

2. Programmierung in C

2.1. Beispiele

```
/* ----- */
/* Programmbeispiel: Vektoren & Matrizen */
/* ----- */
#include "matrix.h"
#define DIM 5

int main(void)
{
    double vect[DIM], prod[DIM];           /* Vektoren definieren */
    double matr[DIM][DIM];                /* Matrix definieren */

    v_init(DIM, vect, MAT_INIT_RANDOM);   /* Vektor mit Zufallszahlen (0...1) füllen */
    m_init(DIM, matr, MAT_INIT_IDENTITY); /* Einheitsmatrix erzeugen */
    m_fmula(DIM, matr, 2.0);               /* Alle Matrixelemente verdoppeln */
    m_vmula(DIM, prod, matr, vect);        /* Matrix mit Vektor multiplizieren */

    /* Matrix und Vektoren auf Konsole ausgeben */
    printf("matr = \n"); m_print(DIM, matr); printf("\n");
    printf("vect = \n"); v_print(DIM, vect); printf("\n");
    printf("prod = \n"); v_print(DIM, prod); printf("\n");

    /* Matrix invertieren und auf Konsole ausgeben */
    printf("Inverse Matrix = \n");
    m_invert(DIM, matr);
    m_print(DIM, matr);

    return 0;
}

/* ----- */
/* Programmbeispiel: lineares Gleichungssystem */
/* ----- */
#include "matrix.hpp"
#include <stdio.h>
#define DIM 3

int main(void)
{
    double x[DIM];
    double y[DIM] = {280, 340, 190};      /* Vektor y mit Werten belegen */
    double a[DIM][DIM] =                 /* Matrix A mit Werten belegen */
    {
        {2, 4, 6},
        {3, 5, 7},
        {4, 6, 1}
    };

    if(MAT_OK == lin_solve(DIM, x, a, y)) /* Gleichungssystem Ax = y lösen */
        v_print(DIM, x);                 /* Ausgabe: 10.00; 20.00; 30.00 */
    else
        printf("Gleichungssystem kann nicht geloest werden!\n");
    return 0;
}
```