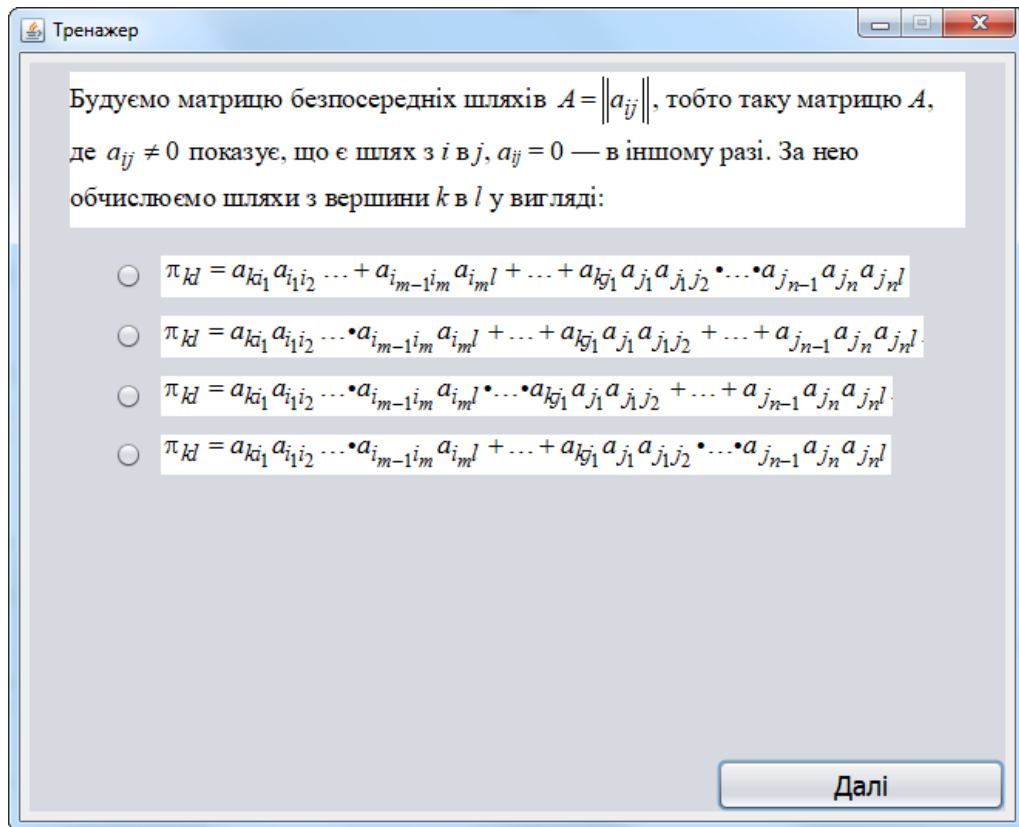


Рисунок 4.7 – Завдання другого кроку

На 4-му кроці вже потрібно ввести відповідь у комірці (рис. 4.3).

Решта кроків аналогічні вже розглянутим.

На кінцевій сторінці пропонується розпочати тестування заново, кнопка «Розпочати знову». При цьому виведеться перший крок, всі вибрані і введені раніше відповіді зітруться.

Для закриття тренажеру необхідно натиснути «Завершити роботу».

ВИСНОВКИ

Розроблено навчальний тренажер з теми «Метод квазімінорів для відшукування шляхів у графі», як складову дистанційного курсу «Теорія систем та математичне моделювання» та виконано наступні завдання роботи:

- розглянуто сутність та переваги дистанційного навчання;
- розглянуто відоме програмне забезпечення для дистанційного навчання;
- розглянуто матеріал за темою роботи;
- розглянуто специфіку застосування методу квазімінорів для відшукування шляхів у графі;
- розроблено алгоритм тренажеру, що дозволить закріпити знання з теми;
- обґрунтовано вибір програмного забезпечення;
- описано процес програмної реалізації;
- описано роботу тренажеру.

Щоб допомогти студенту у проходженні тренінгу розроблено можливість звернутися до теоретичного матеріалу з теми. В алгоритмі покроково описано всі дії при розв'язанні прикладів, де необхідно застосувати метод квазімінорів для відшукування шляхів у графі.

Розроблений тренажер містить наступні елементи:

- стартову сторінку;
- окремий файл з теоретичним матеріалом;
- сторінки із завданнями;
- кінцеву сторінку.

Якщо користувач невірно відповість на завдання, то виводиться підказка.

При завершенні проходження тренажеру реалізовано можливість розпочати проходження знову або завершити його.

Результати роботи впроваджені в дистанційному навчальному курсі
«Теорія систем та математичне моделювання».

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ємець О. О. Методичні рекомендації до виконання дипломної роботи для студентів ступеня магістра спеціальності 122 «Комп'ютерні науки» / О.О.(Олег) Ємець. – Полтава : РВВ ПУЕТ, 2018. – 35 с.
2. Ємець О. О. Методичні рекомендації щодо оформлення пояснювальних записок до курсових проектів (робіт) для студентів напряму підготовки «Інформатика» і спеціальності «Соціальна інформатика» / О. О. Ємець, Ол-ра О. Ємець. – Полтава : РВВ ПУЕТ, 2014. – 68 с.
3. Кудрявцева С.П. Міжнародна інформація : навчальний посібник / С.П. Кудрявцева, В.В. Колос. – К. : Видавничий дім «Слово», 2005. – 400 с.
4. Морзе Н.В. Інформаційні технології в навчанні : навч. посіб. / за ред. Н.В. Морзе. – К. : Видавнича група ВНУ, 2004. – 240 с
5. Положення про дистанційне навчання (Затверджено наказом Міністерства освіти і науки України 21.01.2004 № 40) [Електронний ресурс] // Режим доступу : <http://zakon4.rada.gov.ua/laws/show/z0703-13#n18>
6. Фоменко Н.А. Правова педагогіка / Н.А. Фоменко, М.І. Скрипник, О.В. Фатхутдінова. – Херсон: Олді-плюс, 2015. – 326 с.
7. Болюбаш Н.М. Фактори та умови формування професійної компетентності майбутніх економістів засобами інформаційного середовища Moodle [Електронний ресурс] / Н.М. Болюбаш // Інформаційні технології і засоби навчання. – 2010. – № 3 (17). – Режим доступу : <http://www.ime.eduua.net/em17/emg.html>
8. Самостоятельная работа студента в условиях дистанционного обучения : методические рекомендации / С.С. Рошаль, Т.Ю. Семенова ; [под ред. В.Н. Фокиной]. – М. : СГА, 2004. – 82 с.
9. Осадча К.П. Організаційні проблеми впровадження системи управління курсами у процесі професійної підготовки майбутніх фахівців з інформаційних технологій [Електронний ресурс] / К.П. Осадча, В.В.

Осадчий. – Режим доступу : <http://2013.moodlemoot.in.ua/course/view.php?id=24&lang=ru>.

10. Дистанційний курс «Інформаційні мережі» // Головний центр дистанційного навчання вищого навчального закладу УКООПСПЛКИ «Полтавський університет економіки і торгівлі». – Режим доступу: <http://www2.el.puet.edu.ua/st/course/view.php?id=1004>
11. Java [Електронний ресурс] // Матеріал з Вікіпедії — вільної енциклопедії. – Режим доступу: <http://uk.wikipedia.org/wiki/Java>.
12. Бібліографічний запис. Бібліографічний опис. Загальні вимоги та правила складання: ДСТУ 7.1-2006. – [Чинний від 2007-07-01]. – К. : Держспоживстандарт України, 2007. – 47 с.