

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

**Український державний університет науки і технологій**

Кафедра «Комп’ютерні інформаційні технології»

**Лабораторна робота №2  
з дисципліни «Якість програмного забезпечення та тестування»**

**на тему:**

**«**Тестування методами чорної скриньки.»

Виконав:

студент гр. ПЗ2011

Проценко Р.О.

Прийняла:

Стаднік А. В.

Дніпро, 2023

**Тема.** Тестування методами білої скриньки.

**Мета.** Отримати практичні навички тестування методами білої скриньки.

**Завдання**

Підготувати дві чи більше функцій для тестування. Функції повинні містити не менше шести керуючих конструкцій (розгалужень і циклів). Зазначені структури мають бути вкладеними. Виконати тестування функцій методами білої скриньки.

**Специфікація функції**

Функція здійснює множення двох матриць між собою.

Функція приймає на вхід у явному виді:

* a – двовимірний масив чисел з плаваючою комою.
* b - двовимірний масив чисел з плаваючою комою.

У не явному вигляді функція повертає виключення пов'язані з невідповідністю розмірів матриць.

У явному повертає двовимірний масив чисел з плаваючою комою, результат множення двох матриць.

**Класи еквівалентності**

|  |  |  |
| --- | --- | --- |
| Вхідні умови | Правильні класи еквівалентності | Неправильні класи еквівалентності |
| Значення та розмірність першого двовимірного масиву | Масив представляє скаляр(1) |  |
| Масив представляє вектор(2) |
| Масив представляє матрицю(3) |
| Значення та розмірність першого двовимірного масиву | Масив представляє скаляр(4) |  |
| Масив представляє вектор(5) |
| Масив представляє матрицю(6) |
| Відповідність розмірності двовимірних масивів | Масиви представляють матриці або вектори та кількість стовпців першого рівна кількості рядків другого(7) | Масиви представляють матриці або вектори та кількість стовпців першого не рівна кількості рядків другого (8) |
| Масив або масиви представляють не вектори або матриці(9) |

**Тести за методом еквівалентних розбиттів**

|  |  |  |  |
| --- | --- | --- | --- |
| ***Класи що покриваються*** | ***Тест*** | ***Вхід*** | ***Вихід*** |
| 3, 6, 7 | 1 | а = 1, 2 , 3  4, 5, 6  b = 7, 8  9, 10  11, 12 | Матриця вигляду:  58, 64  139, 154 |
| 3,6,8 | 2 | а = 1, 2 , 3  4, 5, 6  b = 7, 8  9, 10 | IllegalArgumentException  ("Matrix dimensions are not compatible for multiplication.") |
| 1, 5, 9 | 3 | а = 1  b = 7,  8 | IllegalArgumentException  ("The function received a scalar instead of a matrix") |
| 2, 4, 9 | 4 | а = 1, 2, 3, 4, 5, 6  b = 7 | IllegalArgumentException  ("The function received a scalar instead of a matrix") |

**Тести за методом припущення про помилку**Виконаємо тестування методом припущення про помилку. Висунемо ряд припущень:

1. Якщо в нас один з параметрів є скаляром, а інший вектором, чи відбудеться множення?
2. Якщо в нас один з параметрів є скаляром, а інший матрицею, чи відбудеться множення?
3. Якщо в нас один з параметрів є вектором, а інший матрицею, чи відбудеться множення?
4. Якщо в нас обидва параметри є скалярами, чи відбудеться множення?
5. Якщо в нас обидва параметри є векторами відповідних розмірів, чи відбудеться множення?
6. Якщо в нас обидва параметри є векторами не відповідних розмірів, чи відбудеться множення?
7. Якщо в нас обидва параметри є матрицями з правильними розмірами, чи відбудеться множення?
8. Якщо в нас обидва параметри є матрицями з невідповідними розмірами, чи відбудеться множення?

|  |  |  |  |
| --- | --- | --- | --- |
| ***Припущення*** | ***Тест*** | ***Вхід*** | ***Вихід*** |
| 1 | 1 | а = 1  b = 7,  8 | IllegalArgumentException  ("The function received a scalar instead of a matrix") |
| 2 | 2 | а = 1  b = 7, 8  8, 9 | IllegalArgumentException  ("The function received a scalar instead of a matrix") |
| 3 | 3 | а = 1, 2  b = 7, 8  8, 9 | Вектор вигляду:  23  26 |
| 4 | 4 | а = 1  b = 7 | IllegalArgumentException  ("The function received a scalar instead of a matrix") |
| 5 | 5 | а = 1, 2  b = 7  8 | Скаляр:  23 |
| 6 | 6 | а = 1, 2  b = 7,8 | IllegalArgumentException  ("Matrix dimensions are not compatible for multiplication.") |
| 7 | 7 | а = 1, 2 , 3  4, 5, 6  b = 7, 8  9, 10  11, 12 | Матриця вигляду:  58, 64  139, 154 |
| 8 | 8 | а = 1, 2 , 3  4, 5, 6  b = 7, 8  9, 10 | IllegalArgumentException  ("Matrix dimensions are not compatible for multiplication.") |

**Тести за методом причин та наслідків**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Причини та наслідки | Умовне позначення | Значення | Помітки про присутність | | | |
| 1 | 2 | 3 | 4 |
| Причини | П1 | Перший масив представляє скаляр |  |  | **+** |  |
| П2 | Перший масив представляє вектор |  |  |  | **+** |
| П3 | Перший масив представляє матрицю | **+** | **+** |  |  |
| П4 | Масив представляє скаляр |  |  |  | **+** |
| П5 | Масив представляє вектор |  |  | **+** |  |
| П6 | Масив представляє матрицю | **+** | **+** |  |  |
| П7 | Масиви представляють матриці або вектори та кількість стовпців першого рівна кількості рядків другого | **+** |  |  |  |
| П8 | Масиви представляють матриці або вектори та кількість стовпців першого **не** рівна кількості рядків другого |  | **+** |  |  |
| П9 | Масив або масиви представляють не вектори або матриці |  |  | **+** | **+** |
| Наслідки | н1 | Результат представлений матрицею | **+** |  |  |  |
| н2 | IllegalArgumentException  ("Matrix dimensions are not compatible for multiplication.") |  | **+** |  |  |
| н3 | IllegalArgumentException  ("The function received a scalar instead of a matrix") |  |  | **+** | **+** |

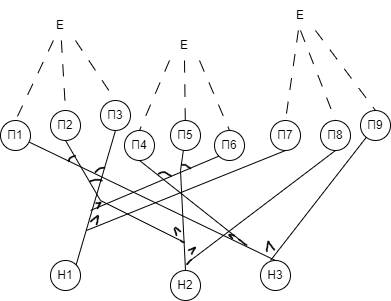


рис. 1, функціональна діаграма

|  |  |  |  |
| --- | --- | --- | --- |
| ***Помітка*** | ***Тест*** | ***Вхід*** | ***Вихід*** |
| 1 | 1 | а = 1, 2 , 3  4, 5, 6  b = 7, 8  9, 10  11, 12 | Матриця вигляду:  58, 64  139, 154 |
| 2 | 2 | а = 1, 2 , 3  4, 5, 6  b = 7, 8  9, 10 | IllegalArgumentException  ("Matrix dimensions are not compatible for multiplication.") |
| 3 | 3 | а = 1  b = 7,  8 | IllegalArgumentException  ("The function received a scalar instead of a matrix") |
| 4 | 4 | а = 1, 2, 3, 4, 5, 6  b = 7 | IllegalArgumentException  ("The function received a scalar instead of a matrix") |

**Тестування методом граничних умов**

Метод граничних умов для даного прикладу не є ефективним з огляду на те, що обмеження накладаються лише формою матриць та відповідності їх розмірностей.

**Специфікація функції №2**

Функція здійснює поелементне додавання двох матриць.

Функція приймає на вхід у явному виді:

* a – двовимірний масив чисел з плаваючою комою.
* b - двовимірний масив чисел з плаваючою комою.

У не явному вигляді функція повертає виключення пов'язані з невідповідністю розмірів матриць.

У явному повертає двовимірний масив чисел з плаваючою комою, результат додавання двох матриць.

**Класи еквівалентності**

|  |  |  |
| --- | --- | --- |
| Вхідні умови | Правильні класи еквівалентності | Неправильні класи еквівалентності |
| Значення та розмірність першого двовимірного масиву | Масив представляє скаляр(1) |  |
| Масив представляє вектор(2) |
| Масив представляє матрицю(3) |
| Значення та розмірність першого двовимірного масиву | Масив представляє скаляр(4) |  |
| Масив представляє вектор(5) |
| Масив представляє матрицю(6) |
| Відповідність розмірності двовимірних масивів | Масиви мають одникову розмірність (7) | Масиви мають різну розмірність (8) |

**Тести за методом еквівалентних розбиттів**

|  |  |  |  |
| --- | --- | --- | --- |
| ***Класи що покриваються*** | ***Тест*** | ***Вхід*** | ***Вихід*** |
| 3, 6, 7 | 1 | а = 1, 2  4, 5  b = 7, 8  9, 10 | Матриця вигляду:  8, 10  13, 15 |
| 2,4,7 | 2 | а = 1, 2  b = 7, 8 | Вектор вигляду:  8, 10 |
| 1, 5, 8 | 3 | а = 1  b = 7,  8 | IllegalArgumentException  ("Matrix dimensions must be same.") |
| 2, 4, 8 | 4 | а = 1, 2, 3, 4, 5, 6  b = 7 | IllegalArgumentException("Matrix dimensions must be same.") |

**Тести за методом припущення про помилку**Виконаємо тестування методом припущення про помилку. Висунемо ряд припущень:

1. Якщо в нас один з параметрів є скаляром, а інший вектором, чи відбудеться додавання?
2. Якщо в нас один з параметрів є скаляром, а інший матрицею, чи відбудеться додавання?
3. Якщо в нас один з параметрів є вектором, а інший матрицею, чи відбудеться додавання?
4. Якщо в нас обидва параметри є скалярами, чи відбудеться додавання?
5. Якщо в нас обидва параметри є векторами відповідних розмірів, чи відбудеться додаванн?
6. Якщо в нас обидва параметри є векторами не відповідних розмірів, чи відбудеться додавання?
7. Якщо в нас обидва параметри є матрицями з правильними розмірами, чи відбудеться додавання?
8. Якщо в нас обидва параметри є матрицями з невідповідними розмірами, чи відбудеться додавання?

|  |  |  |  |
| --- | --- | --- | --- |
| ***Припущення*** | ***Тест*** | ***Вхід*** | ***Вихід*** |
| 1 | 1 | а = 1  b = 7,  8 | IllegalArgumentException  ("Matrix dimensions must be same.") |
| 2 | 2 | а = 1  b = 7, 8  8, 9 | IllegalArgumentException  ("Matrix dimensions must be same.") |
| 3 | 3 | а = 1, 2  b = 7, 8  8, 9 | IllegalArgumentException  ("Matrix dimensions must be same.") |
| 4 | 4 | а = 1  b = 7 | Скаляр: 8 |
| 5 | 5 | а = 1, 2  b = 7, 8 | Вектор вигляду:  8, 10 |
| 6 | 6 | а = 1, 2  b = 7,8, 3 | IllegalArgumentException  ("Matrix dimensions must be same.") |
| 7 | 7 | а = 1, 2  4, 5  b = 7, 8  9, 10 | Матриця вигляду:  8, 10  13, 15 |
| 8 | 8 | а = 1, 2 , 3  4, 5, 6  b = 7, 8  9, 10 | IllegalArgumentException  ("Matrix dimensions must be same.") |

**Тести за методом причин та наслідків**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Причини та наслідки | Умовне позначення | Значення | Помітки про присутність | | | |
| 1 | 2 | 3 | 4 |
| Причини | П1 | Перший масив представляє скаляр |  |  | **+** |  |
| П2 | Перший масив представляє вектор |  | **+** |  | **+** |
| П3 | Перший масив представляє матрицю | **+** |  |  |  |
| П4 | Масив представляє скаляр |  |  |  | **+** |
| П5 | Масив представляє вектор |  | **+** | **+** |  |
| П6 | Масив представляє матрицю | **+** |  |  |  |
| П7 | Масиви мають одникову розмірність | **+** | **+** |  |  |
| П8 | Масиви мають різну розмірність |  |  | **+** | **+** |
| Наслідки | н1 | Результат представлений масивом | **+** | **+** |  |  |
| н2 | IllegalArgumentException  ("Matrix dimensions must be same.") |  |  | **+** | **+** |

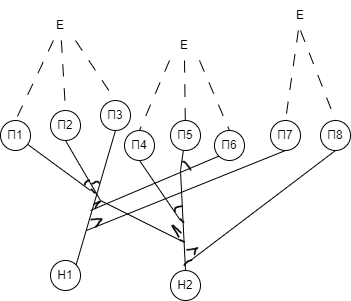


рис. 2, функціональна діаграма

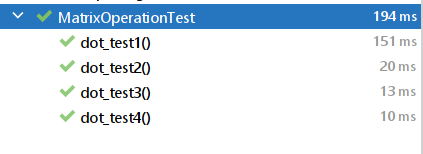
|  |  |  |  |
| --- | --- | --- | --- |
| ***Помітка*** | ***Тест*** | ***Вхід*** | ***Вихід*** |
| 1 | 1 | а = 1, 2  4, 5  b = 7, 8  9, 10 | Матриця вигляду:  8, 10  13, 15 |
| 2 | 2 | а = 1, 2  b = 7, 8 | Вектор вигляду:  8, 10 |
| 3 | 3 | а = 1  b = 7,  8 | IllegalArgumentException  ("Matrix dimensions must be same.") |
| 4 | 4 | а = 1, 2, 3, 4, 5, 6  b = 7 | IllegalArgumentException("Matrix dimensions must be same.") |

**Тестування методом граничних умов**

Метод граничних умов для даного прикладу не є ефективним з огляду на те, що обмеження накладаються лише формою матриць та відповідності їх розмірностей.

**Unit-тестування функції №1**

@org.junit.jupiter.api.Test  
void dot\_test1() {  
 double[][] matrix1 = {{1, 2, 3}, {4, 5, 6}};  
 double[][] matrix2 = {{7, 8}, {9, 10}, {11, 12}};  
 double[][] expected = {{58, 64}, {139, 154}};  
 double[][] result = *dot*(matrix1, matrix2);  
  
 Assertions.*assertArrayEquals*(expected, result);  
}  
  
@org.junit.jupiter.api.Test  
void dot\_test2() {  
 double[][] matrixA = {{1, 2, 3}, {4, 5, 6}};  
 double[][] matrixB = {{7, 8}, {9, 10}};  
 Throwable throwable = *assertThrows*(Throwable.class, () -> {  
 *dot*(matrixA, matrixB);  
 });  
  
 *assertEquals*(IllegalArgumentException.class, throwable.getClass());  
}  
  
@org.junit.jupiter.api.Test  
void dot\_test3() {  
 double[][] matrixA = {{1}};  
 double[][] matrixB = {{7, 8}};  
 Throwable throwable = *assertThrows*(Throwable.class, () -> {  
 *dot*(matrixA, matrixB);  
 });  
  
 *assertEquals*(IllegalArgumentException.class, throwable.getClass());  
}  
  
@org.junit.jupiter.api.Test  
void dot\_test4() {  
 double[][] matrixA = {{1, 2, 3, 4, 5, 6}};  
 double[][] matrixB = {{7}};  
 Throwable throwable = *assertThrows*(Throwable.class, () -> {  
 *dot*(matrixA, matrixB);  
 });  
  
 *assertEquals*(IllegalArgumentException.class, throwable.getClass());  
}

****

**Unit-тестування функції №2**

@org.junit.jupiter.api.Test  
void add\_test1() {  
 double[][] matrix1 = {{1, 2}, {4, 5}};  
 double[][] matrix2 = {{7, 8}, {9, 10}};  
 double[][] expected = {{8, 10}, {13, 15}};  
 double[][] result = *add*(matrix1, matrix2);  
  
 Assertions.*assertArrayEquals*(expected, result);  
}  
  
@org.junit.jupiter.api.Test  
void add\_test2() {  
 double[][] matrixA = {{1, 2}};  
 double[][] matrixB = {{7, 8}};  
 Throwable throwable = *assertThrows*(Throwable.class, () -> {  
 *add*(matrixA, matrixB);  
 });  
  
 *assertEquals*(IllegalArgumentException.class, throwable.getClass());  
}  
  
@org.junit.jupiter.api.Test  
void add\_test3() {  
 double[][] matrixA = {{1}};  
 double[][] matrixB = {{7, 8}};  
 Throwable throwable = *assertThrows*(Throwable.class, () -> {  
 *add*(matrixA, matrixB);  
 });  
  
 *assertEquals*(IllegalArgumentException.class, throwable.getClass());  
}  
  
@org.junit.jupiter.api.Test  
void add\_test4() {  
 double[][] matrixA = {{1, 2, 3, 4, 5, 6}};  
 double[][] matrixB = {{7}};  
 Throwable throwable = *assertThrows*(Throwable.class, () -> {  
 *add*(matrixA, matrixB);  
 });  
  
 *assertEquals*(IllegalArgumentException.class, throwable.getClass());  
}

****

**Висновки**

Тестування методами білої скриньки - це важлива частина процесу розробки програмного забезпечення. Воно дозволяє перевірити внутрішню структуру програми, включаючи модулі, функції та гілки коду. Головною метою цього виду тестування є забезпечення відповідності програми вимогам та забезпечення її стабільної роботи.

Тестування білої скриньки проводиться паралельно з іншими методами тестування для забезпечення повноти оцінки програмного продукту. Під час цього тестування, тестувальники мають доступ до внутрішньої структури програми, що дозволяє створювати тестові випадки на основі коду.

Під час тестування методами білої скриньки можуть бути виявлені різні дефекти, такі як логічні помилки та проблеми з безпекою. В результаті, це сприяє покращенню якості та надійності програмного продукту, що є важливим завданням в розробці програмного забезпечення.