
A Programmierrichtlinien für MATLAB

A.1 Richtlinien für Bezeichner

- Bezeichner sind natursprachliche oder problemnahe Namen oder verständliche Abkürzungen solcher Namen.
- Kein Bezeichner beginnt mit einem Unterstrich
- Generell ist Groß-/Kleinschreibung zu verwenden.
- Zwei Bezeichner dürfen sich nicht nur bezüglich der Groß/Kleinschreibung unterscheiden.
- Die Länge des Bezeichners sollte zwischen 5 und 15 Buchstaben sein.
- Für Laufvariablen in for-Schleifen und als Array-Indizes sind die Kleinbuchstaben `i, j, k, ...` zu verwenden.
- Bestehen Bezeichner aus mehreren Worten, dann beginnt jedes Wort mit einem Großbuchstaben, z.B. `AnzahlWorte`.
- Funktionen und Operationen führen Aktionen aus, daher sollte der Name ein bezeichnendes Verb enthalten.
Beispiel: `Sortiere(int nZahlen []) ;`
- Besteht die Wirkung einer Operation nur im Lesen bzw. Schreiben von Instanzvariablen, dann sollte der Name mit `Get` bzw. `Set` beginnen.
Beispiel: `Wert=GetXKoord();`
- Variablen sind detailliert zu beschreiben, z.B. `nZeilenZaehler, chDateiStatus, ...`
- Bei Variablen vom Typ `bool` sollte jeder Bezeichner eindeutig angegeben sein. Beispiel: `bRichtig, bOffen, ...`

A.2 Folgende Präfixe besitzen eine spezielle Bedeutung

gl-	globale Variable
const-	Konstanten
s-	Structs
a-	Array
h-	Handles auf Matlab-Objekte

A.3 Richtlinien für Bezeichner

ganze Zahlen	n-
Gleitkommazahlen	f-
boolean Expressions	b-
einzelne Zeichen	c-
Zeichenketten	s-

Matlab unterscheidet nicht zwischen ganzen Zahlen oder Gleitkommawerten, eine Wertzuweisung zwischen ihnen ist problemlos möglich. Aus Gründen besserer Verständlichkeit sollten die oben gen. Präfixe dennoch verwendet werden.

Mehrfacheigenschaften sind entsprechend aufzubauen

an-	Array of int
as-	Array of Struct

A.4 Richtlinien für die Formatierung

Leerzeichen

- Bei Operatoren werden Operanden und Operator durch jeweils ein Leerzeichen getrennt.
Beispiel: `nZahl1 + nZahl2 * 3;`
- Zwischen Funktionsname und Klammer steht kein Leerzeichen. Nach der öffnenden und vor schließenden Klammer steht ebenfalls kein Leerzeichen.
Beispiel: `SetBit(nstate,3);`

Leerzeilen

- Leerzeilen sollen Blöcke von logisch zusammenhängenden Anweisung trennen. Der Deklarationsteil soll stets durch eine Leerzeile von den Anweisungen getrennt werden.
- Jeder Funktionsdeklaration soll eine Leerzeile vorausgehen.
- Umfangreiche Kontrollstrukturen sind durch eine Leerzeile zu trennen.
- Umfangreiche Deklarationen sind durch eine Leerzeile zu trennen.

Einrücken von Strukturen

Alle Strukturen sind entsprechend ihrer `for / if / ... end` einzurücken.

Verwendung von Feldern (Arrays)

Felder sollen immer vollständig initialisiert werden.

Beispiel: `nMatrix=zeros(2,4)`

Variablendeklaration

Variablen sollten zu Beginn eines Blocks initialisiert werden.

Verwendung von Konstanten

Symbolische Konstanten werden durchgehend mit Großbuchstaben geschrieben, wobei längere Bezeichnungen zur besseren Lesbarkeit durch Unterstriche gegliedert werden.

Beispiel: `SAMPLES_PRO_PERIODE`