

Zadanie 1

Napisz dwie funkcje operujące na tablicach dwuwymiarowych (macierzach).

```
void mult(const double* a, const double* b, double* c,
          int ra, int carb, int cb);
void prmat(const double* a, int ra, int ca);
```

Pierwsza z nich, **mult**, pobiera trzy macierze liczb typu **double** przekazane jako wskaźnik do pierwszego elementu pierwszego wiersza (czyli odpowiednim typem parametru funkcji jest **double*** lub **const double***). Macierz **a** ma wymiary $ra \times carb$ a macierz **b** — $carb \times cb$. Obie te macierze są zainicjowane przez użytkownika i nie są przez funkcję **mult** modyfikowane. Macierz **c** ma wymiary $ra \times cb$ i zadaniem funkcji jest wpisanie do niej iloczynu macierzy **a** i **b**.

Druga funkcja, **prmat**, wypisuje na ekran przekazaną (jako wskaźnik do pierwszego elementu pierwszego wiersza) macierz **a** wierszami; **ra** i **rb** oznaczają, odpowiednio, liczbę wierszy i kolumn tej macierzy.

Na przykład następujący program

```
download MatMul.cpp

#include <iostream>
#include <iomanip> // if you want to pretty print in prmat

void mult(const double* a, const double* b,
          double* c, int ra, int carb, int cb) {
    // ...
}

void prmat(const double* a, int ra, int ca) {
    // ...
}

int main() {
    using std::cout; using std::endl;
    double a[2][3] = { {3,2,1}, {6,5,4} };
    double b[3][2] = { {1,2}, {4,5}, {-6,-7} };
    double c[2][2];
    cout << "a a[0] " << sizeof(a)
          << " " << sizeof(a[0]) << endl;
    cout << "b b[0] " << sizeof(b)
          << " " << sizeof(b[0]) << endl;
    // explain why a[0] or (double*)b
    mult(a[0], (double*)b, c[0], 2, 3, 2);
    prmat(c[0], 2, 2);
}
```

powinien wypisać

```
a a[0] 48 24
b b[0] 48 16
  5   9
  2   9
```

Wyjaśnij

- jak odwoływać się w ciele funkcji do elementów typu `a[i][j]` mając w `a` tylko adres pierwszego elementu pierwszego wiersza macierzy `a` (i podobnie dla pozostałych macierzy)?
- dlaczego wywołując funkcję nie możemy jako argumentu użyć po prostu `a` (lub `b` lub `c`), tylko musimy pisać albo `a[0]` albo rzutować `a` na `double*`?

Nie twórz żadnych pomocniczych tablic czy innych kolekcji.
