

KURS JĘZYKA C++

MACIERZE

Instytut Informatyki Uniwersytetu Wrocławskiego

Paweł Rzechonek

Macierz w matematyce to układ liczb, symboli lub wyrażeń zapisanych w postaci tablicy. W każdej macierzy można wyróżnić: wiersze (poziomy układ elementów znajdujących się w jednej linii) i kolumny (pionowy układ elementów znajdujących się w jednej linii). Dane w macierzy nazywają się elementami, współczynnikami lub wyrazami, a każdy element można jednoznacznie zdefiniować przez podanie jego indeksu (zwykle w tej kolejności: numer wiersza i kolumny macierzy, w której stoi). Na macierzach można wykonywać podstawowe działania jak: dodawanie, odejmowanie i mnożenie macierzy, mnożenie macierzy przez skalar.

Macierz A o wymiarze $m \times n$ zawiera elementy $a_{i,j}$, które indeksujemy uporządkowaną parą liczb (i, j) , gdzie $i = 1, 2, \dots, m$ oraz $j = 1, 2, \dots, n$. Macierz A zapisujemy w postaci prostokątnej tablicy mającej m wierszy i n kolumn:

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \dots & a_{m,n} \end{bmatrix} = [a_{i,j}]_{m \times n}$$

Jeżeli macierz ma tyle samo wierszy co kolumn czyli $m = n$, to nazywa się ją macierzą kwadratową. Na takim typie macierzy wykonywane jest najwięcej działań, jak obliczanie wyznacznika macierzy czy macierzy odwrotnej.

Zadanie.

Zdefiniuj klasę `macierz` reprezentującą macierz liczb rzeczywistych. Macierz w obiekcie tej klasy powinna być prostokątną tablicą (tablica wskaźników do tablic jednowymiarowych) utworzoną dynamicznie na stacku. Klasa ta powinna być wyposażona w konstruktor tworzący macierz kwadratową jednostkową (macierz diagonalna z jedynkami na przekątnej), konstruktor tworzący macierz prostokątną wypełnioną zerami, konstruktor kopiujący, konstruktor przenoszący, przypisanie kopiujące, przypisanie przenoszące oraz destruktory (usunięcie wszystkich tablic ze sterty).

W klasie `macierz` zdefiniuj operatory umożliwiające wykonywanie podstawowych operacji na macierzach — mnożenie macierzy przez skalar, dodawanie, odejmowanie i mnożenie macierzy oraz transpozycja. Pamiętaj, że klasa reprezentująca macierz może być dość duża i nie powinno się (o ile to nie jest konieczne) przekazywać jej przez wartość za pomocą a raczej przez referencję. Można więc zdefiniować po dwie wersje każdego operatora macierzowego: jeden jako funkcja zaprzyjaźniona a drugi jako składowy operator przypisania. Nie zapomnij przy każdej metodzie zadeklarować jakie wyjątki one zgłaszają.

```

class Macierz {
// ...
public:
    friend Macierz operator+ (const Macierz &x, const Macierz &y)
        throw (rozmiary_nie_pasuja);
    Macierz & operator+= (const Macierz &y)
        throw (rozmiary_nie_pasuja);
// ...
};

```

Elementarnymi przekształceniami macierzy są: 1) przestawienie dwóch wierszy, 2) pomnożenie dowolnego wiersza przez liczbę różną od zera, 3) dodanie do wiersza krotności innego wiersza. Elementarnym przekształceniem drugiego rodzaju są analogiczne działania dokonywane na kolumnach macierzy.

W klasie tej dopisz funkcje składowe implementujące przekształcenia elementarne na macierzach (przestawienie dwóch wierszy/kolumn, pomnożenie dowolnego wiersza/kolumny przez liczbę różną od zera, dodanie do wiersza/kolumny krotności innego wiersza/kolumny). Dopisz też funkcję generującą macierz powstałą z oryginalnej przez wykreślenie zadanego wiersza i kolumny. Funkcje te mają ułatwić obliczenie wyznacznika macierzy oraz macierzy odwrotnej.

Zaprogramuj także zaprzyjaźnione operatory do czytania ze strumienia wejściowego `operator>>` i pisanego do strumienia wyjściowego `operator<<` dla macierzy.

```

class Macierz {
// ...
public:
    friend istream operator>> (istream &we, Macierz &x);
    friend ostream operator<< (ostream &wy, const Macierz &x);
};

```

Zaprojektuj własną hierarchię wyjątków dla operacji macierzowych. Klasa stojąca na szczycie tej hierarchii powinna dziedziczyć po `std::exception`.

Na koniec napisz program, który wczyta macierz kwadratową ze standardowego wejścia, a następnie obliczy i wypisze na standardowe wyjście wyznacznik tej macierzy i macierz odwrotną.

Uzupełnienie.

Definicję klasy `macierz` umieść w przestrzeni nazw `obliczenia`.

Elementy w programie, na które należy zwracać uwagę.

- Podział programu na pliki nagłówkowe i źródłowe.
- Definicja operacji macierzowych.
- Definicja hierarchii klas wyjątków.
- W funkcji `main()` należy przetestować całą funkcjonalność macierzy.