



## Matrix Bibliothek

Darstellung von Matrizen beliebiger Dimension sowie grundlegende Operationen (Addition, Multiplikation, Lösung von linearen Gleichungssystemen, Invertieren, Berechnung der Determinante).

(Zum Test ist eine 7 Tage Demo Lizenz verfügbar.)

### Produktbeschreibung

#### Lizenzierung:

Arbeitsplatzlizenz

Die Matrix Bibliothek bietet einen Datentypen für Matrizen beliebiger Dimension sowie grundlegende Operationen auf Matrizen.

#### Datentypen und Funktionen

Matrizen werden durch den Datentypen `mtx.Matrix` abgebildet. `mtx.Matrix` speichert eine Matrix als ARRAY von LREAL Werten. Die Werte sind zeilenweise abgelegt.

Grundlegende arithmetische Operationen werden als Funktionen angeboten, die 3 Matrizen als VAR\_IN\_OUT Argumente haben: Ergebnis, Quelle, Ziel. Die Funktion zum Addieren von Matrizen hat z.B. folgende Schnittstelle:

```
(* Adds two matrices : C := A + B.
 * A, B, and C must have identical dimensions.
 * Note: A, B, and C may all be the same matrix. *)
FUNCTION AddM : ResultCode
VAR_IN_OUT
  C : Matrix ; (* The result *)
  A : Matrix ; (* The first summand *)
  B : Matrix ; (* The second summand *)
END_VAR
```

Folgende arithmetische Operationen sind enthalten:

- Addition von Matrizen (elementweise): `AddM`
- Subtraktion von Matrizen (elementweise): `SubM`
- Multiplikation von Matrizen (elementweise): `TimesM`
- Division von Matrizen (elementweise): `RDivideM`
- Multiplikation einer Matrix mit einem Skalar: `MultMS`
- Multiplikation von Matrizen: `MultM`
- Transponieren einer Matrix: `TransposeM`

Außerdem gibt es einige Hilfsfunktionen um Matrizen zu initialisieren, zu kopieren und auf Elemente zuzugreifen:

- Initialisierung durch ARRAY von Werten: `InitMatrix`
- Kopieren von Elementen aus einem ARRAY: `CopyElems`
- Kopieren von Matrizen gleicher Dimension: `CopyMatrix`
- Initialisierung zur Einheitsmatrix: `IdentityMatrix`
- Initialisierung zur Nullmatrix: `ZeroMatrix`
- Elemente lesen und Schreiben: `Elem`, `SetElem`

Schließlich gibt es noch einige komplexere Funktionen:

- Lösung eines linearen Gleichungssystems  $A * X = B$ : `SolveLU`
- Invertierung einer quadratischen Matrix: `InvertLU`
- Berechnung der Determinante einer quadratischen Matrix: `DeterminantLU`
- Bestimmung einer LU-Zerlegung (diese Zerlegung dient als Basis für die obigen drei Funktionen): `DecomposeLU`

## Speicherverwaltung

Die Speicherverwaltung wird dem Nutzer der Bibliothek überlassen. Matrizen werden durch Übergabe eines Zeigers auf den Speicher (über die Funktion `InitMatrix`) initialisiert. Je nach Anwendung kann der Anwender den Speicher passend bereitstellen.

Darüber hinaus gibt es noch die Hilfsbausteine `MatrixS`, `ColVectorS` und `RowVectorS`, die Matrizen über `ARRAYS` konstanter Größe initialisieren. Sie implementieren die Schnittstelle `IMatrixAllocator`, die auch dem Anwender offen steht.

## Matrix Examples

Die `Matrix Examples` zeigen die einzelnen Funktionen der Bibliothek im Detail. Ebenso wird zur Veranschaulichung eine Beispielaufgabe zum Lösen eines linearen Gleichungssystems gezeigt.

Für weiterführende Beispiel sieht das Projekt „`MatrixTests.project`“ zur Verfügung.

## Allgemeine Informationen

### Lieferant:

CODESYS GmbH  
 Memminger Straße 151  
 87439 Kempten  
 Deutschland

### Support:

<https://support.codesys.com>

### Artikelname:

Matrix Library

### Artikelnummer:

2111000003

### Vertrieb:

CODESYS Store

<https://store.codesys.com>

### Lieferumfang:

- CODESYS Software und / oder Lizenzschlüssel mit Rechnungsinformation
- Bei Schulungen und Veranstaltungen: Buchungsbestätigung

## Systemvoraussetzungen und Einschränkungen

<b>Programmiersystem</b>	CODESYS Development System Version 3.5.14.0 oder höher
<b>Laufzeitsystem</b>	CODESYS Control Version 3.5.2.0 oder höher
<b>Unterstützte Plattformen/ Geräte</b>	Alle
<b>Zusätzliche Anforderungen</b>	-
<b>Einschränkungen</b>	-
<b>Lizenzierung</b>	Arbeitsplatzlizenz: Die Lizenz kann auf dem Arbeitsplatz verwendet werden, auf dem Sie das CODESYS Development System installieren und ausführen.  Die Lizenzaktivierung erfolgt auf einem softwarebasierten Lizenz-Container (Soft-Container), der fest an den Arbeitsplatz gebunden ist. Alternativ kann die Lizenz auf einem CODESYS Key (USB-Dongle) hinterlegt werden. Durch Umstecken des CODESYS Keys kann die Lizenz an einem anderen Arbeitsplatz genutzt werden.
<b>Erforderliches Zubehör</b>	CODESYS Key for CODESYS < 3.5.14.0

*Bitte beachten Sie: Nicht alle CODESYS-Funktionen sind in allen Ländern verfügbar. Weitere Informationen zu diesen länderspezifischen Einschränkungen erhalten Sie unter [sales@codesys.com](mailto:sales@codesys.com).*

*Bitte beachten Sie: Technische Änderungen, Druckfehler und Irrtümer vorbehalten. Es gilt der Inhalt der aktuellen Online-Version dieses Dokuments.*