МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

МАТЕМАТИЧНИЙ ФАКУЛЬТЕТ

КАФЕДРА ПРОГРАМНОЇ ІНЖЕНЕРІЇ

Дисципліна «Автоматизоване тестування»

Лабораторна робота № 2

Виконав:

студент гр. 6.1213

Явдощен Яків Валерійович

Перевірив:

Викладач

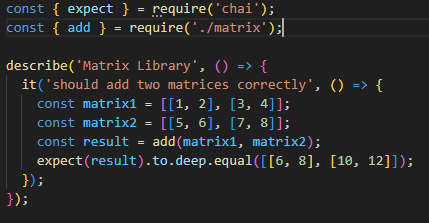
Кудін О.В.

Запоріжжя

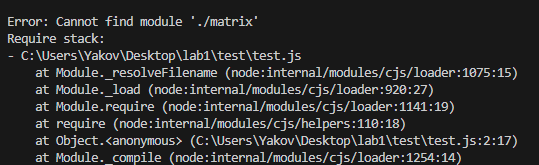
2023

Завдання для самостійної роботи

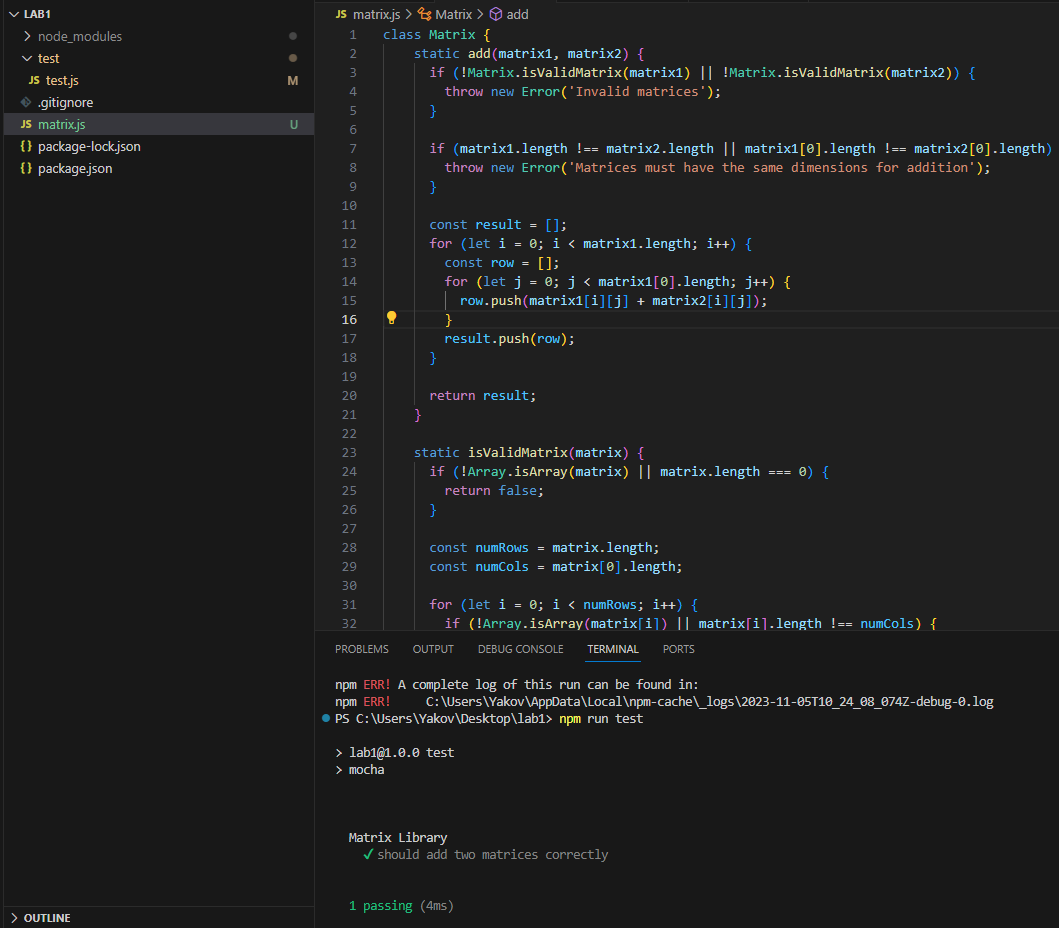
1. Реалізувати підхід TDD при розробці бібліотеки операцій над матрицями.
2. Для модульного тестування використовувати засоби Mocha, Chai.
3. Створити github репозиторій з відображенням процесу TDD.
4. Написання першого тесту



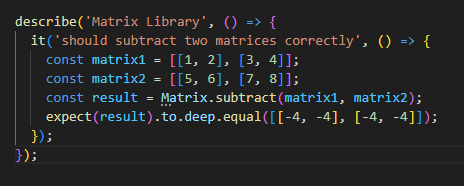
1. Запуск першого тесту

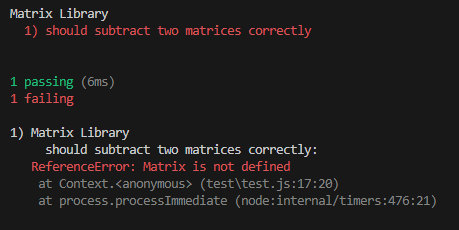


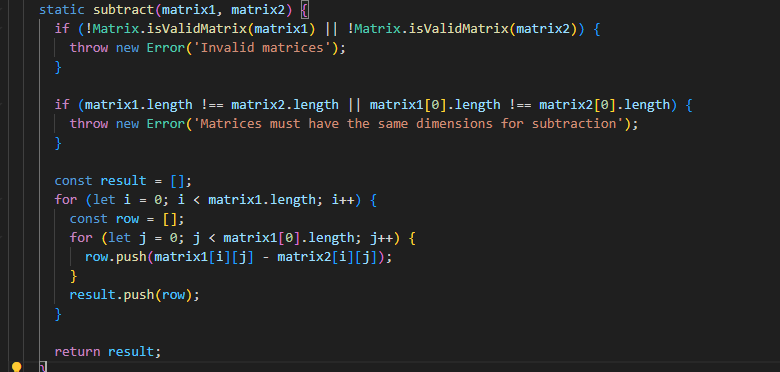
1. Реалізация коду

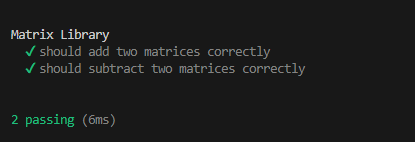


1. Додаємо наступний тест



1. Запускаємо тест 
2. Додаємо нову функціональність та запускаємо код





Контрольні запитання

1. Особливості TDD (Test-Driven Development):

a. Тестування перед розробкою: У TDD спочатку пишуться тести перед написанням фактичного коду. Тести визначають очікувану поведінку програми, яка ще не існує.

b. Ітеративний процес: Розробка в TDD відбувається ітеративно. Спочатку створюється невдалий тест, який не проходить, і лише потім розробляється код для забезпечення пройдення цього тесту. Цей цикл повторюється для кожної нової функціональності.

c. Автоматизовані тести: У TDD використовуються автоматизовані тести, які можна виконувати автоматично під час кожного циклу розробки.

d. Підтримка якості: TDD сприяє підтримці високої якості програмного забезпечення, оскільки кожна функція повинна бути покрита тестами.

e. Відсутність зайвого коду: Розробка згідно з TDD сприяє відсутності зайвого або непотрібного коду, оскільки код пишеться для вирішення конкретних завдань, які покриваються тестами.

2. Модульне тестування використовується у методології TDD на кожному етапі розробки. Перед написанням фактичного коду тестувальник визначає, як повинно поводитися програмне забезпечення і що від нього очікується. На цьому етапі створюються модульні тести, які перевіряють ці очікувані результати.

Після написання тестів, розробник приступає до написання фактичного коду, який вирішує ці вимоги та допомагає пройти тести. Код розробляється так, щоб він був покритий тестами.

Модульні тести використовуються для тестування окремих модулів або функцій програми, перевірки їх правильності та взаємодії. Це допомагає виявляти помилки на ранніх етапах розробки і забезпечує стабільність програмного забезпечення.

1. Специфікація програмного забезпечення враховується в TDD на початковому етапі. В TDD, специфікація програмного забезпечення називається "тестами" або "тестовими кейсами". Ці тести визначають очікувану поведінку програми та конкретні умови, за яких програмне забезпечення має працювати.

Тести визначаються перед розробкою фактичного коду і служать як основа для розробки програми. Таким чином, специфікація програмного забезпечення враховується з самого початку розробки і визначає очікуваний результат для кожної функціональності чи модуля програми.

На кожному етапі розробки розробник створює фактичний код таким чином, щоб він задовольняв тестові кейси, визначені в специфікації. Таким чином, специфікація і тести відображають очікувану поведінку програмного забезпечення і перевіряють її на кожному етапі розробки.