

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

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

Допускається до захисту

Завідувач кафедри _____ О.О. Ємець
(підпис)

« _____ » _____ 2020 р.

**ПОЯСНЮВАЛЬНА ЗАПИСКА
ДО БАКАЛАВРСЬКОЇ РОБОТИ**

на тему

**РОЗРОБКА ТРЕНАЖЕРУ З ТЕМИ «ВИЗНАЧЕННЯ РАНГУ ЕЛЕМЕНТУ
СИСТЕМИ» ДИСЦИПЛІНИ «ТЕОРІЯ СИСТЕМ ТА МАТЕМАТИЧНЕ
МОДЕЛЮВАННЯ»**

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

Виконавець роботи Тітар Євгеній Васильович

_____ « _____ » _____ 2020 р.
(підпис)

Науковий керівник доцент, канд. фіз.-мат. наук Олексійчук Ю. Ф.

_____ « _____ » _____ 2020 р.
(підпис)

ПОЛТАВА 2020 р.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ

I ТЕРМІНІВ.....	3
ВСТУП.....	4
1 ПОСТАНОВКА ЗАДАЧІ.....	6
1.1 Змістовна постановка.....	6
1.2 Модель задачі та її характеристика.....	7
2 ІНФОРМАЦІЙНИЙ ОГЛЯД.....	10
2.1 Огляд робіт, де розглянуто аналогічне до теми роботи завдання.....	10
2.2 Позитивні аспекти оглянутих робіт.....	10
2.3 Вади розробок з оглянутих робіт.....	11
2.4 Необхідність та актуальність теми роботи.....	12
3 ТЕОРЕТИЧНА ЧАСТИНА.....	16
3.1 Алгоритмізація задачі за темою роботи.....	16
3.2 Розробка блок-схем, яка підлягає програмуванню.....	18
3.3 Обґрунтування вибору програмних засобів для реалізації завдання роботи.....	18
4 ПРАКТИЧНА ЧАСТИНА.....	25
4.1 Опис процесу програмної реалізації.....	25
4.2 Опис програми.....	28
4.3 Перевірка валідності (тестування всіх процедур). Дослідження можливостей програмної реалізації.....	28
4.4 Необхідна користувачу програми інструкція.....	37
ВИСНОВКИ.....	38
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	39
ДОДАТОК А. КОД ПРОГРАМИ.....	42

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ І ТЕРМІНІВ

Умовні позначення, символи, скорочення, терміни	Пояснення умовних позначень, символів, скорочень
A	Матриця суміжності.
c_{ij}	Елементи матриці.
P_i	Ранг i -ого елемента матриці.
\tilde{P}_i	Нормований ранг i -ого елемента матриці.

ВСТУП

Дистанційне навчання представляє собою нову організацію освіти, що ґрунтується на використанні як кращих традиційних методів отримання знань, так і нових інформаційних та телекомунікаційних технологій, а також на принципах самоосвіти. Використання технології дистанційного навчання слугує для розробки навчальних дисциплін, структурує навчальні програми, розробляє схему взаємодії між викладачем та студентом.

Метою даної бакалаврської роботи є розробка тренажеру з теми «Визначення рангу елемента системи» дисципліни «Теорія систем та математичне моделювання».

Для досягнення мети в ході виконання роботи необхідно виконати наступні завдання:

- вивчення методичних рекомендацій та стандартів для оформлення та виконання бакалаврських робіт;
- здійснити змістову постановку задачі;
- здійснити математичну постановку задачі;
- детальне вивчення літературних джерел для визначення рангу елемента системи;
- провести інформаційний огляд з даної теми;
- розробити алгоритм навчального тренажеру для визначення рангу елемента системи;
- розробити блок-схеми алгоритму навчального тренажеру для визначення рангу елемента системи;
- обрати мову програмування та середовище для реалізації навчального тренажеру;
- здійснити програмну реалізацію навчального тренажеру для визначення рангу елемента системи;
- провести тестування всіх кроків навчального тренажеру.

Об'єктом розробки бакалаврської роботи є програмна реалізація навчального тренажеру для дистанційного навчального курсу «Теорія систем та математичне моделювання».

Предметом розробки бакалаврської роботи є навчальний тренажер дисципліни «Теорія систем та математичне моделювання» з теми «Визначення рангу елементу системи».

Методи розробки, які використані в бакалаврській роботі:

- Для створення алгоритму використано метод визначення рангу елементу системи.
- Для програмної реалізації використано середовище розробки програм NetBeans IDE 8.1 та алгоритмічну мову програмування Java.

Новизною даної бакалаврської роботи є розробка тренажеру з теми «Визначення рангу елементу системи» дисципліни «Теорія систем та математичне моделювання». Розробка даного навчального тренажеру є унікальною, бо до цього часу не знайдено жодного тренажеру з цієї теми.

Практичною цінністю реалізованого навчального тренажеру є створення програмного засобу для навчання студентів визначенню рангів елементу системи. Реалізований навчальний тренажер рекомендовано до використання студентами спеціальності «Комп'ютерні науки» дистанційного навчального курсу «Теорія систем та математичне моделювання» в ПУЕТ.

Структура пояснювальної записки до бакалаврської роботи включає в себе наступне: вступ, постановку задачі, інформаційний огляд, теоретичну та практичну частину, висновки, список літератури та додаток.

Обсяг пояснювальної записки до бакалаврської роботи містить: 42 сторінки, 31 рисунок, 1 додаток (на 65 сторінках) та 18 літературних джерел.

1 ПОСТАНОВКА ЗАДАЧІ

1.1 Змістовна постановка

Задачею даної бакалаврської роботи є розробка тренажеру з теми «Визначення рангу елементу системи» дисципліни «Теорія систем та математичне моделювання».

Для цього необхідно провести інформаційний огляд з використання тренажерів у навчанні студентів у Вищих навчальних закладах. Також провести аналіз вивчення теми «Визначення рангу елементу системи» у навчанні студентів. Знайти та оглянути навчальні тренажери подібні до завдання даної роботи.

Навчальний тренажер має бути створений за лекціями дистанційного навчального курсу «Теорія систем та математичне моделювання» [1].

Припустимо вимоги до розробки навчального тренажеру:

1. Обґрунтувати вибір мови та середовище розробки для навчального тренажеру.

2. Розробити алгоритм навчального тренажеру з теми «Визначення рангу елементу системи» відповідно до лекцій та прикладів [1] з дистанційного навчального курсу «Теорія систем та математичне моделювання».

3. Розробити блок-схеми навчального тренажеру, які відповідають розробленому алгоритму.

4. Запрограмувати навчальний тренажер, який відповідає блок-схемам та розробленому алгоритму, а також реалізований в обраному програмному середовищі з відповідною мовою програмування.

5. Провести тестування всіх кроків навчального тренажеру, виявити неполадки та виправити їх.

6. Завантажити реалізований навчальний тренажер в дистанційний навчальний курс «Теорія систем та математичне моделювання».

Припустимо вимоги до роботи навчального тренажеру:

1. Розробити легкий та зрозумілий у використанні дизайн навчального тренажеру.

2. Створити перевірки відповідей введених студентом.
3. Реалізувати повідомлення про результати перевірки відповідей.
4. Розробити повідомлення з підказками для студентів.
5. Розробити декілька прикладів по рівню складності.

1.2 Модель задачі та її характеристика

Для практичної частини, використано наступні три приклади:

Приклад 1. Визначити ранг елементу системи, яка представлена на рис. 1.1.

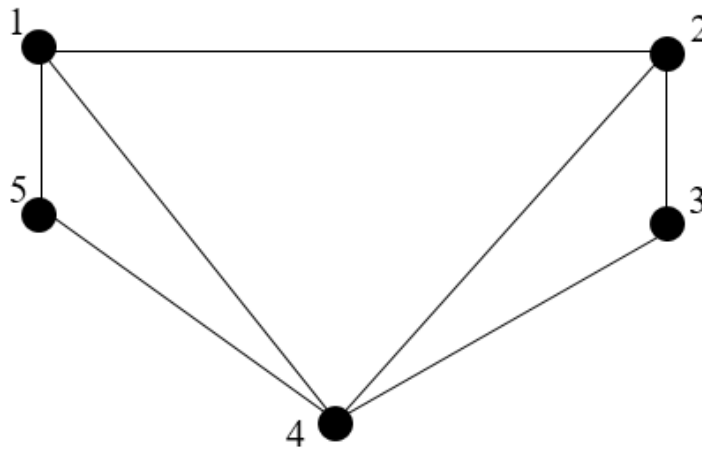


Рисунок 1.1 – Умова прикладу 1

Розв'язок прикладу:

$$A = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \end{pmatrix}; A^2 = \begin{pmatrix} 3 & 1 & 2 & 2 & 1 \\ 1 & 3 & 1 & 2 & 2 \\ 2 & 1 & 2 & 1 & 1 \\ 2 & 2 & 1 & 4 & 1 \\ 1 & 2 & 1 & 1 & 2 \end{pmatrix}$$

$$P = A + A^2 = \begin{pmatrix} 3 & 2 & 2 & 3 & 2 \\ 2 & 3 & 2 & 3 & 2 \\ 2 & 2 & 2 & 2 & 1 \\ 3 & 3 & 2 & 4 & 2 \\ 2 & 2 & 1 & 2 & 2 \end{pmatrix}$$

$$P_1 = P_2 = 12; P_3 = P_5 = 9; P_4 = 14$$

$$\tilde{P}_1 = \tilde{P}_2 = 0,21; \tilde{P}_3 = \tilde{P}_5 = 0,16; \tilde{P}_4 = 0,25$$

Приклад 2. Визначити ранг елементу системи, яка представлена на рис. 1.2.

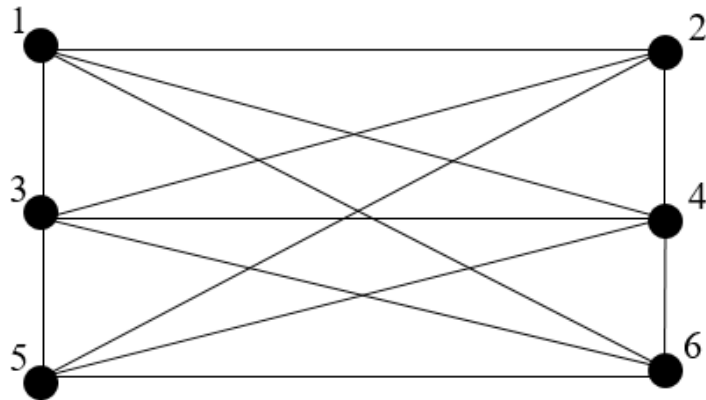


Рисунок 1.2 – Умова прикладу 2

Розв'язок прикладу:

$$A = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 \end{pmatrix}; A^2 = \begin{pmatrix} 4 & 2 & 3 & 3 & 4 & 2 \\ 2 & 4 & 3 & 3 & 2 & 4 \\ 3 & 3 & 5 & 4 & 3 & 3 \\ 3 & 3 & 4 & 5 & 3 & 3 \\ 4 & 2 & 3 & 3 & 4 & 2 \\ 2 & 4 & 3 & 3 & 2 & 4 \end{pmatrix}$$

$$P = A + A^2 = \begin{pmatrix} 4 & 3 & 4 & 4 & 4 & 3 \\ 3 & 4 & 4 & 4 & 3 & 4 \\ 4 & 4 & 5 & 5 & 4 & 4 \\ 4 & 4 & 5 & 5 & 4 & 4 \\ 4 & 3 & 4 & 4 & 4 & 3 \\ 3 & 4 & 4 & 4 & 3 & 4 \end{pmatrix}$$

$$P_1 = P_2 = P_5 = P_6 = 22; P_3 = P_4 = 26 \\ \tilde{P}_1 = \tilde{P}_2 = \tilde{P}_5 = \tilde{P}_6 = 0,16; \tilde{P}_3 = \tilde{P}_4 = 0,19$$

Приклад 3. Визначити ранг елементу системи, яка представлена на рис. 1.3.

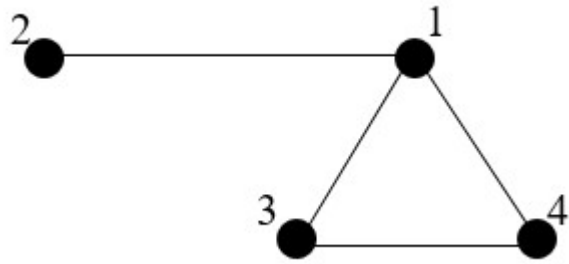


Рисунок 1.3 – Умова прикладу 3

Розв'язок прикладу:

$$A = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}; A^2 = \begin{pmatrix} 3 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 2 & 1 \\ 1 & 1 & 1 & 2 \end{pmatrix}$$

$$P = A + A^2 = \begin{pmatrix} 3 & 1 & 2 & 2 \\ 1 & 1 & 1 & 1 \\ 2 & 1 & 2 & 2 \\ 2 & 1 & 1 & 2 \end{pmatrix}$$

$$P_1=8; P_2=4; P_3=P_4=7$$

$$\tilde{P}_1=0,3; \tilde{P}_2=0,15; \tilde{P}_3=\tilde{P}_4=0,27$$

2 ІНФОРМАЦІЙНИЙ ОГЛЯД

2.1 Огляд робіт, де розглянуте аналогічне до теми завдання

Під час пошуку навчальних тренажерів за подібний до завдання даної роботи [2-4], було знайдено наступні навчальні тренажери з тем «Машина Т'юрінга», «Гradientний метод оптимізації нелінійних функцій», «Аналіз сортування вставками», «Метод Ленда та Дойга» та «Задача директора» [5-9].

Огляд робіт розпочато з навчального тренажеру Кондрашева Д. М. на тему «Розробка тренажера дистанційного навчального курсу «Математична логіка та теорія алгоритмів» з теми «Машина Т'юрінга» [5].

Наступним було розглянуто навчальний тренажер Потерайла О. О. з теми «Програмування навчального тренажера для gradientного методу оптимізації нелінійних функцій дистанційного навчального курсу «Методи оптимізації та дослідження операцій» [6].

Далі розглянуто навчальний тренажер Русіна В. С. на тему «Програмна реалізація елемента тренажеру з теми «Аналіз алгоритму сортування вставками» дисципліни «Аналіз алгоритмів» [7].

Одним з наступних був навчальний тренажер Сивоконя О. Ю. «Розробка та програмна реалізація тренажера з теми «Метод Ленда та Дойга» дистанційного курсу «Методи оптимізації та дослідження операцій» [8].

Останнім було розглянуто тренажер на тему «Алгоритмізація та програмування елемента тренажера з побудови математичної моделі комбінаторної оптимізації «Задачі директора» студента Куркіна В. В. [9].

2.2 Позитивні аспекти оглянутих робіт

Проаналізувавши всі розглянуті роботи, які були створені для дистанційних курсів [5-9], було виявлено наступні позитивні аспекти навчальних тренажерів:

- Навчальні тренажери полегшують роботу викладачам, як при викладі матеріалу так і при перевірці знань студентів.
- Навчальні тренажери були реалізовані на мовах програмування, як Java, Ruby, C#, JavaScript, що дозволяє легко та без перешкод впровадити їх в дистанційні навчальні курси.
- Навчальні тренажери дають можливість студентові отримати, перевірити та закріпити знання і навички реалізованих тем.
- Покращують вміння студента аналізувати та використовувати отриману інформацію.
- Навчальні тренажери створюють новий спосіб отримання інформації та перевірки знань, як новий метод у навчанні студентів.
- За короткий проміжок часу навчальні тренажери дають змогу охопити більше практичних прикладів для розв'язку.

2.3 Вади розробок з оглянутих робіт

При аналізі всіх розглянутих робіт, які були створенні для дистанційних курсів [5-9], було виявлено також наступні вади навчальних тренажерів:

- Неможливість завантаження всіх навчальних тренажерів на персональний комп'ютер студенту.
- Не всі навчальні тренажери мають довідку з теоретичним матеріалом.
- Навчальні тренажери не мають можливості навчати іноземних студентів.
- Не всі навчальні тренажери мають можливість пропуск завдання або повернення до попереднього. Також не мають функції запам'ятовування останнього місця зупинки студентом.
- Деякі навчальні тренажери містять підказки, навіть в легких завданнях, що не дає можливості аналізувати студентові свої помилки при виконанні поставлених завдань.

2.4 Необхідність та актуальність даної теми

Одним із наслідків інформатизації суспільства та постійного зниження вартості послуг на використання глобальної мережі Internet стало суттєво поглиблення процесів впровадження інформаційних технологій в освітню практику [10].

Історично дистанційне навчання виникло у 1840 році, коли Ісаак Пітман запропонував навчання через поштовий зв'язок студентам Англії [11]. Інтенсивний розвиток дистанційної освіти у Європі і Америці розпочався в 70-і роки. В Україні дистанційна форма освіти запроваджується з 2000 року і регулюється Концепцією розвитку дистанційної освіти в Україні і Положенням про дистанційну освіту МОН України.

Досить часто дистанційне навчання плутають із заочною освітою, але це не одне і теж. Якщо, навчаючись на заочному, студент зустрічається з викладачем на вичитування лекцій та іспитах, а основний матеріал освоює самостійно і до всього доходить сам, то дистанційне навчання має на увазі, що обидві сторони знаходяться на зв'язку практично постійно і спілкуються за допомогою передових технологій.

Головною метою створення системи дистанційної освіти є забезпечення загальнонаціонального доступу до освітніх ресурсів шляхом використання сучасних інформаційних технологій та телекомунікаційних мереж і створення умов для реалізації громадянами своїх прав на освіту.

Дистанційне навчання забезпечує можливість навчатися тоді, коли зручно студенту, у тому темпі, що він сам обирає (в рамках установлених строків проведення курсів), в тому місці де він перебуває (не має потреби витратити час на дорогу до ВНЗ, для здачі поточних, а іноді і підсумкових контролів). Тому до контингенту потенційних студентів дистанційної форми навчання можна віднести тих, хто часто перебуває у відрядженнях, військовослужбовців, територіально віддалених слухачів, жінок, що перебувають у декретній відпустці, людей з фізичними вадами, тих, хто поєднує навчання й роботу, співробітників, що підвищують свою кваліфікацію [12].

Варто зазначити, що дистанційне навчання в сучасному світі здійснюється за допомогою різноманітних технологій, які відрізняються за: формою подання учбових матеріалів; наявністю посередника в системі навчання або за централізованою формою навчання; за ступенем використання телекомунікацій і персональних комп'ютерів; за технологією організації контролю учбового процесу; за ступенем впровадження в технології навчання звичайних методів ведення освітнього процесу; за методами ідентифікації студентів при складанні іспитів [13].

Вивчення досвіду впровадження дистанційної форми навчання в системи освіти різних країн дозволило виділити основні мотиваційні причини, що спонукають навчальні заклади до організації системи дистанційного навчання: поліпшення якості навчання; переваги нових педагогічних технологій; зростаючий попит на нову форму навчання; отримання доходів; можливість постійної взаємодії викладачів і студентів; необхідність виживання в інформаційному суспільстві; можливість скорочення витрат на реорганізацію освіти [14].

Незважаючи на досить об'ємний перелік переваг дистанційного навчання воно має і ряд недоліків (таблиця 1.1) [15,16].

Таблиця 1.1 Переваги та недоліки дистанційного навчання для студентів

Переваги	Недоліки
Навчання у відповідності до власного темпу, особистісних особливостей та освітніх потреб.	Відсутність особистісного спілкування між викладачем та студентом (відбувається менш ефективна, безособистісна передача знань).
Не обмежений вибір навчального закладу та освітніх можливостей, незалежно від місцеперебування.	Не вистачає спілкування з колегами і студентами для обміну досвідом
Використання під час процесу навчання сучасні технології, тобто, паралельно засвоювати навички, які згодом знадобляться під час роботи.	Необхідність наявності у студента сильної особистісної мотивації, вміння навчатися самостійно, без постійної підтримки та підштовхування з боку викладача
Самостійне планування часу занять.	Відсутність можливості негайного

	практичного застосування отриманих знань із наступним обговоренням виниклих питань з викладачем і роз'яснення ситуації на конкретних прикладах
Навчання у найбільш приємній та сприятливій атмосфері, котра створюється самостійно.	Студенти не завжди можуть забезпечити себе достатнім технічним обладнанням - мати комп'ютер та постійний вихід у Інтернет.

Крім того, досить вагомою проблемою є низька пропускна спроможність електронної мережі під час навчальних чи екзаменаційних телеконференцій. Від цього, передовсім, страждають студенти невеликих містечок України, яким, власне, найбільше підходить дистанційна освіта через географічну віддаленість від наукових осередків.

Впровадження повноцінного дистанційного навчання в Україні гальмується багатьма чинниками. Це пояснюється слабким проробленням методологічних та психолого-педагогічних особливостей дистанційної освіти, дуже високими вимогами до «віртуального» викладача, який крім звичайних знань має вміння користуватися засобами інформаційних і комунікаційних технологій. Можна виділити найважливіші проблеми, що безпосередньо відбиваються на методології та якості дистанційної підготовки [17]:

1. недосконалість, а то і відсутність нормативно-правового і організаційно-методичного забезпечення;
2. недостатня кваліфікація викладачів та проблема їх підготовки й перепідготовки;
3. надмірний бюрократизм дистанційного навчання;
4. фінансування розробки дистанційних технологій, оновлення комп'ютерної техніки та матеріальної бази, забезпечення доступу до Internet мережі викладачів;
5. інформованості населення про дистанційне навчання;

б. відсутність у деяких потенційних студентів, що проживають у сільських населених пунктах, відповідного технічного оснащення та можливості доступу до мережі Інтернет.

Таким чином, розвиток дистанційного навчання в Україні не відповідає вимогам інформаційного суспільства, що в першу чергу, викликано недосконалим законодавчим та нормативним забезпеченням. Для того, щоб система дистанційного навчання зайняла гідне місце в системі освіти України, потрібно вирішити зазначенні вище проблеми. А насамперед створити глобальну комп'ютерну мережу освіти й науки, оскільки саме комп'ютер дає змогу отримувати навчальний матеріал, є водночас і бібліотекою, і центром довідкової інформації, і комунікативним центром.

3 ТЕОРЕТИЧНА ЧАСТИНА

3.1 Алгоритмізація задачі за темою роботи

При запуску навчального тренажера перед студентом з'являється початковий екран, який містить: назву навчального закладу, тему навчального тренажера, а саме «Визначення рангу елемента системи», відомості про автора. Також початковий екран містить три кнопки для обрання складності прикладу, а саме: кнопка «Низький рівень» умова прикладу 3 (рис. 1.3), кнопка «Середній рівень» умова прикладу 1 (рис. 1.1) та кнопка «Складний рівень» умова прикладу 2 (рис. 1.2).

Після того як студент обрав складність прикладу для подальшого визначення рангу елемента системи перед ним з'являється умова даного прикладу.

Алгоритм тренажера наступний:

Крок 1-й: Перед студентом з'являється перше завдання «Складіть за умовою прикладу матрицю суміжності».

Після заповнення матриці студентом та натисненні на кнопку «Перевірити» перед ним з'являється повідомлення про невірність хоча б однієї відповіді. Якщо відповіді всі вірні, відбувається перехід на наступний крок.

Також у студент присутня кнопка «Допомога». Якщо бракує знань з побудови матриці суміжності, то при натисканні на дану кнопку, студент отримає наступне повідомлення «Матриця суміжності графа G зі скінченною кількістю вершин n (пронумерованих числами від 1 до n) – це квадратна матриця A розміру n , в якій значення елемента a_{ij} рівне числу ребер з i -ї вершини графа в j -у вершину».

Крок 2-й: Перед студентом відкривається наступне вікно, де наведена матриця суміжності із попереднього кроку, а також стоїть завдання «Помножити матрицю A на матрицю A ». Студенту необхідно обчислити кожен елемент нової матриці. Після всіх розрахунків відбувається перевірка і перехід на наступний крок, як і в минулому кроці.

При натисканні на кнопку «Допомога» з'являється повідомлення наступного вигляду «Добуток матриць $A \cdot B$ складається з усіх можливих комбінацій скалярних

добутків вектор-рядків матриці A і вектор-стовпців матриці B . Елемент матриці $A \cdot B$ з індексами i, j є скалярним добутком i -го вектор-рядка матриці A і j -го вектор-стовпця матриці B ».

Крок 3-й: На даному етапі перед студентом стоїть завдання додати дві попередньо отримані матриці, відповідно результуючу матрицю заповнити. Після заповнення відповідними обчисленнями відбувається перевірка введених даних та перехід на наступний крок алгоритму.

Також є повідомлення при натисканні на кнопку «Допомога», яке має наступний вигляд «Сумою двох матриць A та B буде матриця з такою ж кількістю рядків та стовпців, що й у початкових матрицях. Сума A та B , записується як $A+B$, розраховується за допомогою додавання відповідних елементів A та B ».

Крок 4-й: Перед студентом з'являється обчислена матриця з попереднього кроку. На цьому кроці, студенту необхідно обрахувати суму кожного рядка матриці. Введені дані перевіряються і якщо всі вірні відбувається перехід на наступний крок, інакше виправлення помилок.

Повідомлення з довідки має наступний вигляд «Сума елементів i -го рядка матриці».

Крок 5-й: Виходячи із попереднього кроку алгоритму студенту необхідно записати чому дорівнює кожен ранг елементу матриці. Після перевірки студент переходить на наступний крок.

Якщо бракує знань з обрахунку рангу елементу матриці, то при натисканні на кнопку «Допомога», студент отримає наступне повідомлення «Під рангом i -го елементу розуміють суму елементів i -го рядка матриці».

Крок 6-й: На цьому кроці, студенту необхідно порахувати нормовані ранги елементу матриці. Для цього йому дається інформація з попереднього кроку та сума всіх рядків з 5-ого кроку алгоритму. Після перевірки всіх введених значень студент переходить на наступний крок.

Також є повідомлення при натисканні на кнопку «Допомога», яке має наступний вигляд «Під нормованим рангом i -го елементу розуміють ділення суми i -го рядка матриці на суму всієї матриці».

Крок 7-й: Перед студентом з'являється вікно, яке інформує, що приклад успішно розв'язаний. Студент має змогу вийти з тренажера, або повернутися до початкового екрану та спробувати ще раз розв'язати будь-який із трьох прикладів.

Даний алгоритм застосовується для всіх трьох прикладів.

3.2 Розробка блок-схеми, яка підлягає програмуванню

На рисунках 3.1-3.4 показано блок-схеми алгоритму навчального тренажера визначення рангу елементу системи. На рис. 3.1 — показана блок-схема алгоритму навчального тренажера. На рис. 3.2-3.4 – зображено блок-схему постановки завдання для кожного з трьох прикладів.

3.3 Обґрунтування вибору програмних засобів для реалізації завдання роботи

Java – об'єктно-орієнтована мова програмування, що розробляється компанією Sun Microsystems з 1991 року і офіційно випущений 23 травня 1995 року. Спочатку нову мову програмування називали Oak (James Gosling) і розроблявся для побутової електроніки, але згодом був перейменований в Java і став використовуватися для написання аплетів, додатків і серверного програмного забезпечення.

Відмінною особливістю Java в порівнянні з іншими мовами програмування загального призначення є забезпечення високої продуктивності програмування, ніж продуктивність роботи програми або ефективність використання пам'яті.

В Java використовуються практично ідентичні угоди для оголошення змінних, передачі параметрів, операторів і для управління потоком виконання коду. В Java додані всі хороші риси C++.

Три ключові елементи об'єдналися в технології мови Java:

- Java надає для широкого використання свої аплети (applets) – невеликі, надійні, динамічні, які не залежать від платформи активні мережеві додатки, що

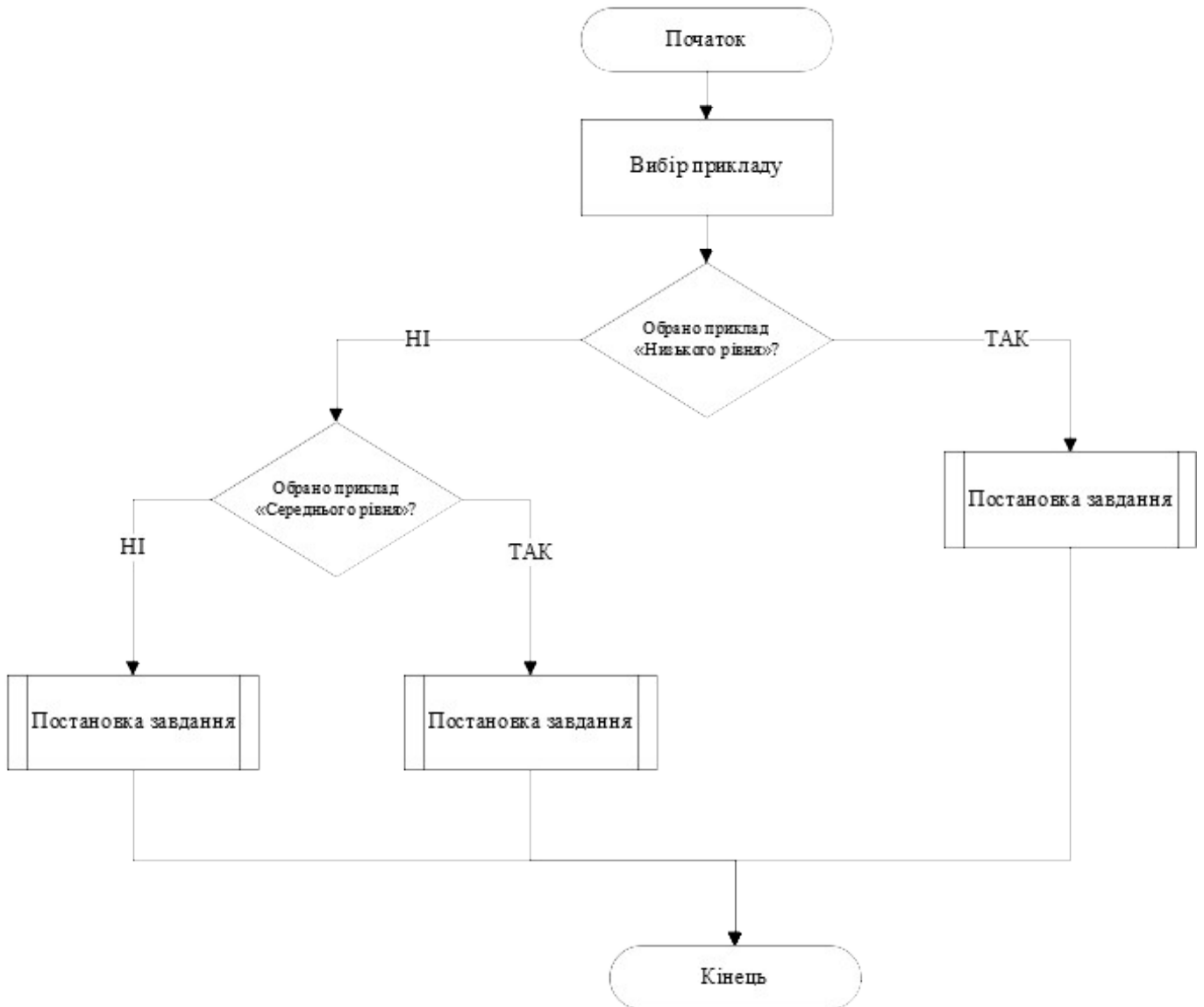


Рисунок 3.1 – Блок-схема алгоритму навчального тренажера

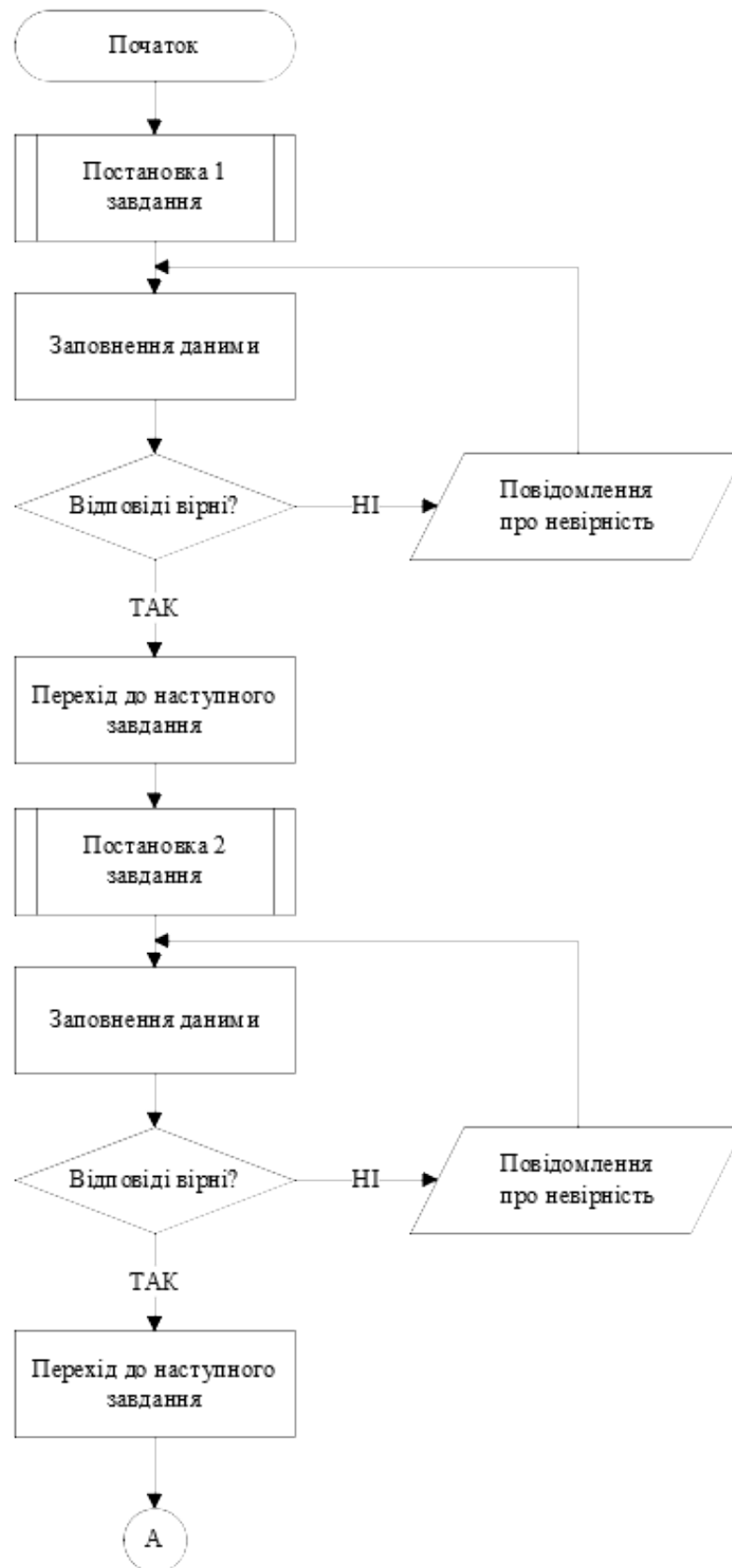


Рисунок 3.2 – Блок-схема постановки завдання

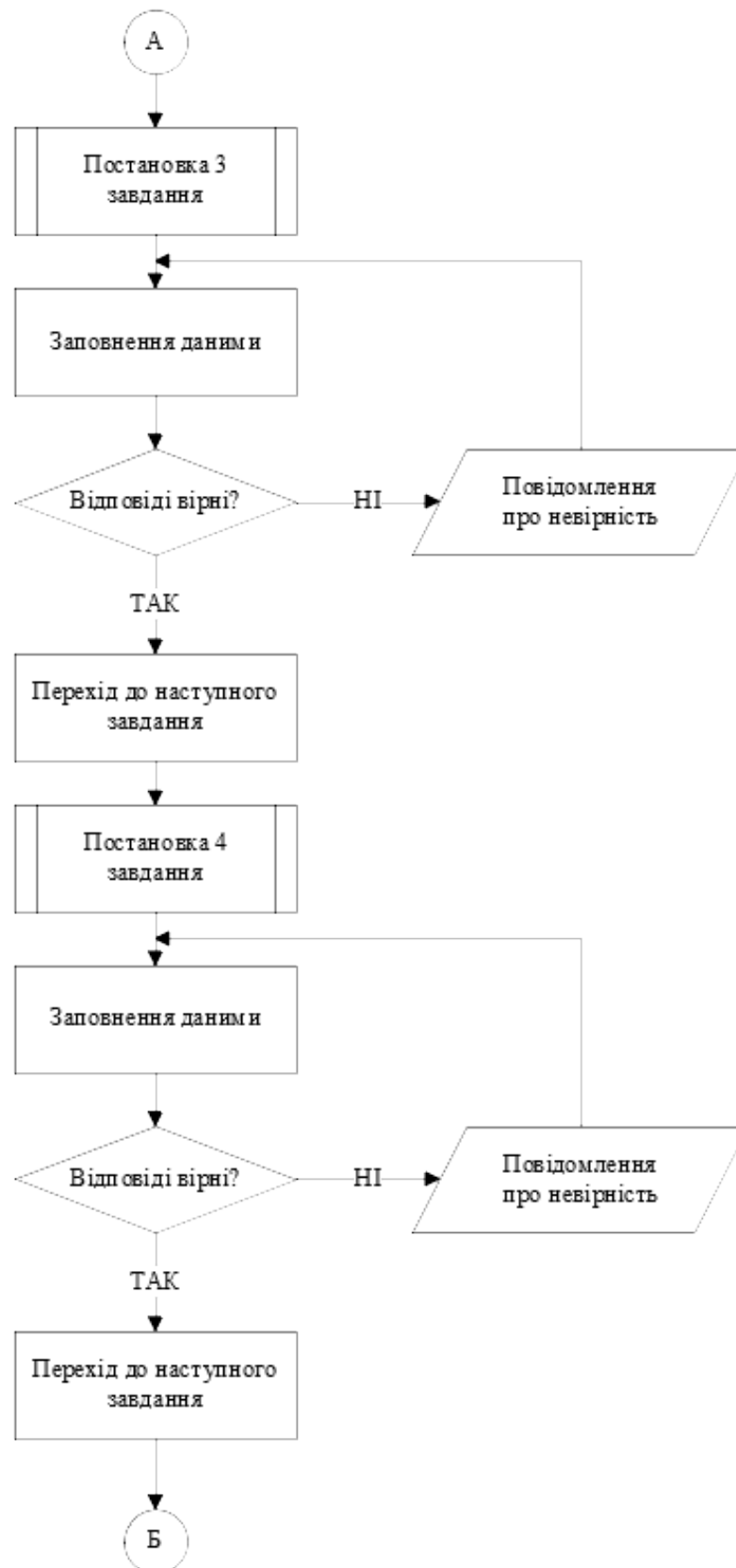


Рисунок 3.3 – Продовження блок-схеми постановки завдання

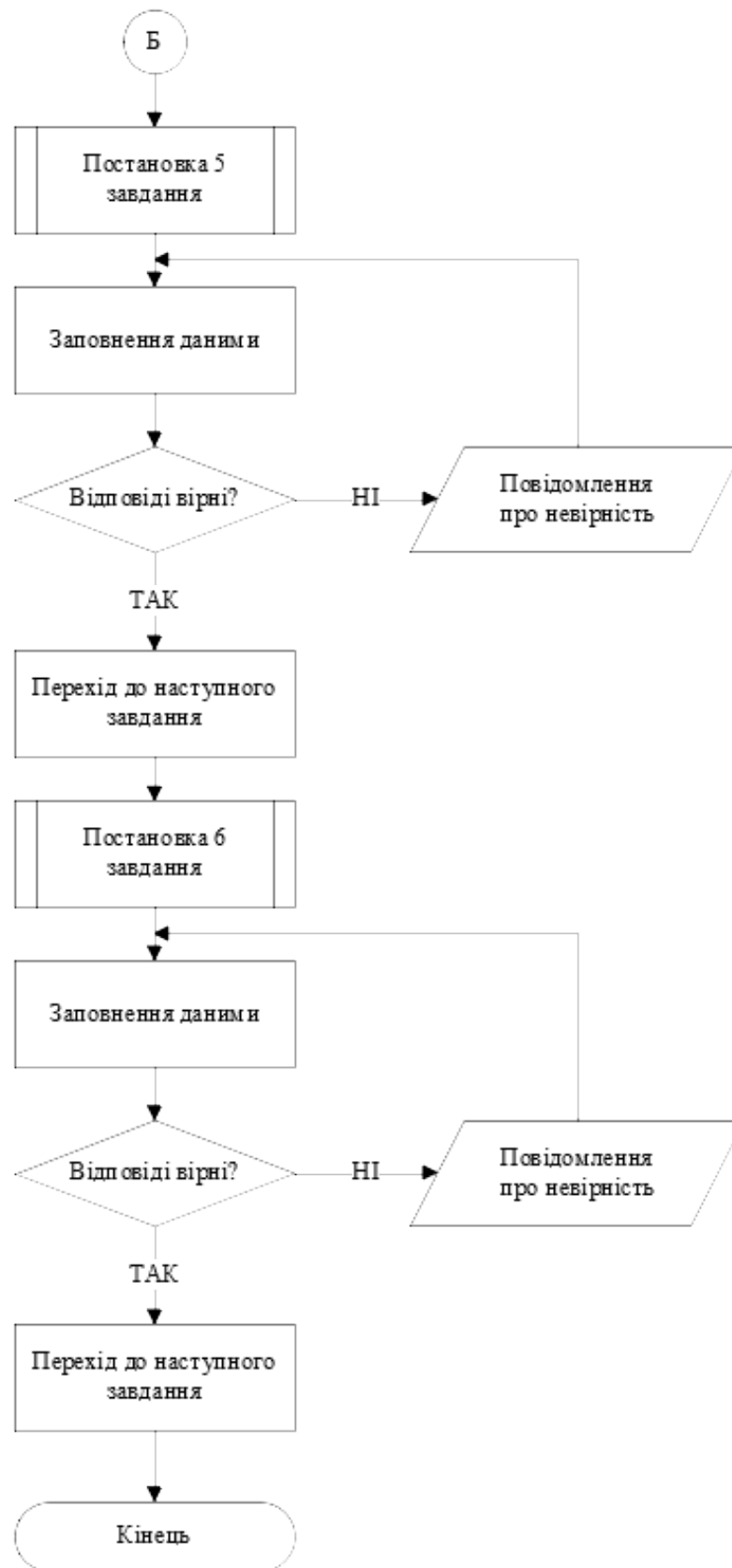


Рисунок 3.4 – Продовження блок-схеми постановки завдання

вбудовуються в сторінки Web. Аплети Java можуть налаштуватися і поширюватися споживачам з такою легкістю, як будь-які документи HTML.

- Java вивільняє міць об'єктно-орієнтованої розробки додатків, поєднуючи простий і знайомий синтаксис з надійним і зручним в роботі середовищем розробки. Це дозволяє широкому колу програмістів швидко створювати нові програми і нові аплети.

- Java надає програмісту багатий набір класів об'єктів для ясного абстрагування багатьох системних функцій, використовуваних при роботі з вікнами, мережею і для введення-виведення. Ключова риса цих класів полягає в тому, що вони забезпечують створення незалежних від використовуваної платформи абстракцій для широкого спектра системних інтерфейсів.

Величезна перевага Java полягає в тому, що на цій мові можна створювати додатки, здатні працювати на різних платформах. Вони призначені для роботи на різних платформах і не залежать від конкретного типу процесора і операційної системи.

Програми, складені на мові програмування Java, можна розділити за своїм призначенням на дві великі групи.

До першої групи належать додатки Java, які призначені для автономної роботи під управлінням спеціальної машини Java. Реалізація цієї машини створена для всіх основних комп'ютерних платформ.

Друга група – це так звані аплети. Аплети являють собою різновид програм Java, які інтерпретуються віртуальною машиною Java, вбудованої практично в усі сучасні браузері.

Додатки, що відносяться до першої групи – це звичайні автономні програми. Так як вони не містять машинного коду і працюють під управлінням спеціального інтерпретатора, їх продуктивність помітно нижча, ніж у звичайних програм, складених, наприклад, на мові програмування C++. Програми Java без перетрансляції здатні працювати на будь-якій платформі, що саме по собі має велике значення в плані розробок для Internet.

Аплети Java вбудовуються в документи HTML, що зберігаються на сервері Web. За допомогою алетів можна зробити сторінки сервера Web динамічними та інтерактивними. Аплети дозволяють виконувати складну локальну обробку даних, отриманих від сервера Web або введених користувачем з клавіатури. З міркувань безпеки аплети (на відміну від звичайних програм Java) не мають ніякого доступу до файлової системи локального комп'ютера. Всі дані для обробки вони можуть отримати тільки від сервера Web.

Отже, мова Java є об'єктно-орієнтованим і поставляється з досить об'ємною бібліотекою класів. Так само як і бібліотека класів системи розробки додатків на мові C++, бібліотеки класів Java значно спрощують розробку додатків, представляючи в розпорядження програміста потужні засоби вирішення поширених завдань. Тому програміст може більше уваги приділити вирішенню прикладних задач.

NetBeans IDE - вільна інтегрована середовище розробки додатків (IDE) на мовах програмування Java, JavaFX, Python, PHP, JavaScript, C + +, Ада та ряду інших.

Основні можливості:

- ApacheAntвикористовується в якості машини побудови.
- Метадані проекту - це скрипти побудови Ant.
- Доступна можливість побудови додатків поза інтегрованої середис допомогою Ant, при цьому не потрібно ніякого спеціального дії (на кшталт "експорту вAnt").
- Повністю інтегрована підтримка модульного тестування (JUnit).
- Вихідні файли показуються в контексті проекту в логічному вигляді.
- Користувач може працювати з декількома проектами одночасно.
- Все вищезазначене доступно прямо з коробки, від користувача не потрібно копатися в жодних налаштуваннях.

4 ПРАКТИЧНА ЧАСТИНА

4.1 Опис процесу програмної реалізації

Для виконання поставленої задачі було застосовано високорівневу мову програмування Java. Середовищем розробки виступила програма NetBeans IDE 8.1.

Реалізація програми розпочато з вивчення матеріалу лекції про визначення рангу елементів системи.

Роботу розпочато зі створення нового проекту Java Application (рис. 4.1) і заповнення його формами JFrame Form, як зображено на рисунку 4.2.

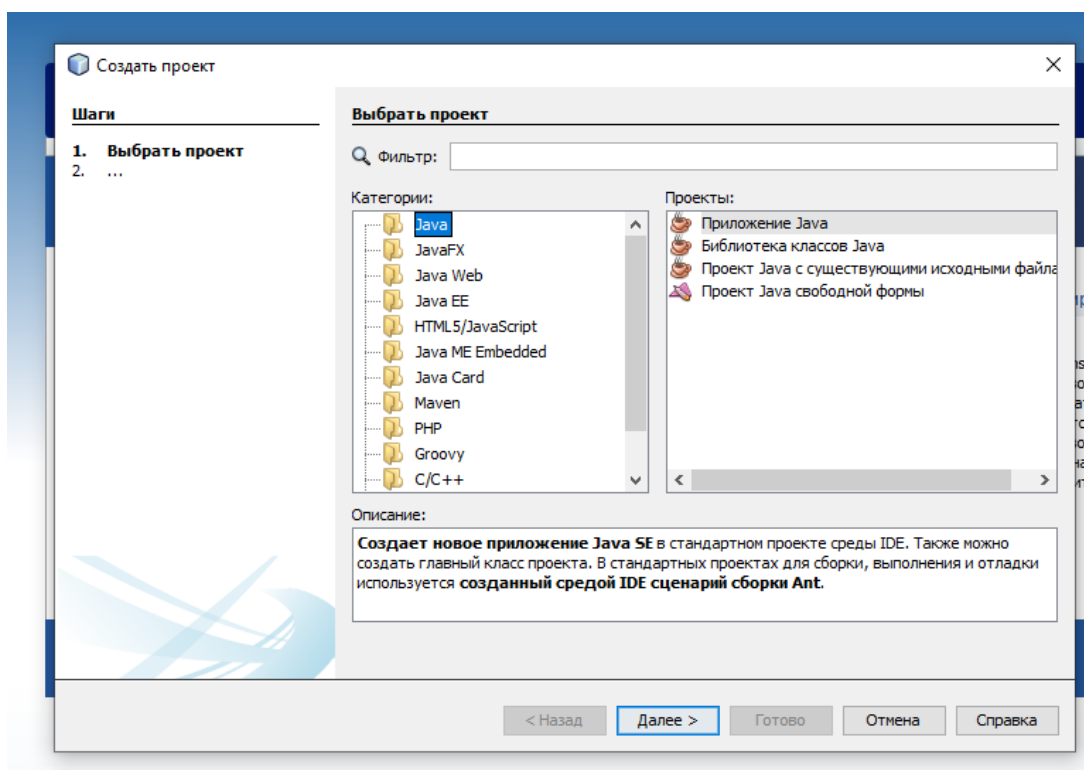


Рисунок 4.1 – Створення проекту Java Application

Після створення форми, продовжено заповнювання її компонентами, а саме, такими елементами як JButton, JLabel, JTextField.

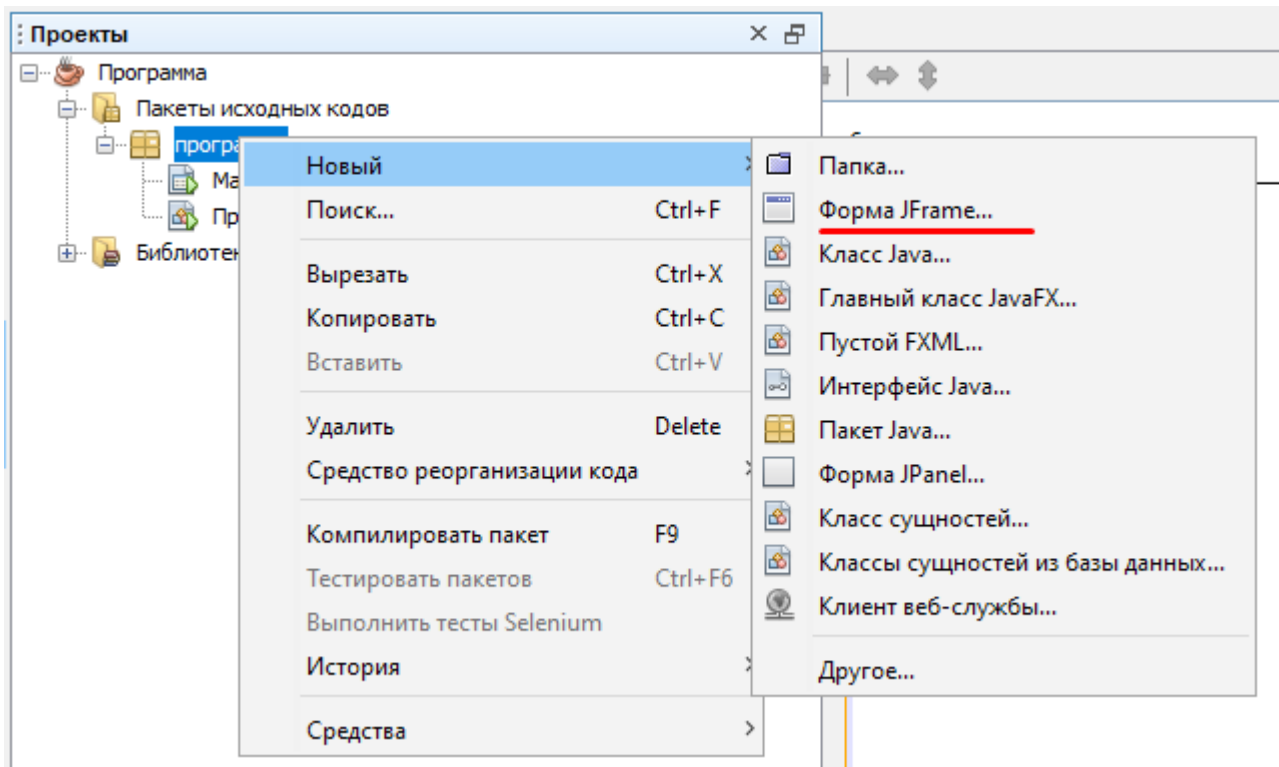


Рисунок 4.2 – Додавання форм до проекту

Оскільки в програмі проводяться перевірки на вірність введених даних, було прийнято рішення виконати цю перевірку через оператор *if*. Нижче на рисунку 4.4 наведений приклад реалізації перевірки правильності відповіді.

```

if ((a1 == 2) && (a2 == 2) && (a3 == 2) && (a4 == 2)
    && (a5 == 0) && (a6 == 2) && (a7 == 0) && (a8 == 0)
    && (a9 == 2) && (a10 == 1) && (a11 == 1) && (a12 == 2)
    && (a13 == 2) && (a14 == 1) && (a15 == 2) && (a16 == 2)) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new F_4().setVisible(true);
            dispose();
        }
    });
} else {
    JOptionPane.showMessageDialog(null, "Не вірно!", "Помилка", JOptionPane.WARNING_MESSAGE);
}

```

Рисунок 4.3 –Перевірки вибраної відповіді

Також було реалізовано вікно-повідомлення, для індикації правильності відповіді за допомогою `JOptionPane`. Приклад приведений на рисунку 4.4 та рис. 4.6.

```

'''
} else {
    JOptionPane.showMessageDialog(null, "Не вірно!", "Помилка", JOptionPane.WARNING_MESSAGE);
,

```

Рисунок 4.4 – Застосування JOptionPane для вікна про неправильну відповідь

Для реалізації підказок з кнопки «Допомога» також було реалізовано вікна через JOptionPane, один з прикладів представлено на рисунку 4.5 та рисунок 4.7.

```

JOptionPane.showMessageDialog(null,
    "Матриця суміжності графа G зі "
    + "скінченною кількістю вершин n \n"
    + "(прономерованих числами від 1 до n) "
    + "— це квадратна матриця A розміру n, \n"
    + "в якій значення елемента aij рівне "
    + "числу ребер з i-ї вершини графа "
    + "в j-у вершину.",
    "Помилка",
    JOptionPane.WARNING_MESSAGE);

```

Рисунок 4.5 – Застосування JOptionPane для вікна з підказкою

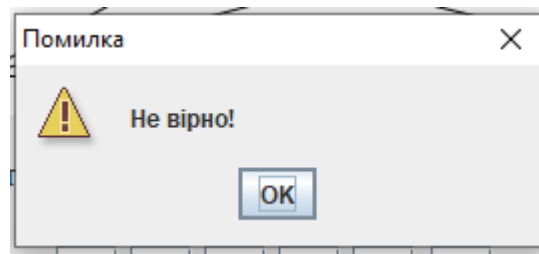


Рисунок 4.6 – Вікно-повідомлення

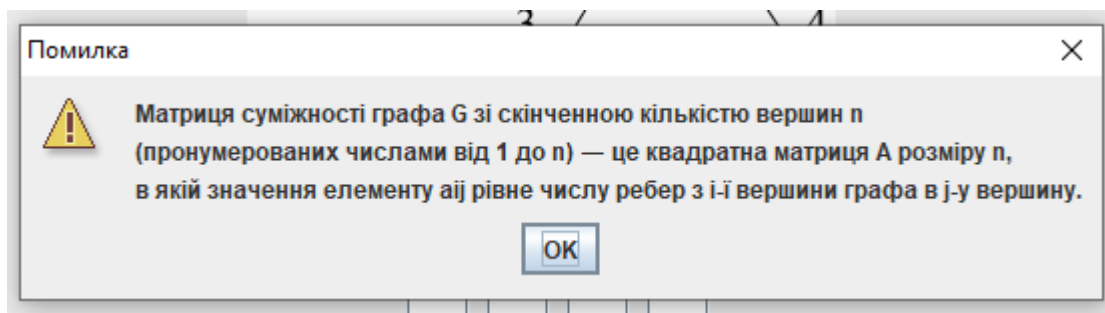


Рисунок 4.7 – Вікно-підказка

4.2 Опис програми

При запуску навчального тренажера з теми «Визначення рангу елементу системи» студенту необхідно обрати складність прикладу. Початковий екран показано на рисунку 4.8.

На рис. 4.9-4.11 показано перше завдання кожного з прикладів та умови задач.

Після введення вірних даних, програма переходить до наступних кроків, де також буде завдання на обрахунок, на рисунках 4.12-16 продемонстровано кроки навчального алгоритму для програми з теми «Визначення рангу елементу системи».

На рис. 4.17 показано останню форму, яка з'являється після завершення виконання всіх навчальних завдань студентом.

4.3 Перевірка валідності (тестування всіх процедур). Дослідження можливостей програмної реалізації

Перевірку всіх реалізованих процедур в навчальному тренажері проведено по першому прикладу (низький рівень).

На рисунках 4.18-24 продемонстровано перевірку всіх кроків навчального тренажера.

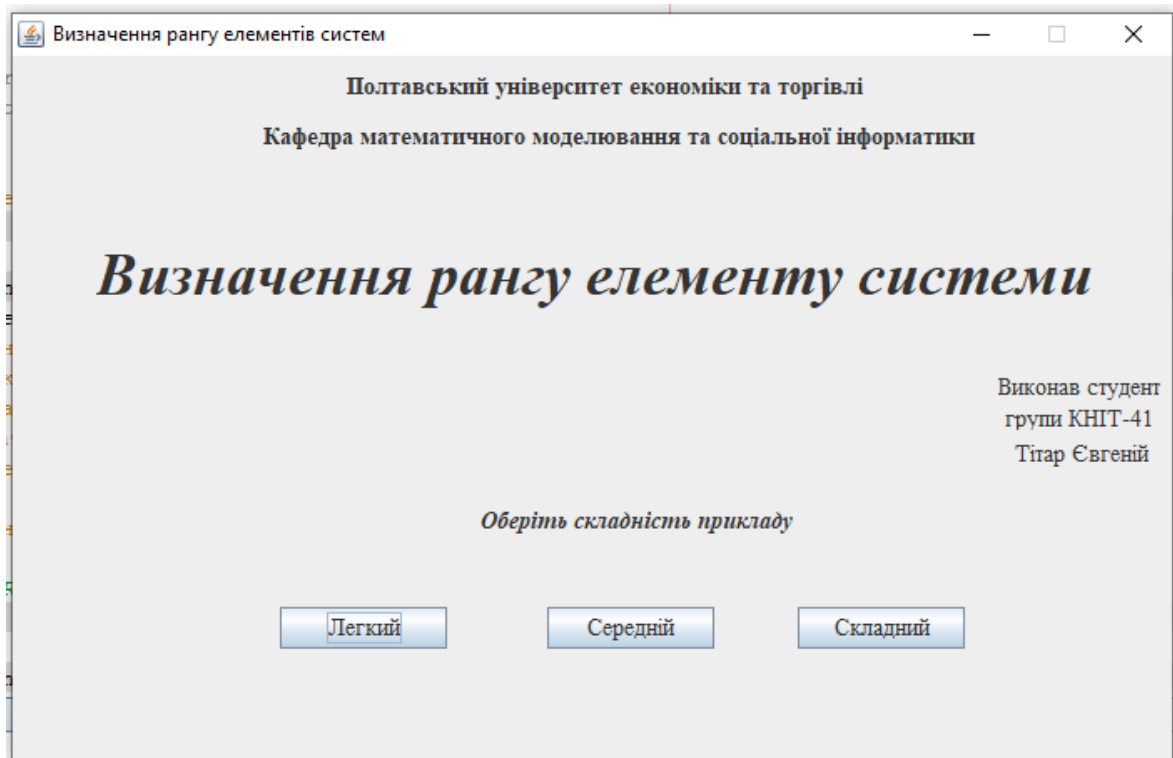


Рисунок 4.8 – Початковий екран

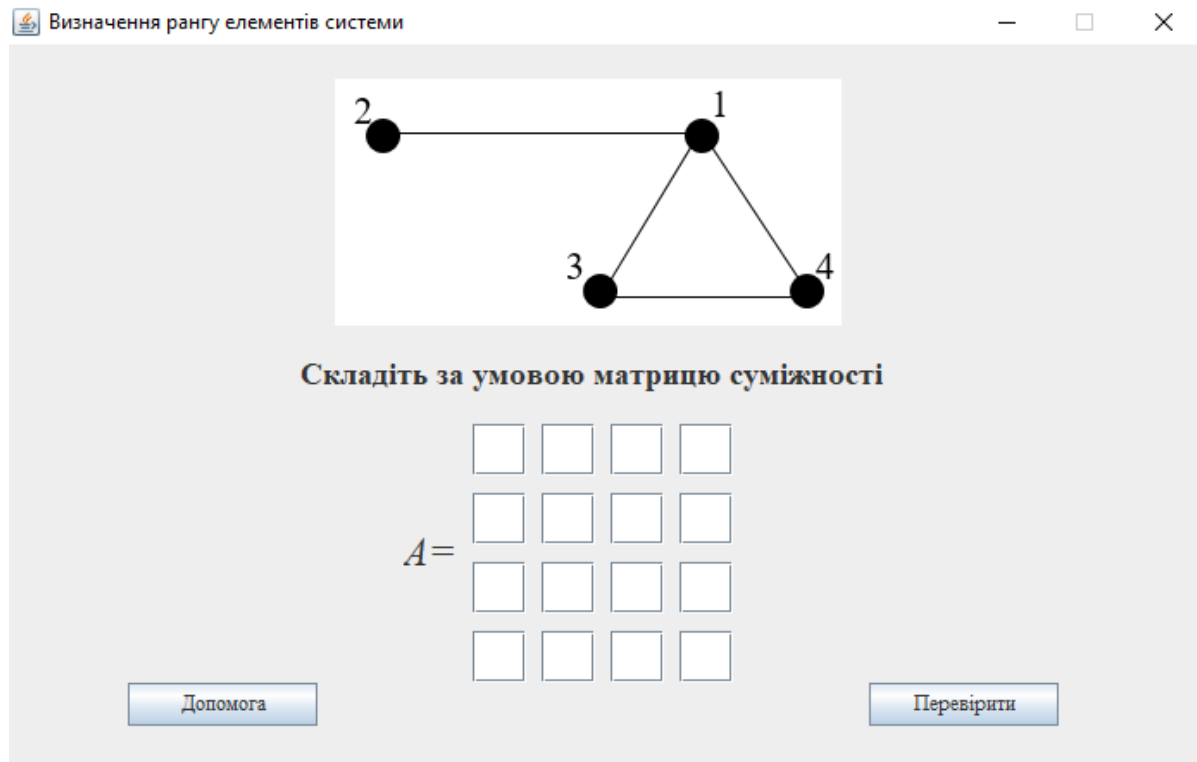


Рисунок 4.9 – Приклад низького рівня

Визначення рангу елементів системи

Складіть за умовою матрицю суміжності

$A =$

<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Допомога Перевірити

Рисунок 4.10 – Приклад середнього рівня

Визначення рангу елементів системи

Складіть за умовою матрицю суміжності

$A =$

<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Допомога Перевірити

Рисунок 4.11 – Приклад складного рівня

Визначення рангу елементів системи

$$\begin{pmatrix} 3 & 1 & 2 & 2 \\ 1 & 1 & 1 & 1 \\ 2 & 1 & 2 & 2 \\ 2 & 1 & 2 & 2 \end{pmatrix}$$

Обрахувати суму кожного рядка та суму всіх елементів матриці

$\sum_1 =$ $\sum_2 =$

$\sum_3 =$ $\sum_4 =$

$\sum =$

Допомога Перевірити

Рисунок 4.14 – Четвертий крок алгоритму

Визначення рангу елементів системи

$$\begin{array}{ll} \sum_1 = 8 & \sum_2 = 4 \\ \sum_3 = 7 & \sum_4 = 7 \\ & \sum = 26 \end{array}$$

Записати значення рангів

$P_1 =$ $P_2 =$

$P_3 =$ $P_4 =$

Допомога Перевірити

Рисунок 4.15 – П'ятий крок алгоритму

Визначення рангу елементів системи

$$\begin{matrix} P_1 = 8 & P_2 = 4 \\ P_3 = 7 & P_4 = 7 \end{matrix}$$

$$\begin{matrix} \sum_1 = 8 & \sum_2 = 4 \\ \sum_3 = 7 & \sum_4 = 7 \\ \sum = 26 \end{matrix}$$

Записати значення нормованих рангів

$\tilde{P}_1 =$

$\tilde{P}_2 =$

$\tilde{P}_3 =$

$\tilde{P}_4 =$

Допомога

Перевірити

Рисунок 4.16 – Шостий крок алгоритму

Визначення рангу елементів системи

Ви успішно розв'язали приклад!

Повернутися на початок

Вихід

Рисунок 4.17 – Остання форма

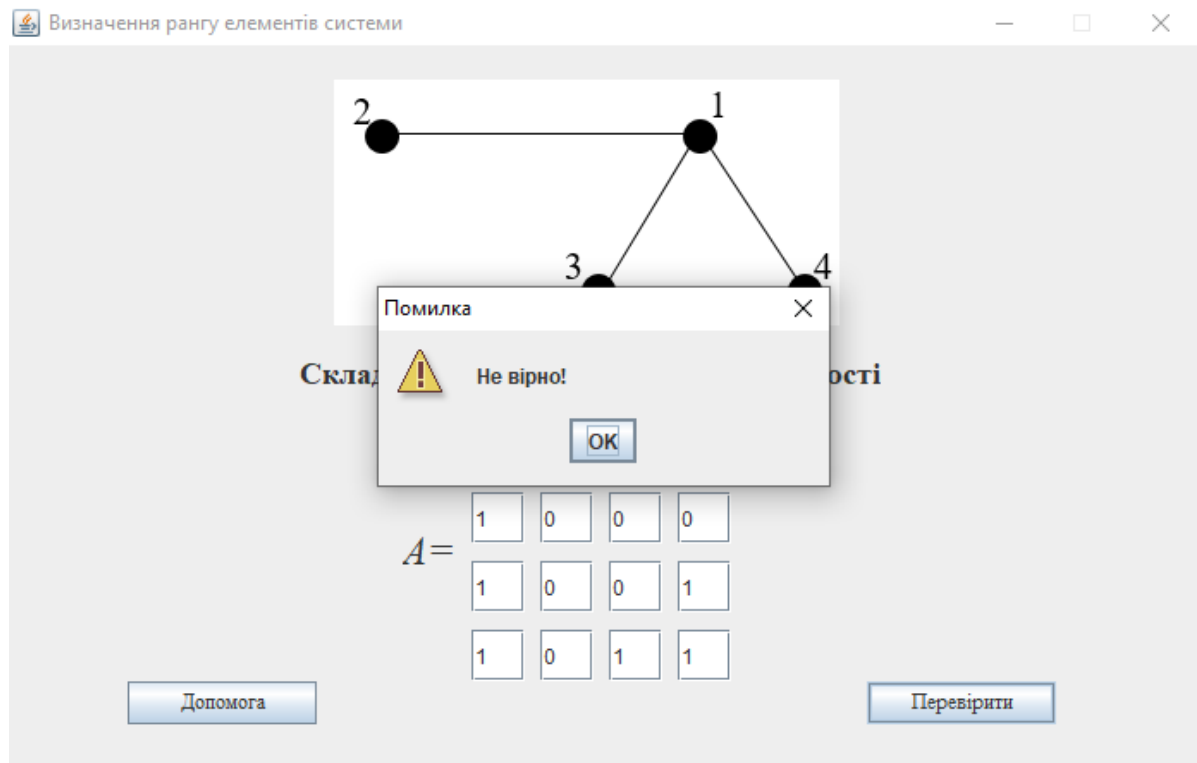


Рисунок 4. 18 – Не вірні дані першого кроку

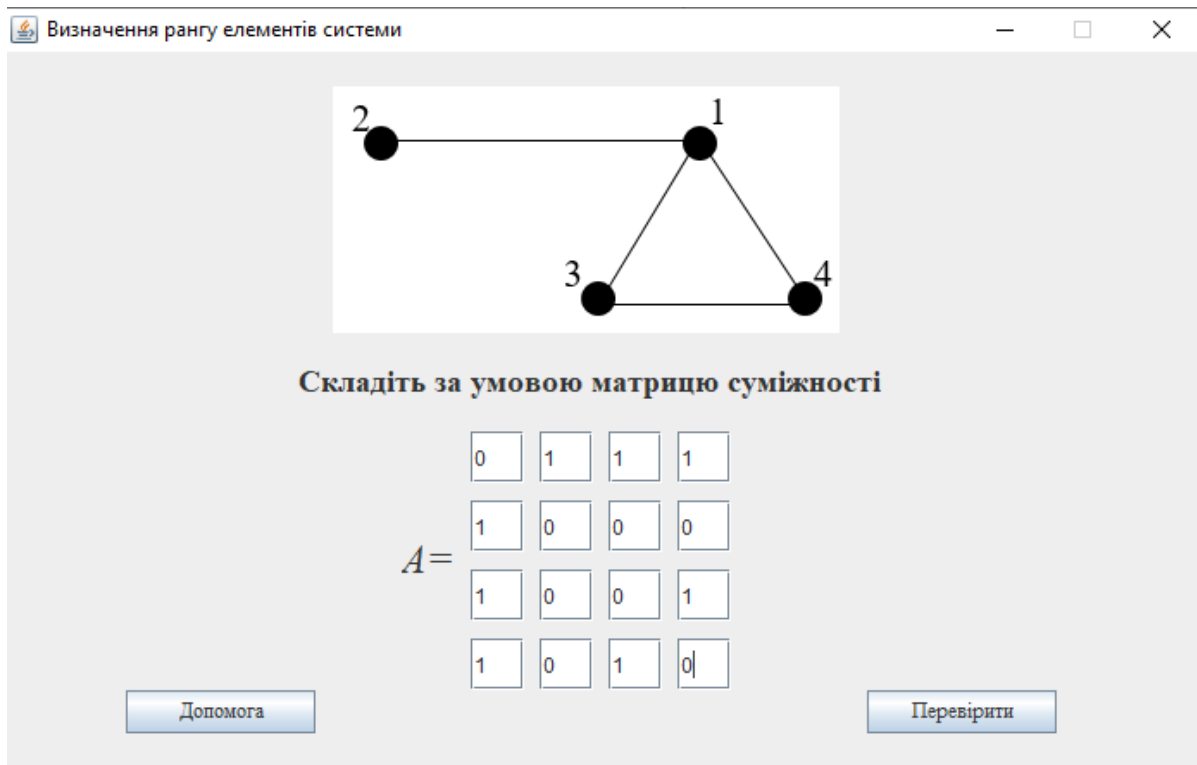


Рисунок 4. 19 – Вірні дані першого кроку

Визначення рангу елементів системи

$$\begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \times \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}$$

Обчислити кожен елемент нової матриці

$c_{11} =$
 $c_{12} =$
 $c_{13} =$
 $c_{14} =$

$c_{21} =$
 $c_{22} =$
 $c_{23} =$
 $c_{24} =$

$c_{31} =$
 $c_{32} =$
 $c_{33} =$
 $c_{34} =$

$c_{41} =$
 $c_{42} =$
 $c_{43} =$
 $c_{44} =$

Допомога Перевірити

Рисунок 4. 20 – Вірні дані другого кроку

Визначення рангу елементів системи

$$P = A + A^2 = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} + \begin{pmatrix} 3 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 2 & 1 \\ 1 & 1 & 1 & 2 \end{pmatrix}$$

Обчислити P додавши дві матриці

$P =$

<input type="text" value="3"/>	<input type="text" value="1"/>	<input type="text" value="2"/>	<input type="text" value="2"/>
<input type="text" value="1"/>	<input type="text" value="1"/>	<input type="text" value="1"/>	<input type="text" value="1"/>
<input type="text" value="2"/>	<input type="text" value="1"/>	<input type="text" value="2"/>	<input type="text" value="2"/>
<input type="text" value="2"/>	<input type="text" value="1"/>	<input type="text" value="2"/>	<input type="text" value="2"/>

Допомога Перевірити

Рисунок 4. 21 – Вірні дані третього кроку

Визначення рангу елементів системи

$$\begin{pmatrix} 3 & 1 & 2 & 2 \\ 1 & 1 & 1 & 1 \\ 2 & 1 & 2 & 2 \\ 2 & 1 & 2 & 2 \end{pmatrix}$$

Обрахувати суму кожного рядка та суму всіх елементів матриці

$\sum_1 =$ $\sum_2 =$

$\sum_3 =$ $\sum_4 =$

$\sum =$

Рисунок 4.22 – Вірні дані четвертого кроку

Визначення рангу елементів системи

$$\begin{array}{l} \sum_1 = 8 \quad \sum_2 = 4 \\ \sum_3 = 7 \quad \sum_4 = 7 \\ \sum = 26 \end{array}$$

Записати значення рангів

$P_1 =$ $P_2 =$

$P_3 =$ $P_4 =$

Рисунок 4.23 – Вірні дані п'ятого кроку

Визначення рангу елементів системи

$\begin{matrix} P_1 = 8 & P_2 = 4 \\ P_3 = 7 & P_4 = 7 \end{matrix}$	$\begin{matrix} \sum_1 = 8 & \sum_2 = 4 \\ \sum_3 = 7 & \sum_4 = 7 \\ \sum = 26 \end{matrix}$
Записати значення нормованих рангів	
$\bar{P}_1 = $ <input style="width: 50px;" type="text" value="0.3"/>	$\bar{P}_2 = $ <input style="width: 50px;" type="text" value="0.15"/>
$\bar{P}_3 = $ <input style="width: 50px;" type="text" value="0.27"/>	$\bar{P}_4 = $ <input style="width: 50px;" type="text" value="0.27"/>
<input type="button" value="Допомога"/>	<input type="button" value="Перевірити"/>

Рисунок 4.24 – Вірні дані шостого кроку

4.4 Необхідна користувачу програми інструкція

Після запуску навчального тренажеру перед користувачем з'являється вибір, а саме обрати «Низький», «Середній» або «Складний» рівень складності прикладу для визначення рангу елементу системи.

Кожен приклад дає студентові 6 завдань для обрахунку. Також кожне завдання – це завдання з веденням даних.

По завершенню проходження навчального тренажеру студент бачить останню форму, з якої він може повернутися на початковий екран або закрити навчальний тренажер.

ВИСНОВКИ

В ході виконання даної бакалаврської роботи було проаналізовано літературні джерела та програми-тренажери схожі до завдання роботи. Була виявлена повна відсутність навчальних тренажерів на дану тему.

Була досягнута поставлена мета бакалаврської роботи – створений навчальний тренажер з теми «Визначення рангу елементу системи», який буде використовуватися в навчанні студентів в дистанційному навчальному курсі «Теорія систем та математичне моделювання».

При цьому були здобуті, закріплені знання та вміння у програмуванні навчального тренажеру.

Розроблений алгоритм роботи та блок-схеми навчального тренажеру. Було виконано всі припущення щодо розробки та роботи навчального тренажеру.

Також було проведено тестування всіх процедур навчального тренажеру на виявлення помилок.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Лекція № 10-11 Якість структурної схеми та її аналіз дисципліни «Теорія систем та математичне моделювання», – 4 с.
2. Інформатика та системні науки (ІСН-2017): матеріали VIII Всеукраїнської науково-практичної конференції за міжнародною участю (м. Полтава, 16-18 березня 2017 р.) / за ред. Ємця О. О. – Полтава: ПУЕТ, 2017. – 333 с. Режим доступу – <http://dspace.puet.edu.ua/handle/123456789/5798>
3. Комп'ютерні науки і прикладна математика (КНіПМ-2018): матеріали науково-практичного семінару: Випуск 1 / за ред. Ємця О. О. – Полтава: Кафедра ММСІ ПУЕТ, 2018. – 64 с. Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/6563>
4. Комп'ютерні науки і прикладна математика (КНіПМ-2018): матеріали науково-практичного семінару: Випуск 2 / за ред. Ємця О. О. – Полтава: Кафедра ММСІ ПУЕТ, 2018. – 27 с. Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/6987>
5. Кондрашев Д. М. Розробка тренажера дистанційного навчального курсу «Математична логіка та теорія алгоритмів» з теми «Машини Т'юрінга» / Д. М. Кондрашев // Інформатика та системні науки (ІСН – 2017) : 47 матеріали VIII Всеукраїнської науково-практичної конференції за міжнародною участю (м. Полтава, 16–18 березня 2017 р.) / за ред. Ємця О. О. – Полтава : ПУЕТ, 2017. — С. 146-149. Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/5798>
6. Потерайло О. О. Програмування навчального тренажера для градієнтного методу оптимізації нелінійних функцій дистанційного навчального курсу «Методи оптимізації та дослідження операцій» / О. О. Потерайло, О. О. Ємець // Інформатика та системні науки (ІСН – 2017) : матеріали VIII Всеукраїнської науково-практичної конференції за міжнародною участю (м. Полтава, 16–18 березня 2017 р.) / за ред. Ємця О. О. – Полтава : ПУЕТ, 2017. — С. 223-225. Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/5798>

7. Русін В. С. Програмна реалізація елементу тренажеру з теми "Аналіз алгоритму сортування вставками" дисципліни "Аналіз алгоритмів" / В. С. Русін, Ю. Ф. Олексійчук // Інформатика та системні науки (ІСН-2017): матеріали VIII Всеукраїнської науково-практичної конференції за міжнародною участю (м. Полтава, 16–18 березня 2017 р.) – Полтава: ПУЕТ, 2017. – С. 236-237.

8. Сивокінь О. Ю. Розробка та програмна реалізація тренажера з теми «Метод Ленда та Дойга» дистанційного курсу «Методи оптимізації та дослідження операцій» / О. Ю. Сивокінь, О. О. Ємець // Комп'ютерні науки і прикладна математика (КНіПМ-2018): матеріали науково-практичного семінару: Випуск 2 / за ред. Ємця О. О. – Полтава: Кафедра ММСІ ПУЕТ, 2018. – С. 4-8. Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/5988>

9. Куркін В. В. Алгоритмізація та програмування елементу тренажера з побудови математичної моделі комбінаторної оптимізації «Задачі директора» / В. В. Куркін, О. О. Ємець // Комп'ютерні науки і прикладна математика (КНіПМ-2018): матеріали науково-практичного семінару: Випуск 2 / за ред. Ємця О. О. – Полтава: Кафедра ММСІ ПУЕТ, 2018. – С. 21-26. Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/6482>

10. Маркозов Д. О. Дистанційне навчання як інноваційна форма освіти [Електронний ресурс]. – Режим доступу: http://www.rusnauka.com/24_PNR_2009/Pedagogica/50819.doc.htm

11. Дистанційна освіта в країнах світу: що, де і як? [Електронний ресурс]. – Режим доступу: <https://www.chasipodii.net/mp/article/1369/>

12. Ахмад І.М. Навчання в дистанційній і змішаній формі студентів ВНЗ [Електронний ресурс]. – Режим доступу: <http://interconf.fl.kpi.ua/node/1067>

13. Толочко В. М. Проблемні аспекти дистанційної форми освіти та можливості її використання в Україні [Електронний ресурс]. – Режим доступу: http://www.provisor.com.ua/archive/2009/N11/padfo_119.php

14. Кудрявцева С.П., Колос В.В. Міжнародна інформація: Навчальний посібник – К.: Видавничий Дім «Слово», 2005. – 400с

15. Дистанційна освіта: плюси та мінуси [Електронний ресурс]. – Режим доступу: <http://www.osvita.org.ua/distance/articles/18/>
16. Моїсєєва Ю.Ю. Дистанційне навчання: інноваційна форма вищої освіти [Електронний ресурс]. – Режим доступу: <http://ea.donntu.edu.ua:8080/jspui/bitstream/123456789/7639/1/1.pdf>
17. Воронкин А.С. Предварительные итоги опроса «Дистанционное образование сегодня» [Електронний ресурс] – Режим доступу: <http://tdo.at.ua/news/do/2011-06-23-36>
18. Ємець О. О. Методичні рекомендації до виконання бакалаврської роботи для студентів за освітньою програмою «Комп'ютерні науки», спеціальності 122 «Комп'ютерні науки та інформаційні технології» / О. О. Ємець // – Полтава, Кафедра ММСІ ПУЕТ, 2017 – 71 с.