**Sprawozdanie z przedmiotu Inżynieria Oprogramowania**

**Laboratorium nr. 2 – Rozwój podyktowany testami**

Katowice, 22.03.2023r

Natalia Omiotek

Na początku stworzono pomocniczą klasę Lab\_2\_Param która będzie przekazywała do testów paramerycznych 3 wartości int.

class Lab\_2\_Param : public ::testing::TestWithParam<std::tuple<int, int, int>>

{

};

W ramach zajęć napisano funkcje testujące dla poszczególnych metod w klasie:

- Konstruktora parametrycznego

- Konstruktora kopiującego

- Konstruktora parametrycznego biorącego za parametr listę inicjalizująca

- Operator porównania

- Operator przypisania

- Operator dodawania

- Operator odejmowania

- Operator mnożenia

- Funkcji Det()

Następujące metody okazały się błędne:

- Konstruktor parametryczny

TEST(Lab2, Konstruktor\_Parametryczny) {

Matrix A(2, 3, 1.0);

EXPECT\_EQ(A.Rows(), 2);

EXPECT\_EQ(A.Columns(), 3);

EXPECT\_DOUBLE\_EQ(A(0, 0), 1.0);

EXPECT\_DOUBLE\_EQ(A(1, 2), 1.0);

}

Komunikat:

#1 - Expected equality of these values:

A(0, 0)

Which is: 0

1.0

Which is: 1

#2 - Expected equality of these values:

A(1, 2)

Which is: 0

1.0

Which is: 1

- Operator odejmowania

TEST(Lab2, Odejmowanie) {

Matrix A = { {1.0, 2.0}, {3.0, 4.0} };

Matrix B = { {1.0, 1.0}, {1.0, 1.0} };

Matrix wynik = A - B;

EXPECT\_EQ(wynik.Rows(), 2);

EXPECT\_EQ(wynik.Columns(), 2);

EXPECT\_DOUBLE\_EQ(wynik(0, 0), 0.0);

EXPECT\_DOUBLE\_EQ(wynik(1, 1), 3.0);

}

Komunikat:

#1 - Expected equality of these values:

wynik(0, 0)

Which is: 2

0.0

Which is: 0

#2 - Expected equality of these values:

wynik(1, 1)

Which is: 5

3.0

Which is: 3

- Mnożenie (podczas mnożenia macierzy stworzonych wadliwym konstruktorem parametrycznym)

TEST(Lab2, Mnozenie) {

Matrix A(1, 1, 3.0);

Matrix B(1, 1, 7.0);

Matrix wynik = A \* B;

Matrix C(A \* B);

EXPECT\_DOUBLE\_EQ(C(0, 0), (3.0 \* 7.0));

EXPECT\_DOUBLE\_EQ(wynik(0, 0), (3.0 \* 7.0));

//mnozenie nie dziala poniewaz konstruktor parametryczny nie dziala!

Matrix D = { {1.0, 2.0}, {3.0, 4.0} };

Matrix E = { {1.0, 1.0}, {1.0, 1.0} };

Matrix wynik1 = D \* E;

EXPECT\_DOUBLE\_EQ(wynik1(0, 0), 3);

EXPECT\_DOUBLE\_EQ(wynik1(1, 1), 7);

}

Komunikat:

#1 - Expected equality of these values:

C(0, 0)

Which is: 0

(3.0 \* 7.0)

Which is: 21

#2 - Expected equality of these values:

wynik(0, 0)

Which is: 0

(3.0 \* 7.0)

Which is: 21

Wnioski: