



AKADEMIA GÓRNICZO-HUTNICZA

Dokumentacja do projektu

Biblioteka do obsługi macierzy

z przedmiotu

Języki Programowania Obiektowego

Elektronika i Telekomunikacja, Rok III

Marcin Dziedzic

Grupa IV, piątek 11:30

Prowadzący: Jakub Zimnol

9.01.2025

Biblioteka Matrix umożliwia tworzenie i manipulowanie macierzami w języku C++. Zawiera klasy do obsługi macierzy ogólnych, macierzy kwadratowych oraz macierzy jednostkowych, wraz z podstawowymi operacjami arytmetycznymi i funkcjami pomocniczymi.

Klasy

1. Klasa Matrix

Opis:

Podstawowa klasa reprezentująca macierz o dowolnym rozmiarze. Umożliwia wykonywanie podstawowych operacji arytmetycznych oraz manipulację strukturą macierzy (dodawanie/usuwanie wierszy i kolumn).

Konstruktory:

1) Matrix(size_t rows, size_t cols)

Tworzy macierz o podanej liczbie wierszy i kolumn, inicjalizując wszystkie elementy zerem.

2) Matrix(const std::vector<std::vector<double>> &values)

Tworzy macierz na podstawie podanego wektora wektorów.

Operatory:

1) double &operator()(size_t i, size_t j)

Umożliwia modyfikację elementu macierzy znajdującego się w wierszu „i” i kolumnie „j”.

2) const double operator()(size_t i, size_t j) const

Umożliwia odczytanie wartości elementu macierzy znajdującego się w wierszu „i” i kolumnie „j”.

3) Matrix operator+(const Matrix &other) const

Zwraca nową macierz będącą sumą bieżącej macierzy i macierzy „other”. Macierze muszą mieć takie same wymiary.

4) *Matrix operator-(const Matrix &other) const*

Zwraca nową macierz będącą różnicą bieżącej macierzy i macierzy „other”. Macierze muszą mieć takie same wymiary.

5) *Matrix operator*(const Matrix &other) const*

Zwraca nową macierz będącą iloczynem bieżącej macierzy i macierzy „other”. Liczba kolumn bieżącej macierzy musi być równa liczbie wierszy macierzy „other”.

6) *Matrix operator*(double scalar) const*

Zwraca nową macierz będącą wynikiem mnożenia bieżącej macierzy przez skalar.

7) *friend Matrix operator*(double scalar, const Matrix &matrix)*

Pozwala na przemienność mnożenia przez skalar (*skalar * macierz lub macierz * skalar*(w tym wypadku dodaje możliwość mnożenia „*skalar * macierz*”).

8) *Matrix operator/(const Matrix &other) const*

Zwraca nową macierz będącą wynikiem podzielenia bieżącej macierzy przez macierz „other”, poprzez mnożenie przez odwrotność.

Metody:

1) *transpose()*

Matrix transpose() const

Zwraca transponowaną macierz, zamieniając wiersze z kolumnami.

2) *determinant()*

double determinant() const

Oblicza i zwraca wyznacznik macierzy. Macierz musi być kwadratowa.

3) *inverse()*

Matrix inverse() const

Zwraca macierz odwrotną do bieżącej macierzy. Macierz musi być kwadratowa i wyznacznik musi być różny od 0.

4) `addRow()`

void addRow(const std::vector<double> &newRow, bool atBottom = true)

Dodaje nowy wiersz „newRow” do macierzy. Jeśli „atBottom” jest „true”, dodaje na końcu; w przeciwnym razie na początku.

5) `addColumn()`

void addColumn(const std::vector<double> &newColumn, bool atRight = true)

Dodaje nową kolumnę „newColumn” do macierzy. Jeśli „atRight” jest „true”, dodaje na końcu; w przeciwnym razie na początku.

6) `removeRow()`

void removeRow(bool fromBottom = true)

Usuwa wiersz z macierzy. Jeśli „fromBottom” jest „true”, usuwa ostatni wiersz; w przeciwnym razie pierwszy.

7) `removeColumn()`

void removeColumn(bool fromRight = true)

Usuwa kolumnę z macierzy. Jeśli „fromRight” jest „true”, usuwa ostatnią kolumnę; w przeciwnym razie pierwszą.

8) `print()`

void print() const

Wyświetla zawartość macierzy w zdefiniowanych odstępach.

2. Klasa `SquareMatrix` (dziedziczy po `Matrix`)

Opis:

Klasa reprezentująca macierz kwadratową, czyli macierz o tej samej liczbie wierszy i kolumn. Dodatkowo umożliwia przypisanie jednej wartości do wszystkich elementów macierzy.

Konstruktory:

1) `SquareMatrix(size_t size)`

Tworzy macierz kwadratową o podanym rozmiarze (rozmiar jest taki sam dla wierszy i kolumn, ponieważ jest to macierz kwadratowa).

Operatory:

1) `SquareMatrix &operator=(double value)`

Przypisuje wartość „value” do wszystkich elementów macierzy.

3. Klasa `SquareMatrixIdentity` (dziedziczy po `SquareMatrix`)

Opis:

Klasa reprezentująca macierz jednostkową, czyli macierz kwadratową z jedynkami na przekątnej głównej i zerami w pozostałych miejscach.

Konstruktory:

1) `SquareMatrixIdentity(size_t size)`

Tworzy macierz jednostkową kwadratową o podanym rozmiarze.