

```

#include <iostream>
#include <iomanip>
using namespace std;

// Klassendefinition
class Matrix_3x3 {
private:
    int m[3][3];

public:
    Matrix_3x3();
    void ausgabe() const;
    void einlesen();
    void set(int value, int i, int j);
    int get(int i, int j) const;
    void addieren(const Matrix_3x3 & ref_m);
    void addieren(const Matrix_3x3 & ref_m1, const Matrix_3x3 & ref_m2);
    int determinante() const;
};

// Konstruktor: 3x3-Matrix initialisieren
Matrix_3x3 :: Matrix_3x3()
{
    for(int i=0;i<3;i++)
        for(int j=0;j<3;j++)
            m[i][j] = 0;
}

// Ausgabe der Matrixelemente
void Matrix_3x3 :: ausgabe() const
{
    cout << endl;
    cout << "Ausgabe" << endl;
    cout << "-----" << endl << endl;
    for(int i=0;i<3;i++){
        for(int j=0;j<3;j++){
            cout << setw(5) << m[i][j];
            cout << endl << endl;
        }
    }
}

// zeilenweises Einlesen der Matrixelemente
void Matrix_3x3 :: einlesen()
{
    cout << endl << endl;
    cout << "Einlesen der Matrix" << endl;
    cout << "-----" << endl << endl;
    for(int i=0;i<3;i++){
        cout << "\nGeben Sie die " << i+1 << ". Zeile ein: ";
        for(int j=0;j<3;j++)
            cin >> m[i][j];
    }
}

```

```

// Setzen eines Wertes an Zeile i und Spalte j
void Matrix_3x3 :: set(int value, int i, int j)
{
    m[i][j] = value;
}

// Ermitteln eines Wertes an Zeile i und Spalte j
int Matrix_3x3 :: get(int i, int j) const
{
    return m[i][j];
}

// Addition der Matrix ref_m zu aktuellen Matrix
void Matrix_3x3 :: addieren(const Matrix_3x3 & ref_m)
{
    for(int i=0;i<3;i++){
        for(int j=0;j<3;j++){
            set(ref_m.get(i,j)+get(i,j),i,j);
        }
    }
}

// Addition der Matrizen ref_m1 und ref_m2 sowie Zuweisung zur aktuellen
void Matrix_3x3 :: addieren(const Matrix_3x3 & ref_m1, const Matrix_3x3 &
ref_m2)
{
    for(int i=0;i<3;i++){
        for(int j=0;j<3;j++){
            set(ref_m1.get(i,j)+ref_m2.get(i,j),i,j);
        }
    }
}

// Determinante der Matrix errechnen
int Matrix_3x3 :: determinante() const
{
    return m[0][0]*m[1][1]*m[2][2] +
        m[0][1]*m[1][2]*m[2][0] +
        m[0][2]*m[1][0]*m[2][1] -
        m[0][2]*m[1][1]*m[2][0] -
        m[0][0]*m[1][2]*m[2][1] -
        m[0][1]*m[1][0]*m[2][2];
}

// Hauptprogrammfunktion
int main()
{
    Matrix_3x3 matrix1, matrix2, matrix3;
    matrix1.einlesen();
    matrix2.einlesen();
    matrix1.addieren(matrix2);
    matrix1.ausgabe();
    cout << "Determinante = " << matrix1.determinante() << endl;
    matrix3.addieren(matrix1,matrix2);
    matrix3.ausgabe();
    return 0;
}

```