

Programmazione ad Oggetti

Homework 4 (Classe matrice)

Esercizio

Implementare i metodi della seguente classe Matrice (float):

```
class Matrice
{
    int rows, cols;                // numero di righe e colonne
    float **m;                    // puntatore alla matrice allocata dinamicamente
    void set(int val, int i, int j); // Assegna il valore val all'elemento m[i,j]
    float get(int i , int j );    // Restituisce il valore m[i,j]
    int getRows(); // Restituisce il numero di righe
    int getCols(); // Restituisce il numero di colonne
    // OPERATORI
    Matrice operator+(const Matrice &mat); // somma
    Matrice operator*(const Matrice &mat); // moltiplicazione
    bool operator==(const Matrice &mat); // uguaglianza: restituisce true se m == mat
    Matrice operator+=(Matrice mat); // forma abbreviata dell'operatore somma
    Matrice operator*=(Matrice mat); // forma abbreviata dell'operatore
    float operator()(int i , int j ); // Operatore di accesso (da usare a destra e SINISTRA)

    // FUNZIONI FRIEND (ESTERNE)
    friend Matrice operator+(float a, const Matrice mat); // somma un float (a sinistra)
    friend Matrice operator+(const Matrice mat, float a); // somma un float (a destra)
    friend Matrice operator*(float a, const Matrice mat); // moltiplica un float (a sinistra)
    friend Matrice operator*(const Matrice mat, float a); // somma un float (a destra)

    // OPERATORI DI I/O
    friend ostream& operator<<(ostream &os, Matrice mat); // Inseritore
    friend istream& operator>>(istream &in, Matrice mat); // Estrattore
};
```

SPECIFICHE

- Si usino i qualificatori di accesso in maniera tale da implementare l'incapsulamento delle variabili in maniera appropriata.
- Si implementino inoltre, il costruttore, il distruttore ed il costruttore di copia e l'operatore di assegnazione, tenendo conto del fatto che la memoria per la matrice è allocata dinamicamente e puntata dal puntatore m;
- Si usi, in maniera opportuna, il qualificatore const per:
 - Rendere le funzioni const
 - Passare per riferimento i parametri alle funzioni
 - Restituire i valori delle funzioni per riferimento
- per l'operatore di input (estrattore) si estraggano **ROWS** righe e **COLS** colonne.

SUGGERIMENTI

- si implementino la seguenti funzioni, che facilitano l'implementazione di costruttori ed operatore di assegnazione:
 - **float**** alloc_mat(**int** r, **int** c);

- che alloca dinamicamente una matrice di r righe e c colonne
- `Matrice::copia(const Matrice &src)`
- che copia in se stesso la matrice `src`
- per quanto riguarda la somma tra due matrici si ricorda che date 2 matrici **A** e **B**, la matrice somma **A+B** è ottenuta:

$$\mathbf{A} + \mathbf{B} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & \cdots & b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{m1} & b_{m2} & \cdots & b_{mn} \end{bmatrix}$$

$$= \begin{bmatrix} a_{11} + b_{11} & a_{12} + b_{12} & \cdots & a_{1n} + b_{1n} \\ a_{21} + b_{21} & a_{22} + b_{22} & \cdots & a_{2n} + b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} + b_{m1} & a_{m2} + b_{m2} & \cdots & a_{mn} + b_{mn} \end{bmatrix}$$

pertanto due matrici possono essere sommate se e solo hanno lo stesso numero di righe e di colonne

- per quanto riguarda il prodotto tra due matrici si ricorda che date 2 matrici **A** e **B**, dove la matrice **A** è composta da n righe e m colonne e la matrice **B** è composta m righe e p colonne:

$$\mathbf{A} = \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1m} \\ A_{21} & A_{22} & \cdots & A_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nm} \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} B_{11} & B_{12} & \cdots & B_{1p} \\ B_{21} & B_{22} & \cdots & B_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ B_{m1} & B_{m2} & \cdots & B_{mp} \end{pmatrix}$$

la matrice prodotto **AB** è composta n righe e p colonne:

$$\mathbf{AB} = \begin{pmatrix} (\mathbf{AB})_{11} & (\mathbf{AB})_{12} & \cdots & (\mathbf{AB})_{1p} \\ (\mathbf{AB})_{21} & (\mathbf{AB})_{22} & \cdots & (\mathbf{AB})_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ (\mathbf{AB})_{n1} & (\mathbf{AB})_{n2} & \cdots & (\mathbf{AB})_{np} \end{pmatrix}$$

è l'elemento $(\mathbf{AB})_{ij}$ è ottenuto dalla seguente formula:

$$(\mathbf{AB})_{ij} = \sum_{k=1}^m A_{ik} B_{kj}.$$

pertanto due matrici possono essere moltiplicate fra loro se e solo se il numero di colonne della prima è uguale al numero di righe della seconda