

Einführung in die Informatik I

Kapitel I.4: Logische Ausdrücke

Prof. Dr.-Ing. Marcin Grzegorzek

Juniorprofessur für Mustererkennung im Institut für Bildinformatik

Department Elektrotechnik und Informatik

Fakultät IV der Universität Siegen

07.11.2012

Inhaltsverzeichnis

- I. MATLAB-Einführung
 - 1. Voraussetzungen und Konventionen
 - 2. Variablen und arithmetische Ausdrücke
 - 3. Automatisierung von Berechnungen
 - 4. Logische Ausdrücke**
 - 5. Verzweigungen
 - 6. Schleifen
 - 7. Fehlersuche in Programmen
 - 8. Funktionen
 - 9. Arbeitsweise von Funktionen
 - 10. Vektoren
 - 11. Matrizen
- II. Algorithmen
- III. MATLAB-Fortsetzung
- IV. Wissenschaftliche Werkzeuge

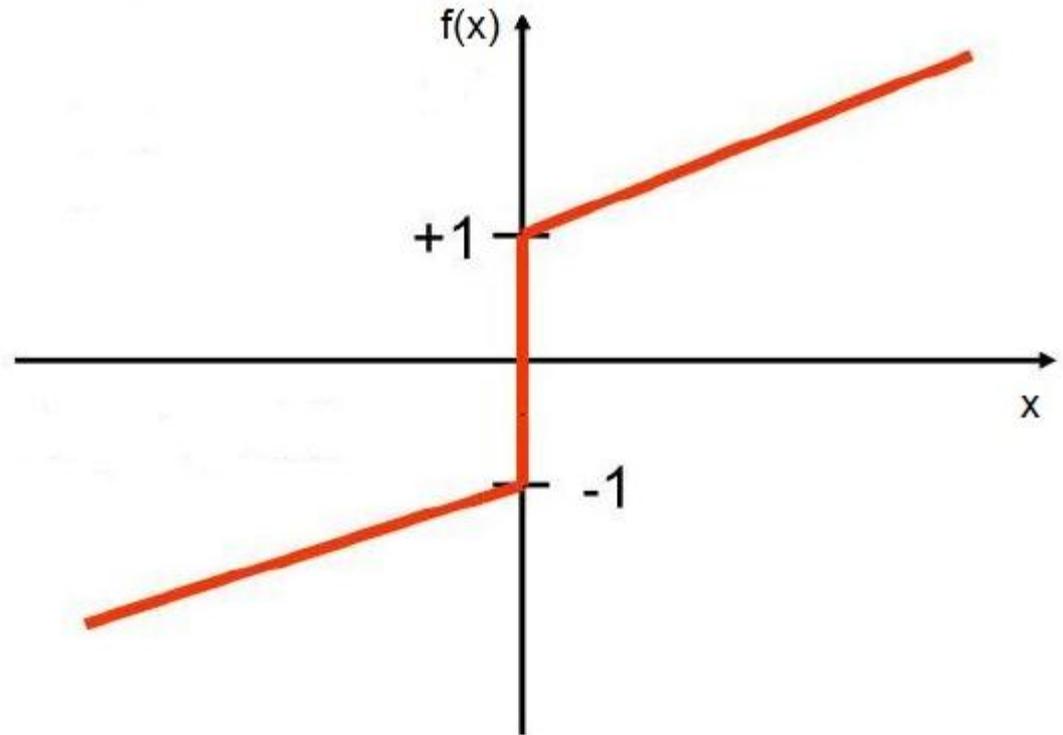
Wahrheitswerte/ Vergleichsoperatoren

- MATLAB drückt logische Werte durch Null (wahre Aussage) und Eins (falsche Aussage) aus. Die Vergleichsoperationen bestimmen aus zwei (**Beachte: Unterschied zur Mathematik, z.B. $0 < x < 7$ existiert in der Programmierung nicht!**) Zahlen einen Wahrheitswert
 - $8 > 5$ bedeutet: Ist 8 größer als 5?, Ja!, also Ergebnis: 1
 - $8 < 5$ bedeutet: Ist 8 kleiner als 5?, Nein!, also Ergebnis: 0
- MATLAB kennt folgende **logische Vergleichsoperatoren**:

Operator	Bedeutung
>	größer
<	Kleiner
<=	Kleiner gleich (Beachte: Unterschied zur Mathematik: \leq)
>=	Größer gleich (Beachte: Unterschied zur Mathematik: \geq)
==	Gleich (Beachte: Vergleich == ist etwas anderes als Wertzuweisung =)
~=	ungleich \neq

Beispiel: Stückweise definierte Funktion

$$f(x) = \begin{cases} 0.5 \cdot x - 1, & \text{falls } x < 0 \\ 0, & \text{falls } x = 0 \\ 0.5 \cdot x + 1, & \text{falls } x > 0 \end{cases}$$



mit Vergleichsoperatoren formuliert:

$$f(x) = (0.5 \cdot x - 1) \cdot (x < 0) + (0.5 \cdot x + 1) \cdot (x > 0)$$

Verknüpfungen: Logische Operatoren

Operator	Bedeutung	Bemerkung
&	Und	Beide Bedingungen müssen erfüllt sein, damit die Gesamtaussage wahr wird.
	Oder	Eine Bedingung muss erfüllt sein, damit die Gesamtaussage wahr wird.
~	Nicht	Invertiert die Bedingung (d.h. kehrt sie um)

- **UND-Verknüpfung &**

- Abitur: $\text{max}(\text{Abi_noten}) \leq 4 \& (\text{sum}(\text{Punkte}) \geq 300)$
- Auto: $(\text{Maengel} == 0) \& (\text{Preis} > 3.5e5) \& (\text{Preis} < 1.2e4)$

- **ODER-Verknüpfung |**

- Türsteher: $(\text{Outfit} > 12) | ((\text{Outfit} > 2) \& (\text{mit_Frau} > 0))$

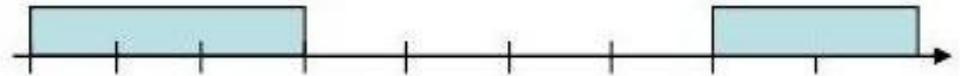
Beispiele für logische Ausdrücke

- Logische Grundoperationen und Vergleichsoperationen

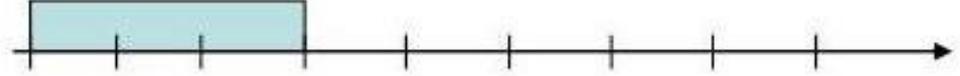
- $(x > 3) \ \& \ (x < 4)$



- $(x < 3) \ | \ (x > 4)$



- $\sim(x > 3)$

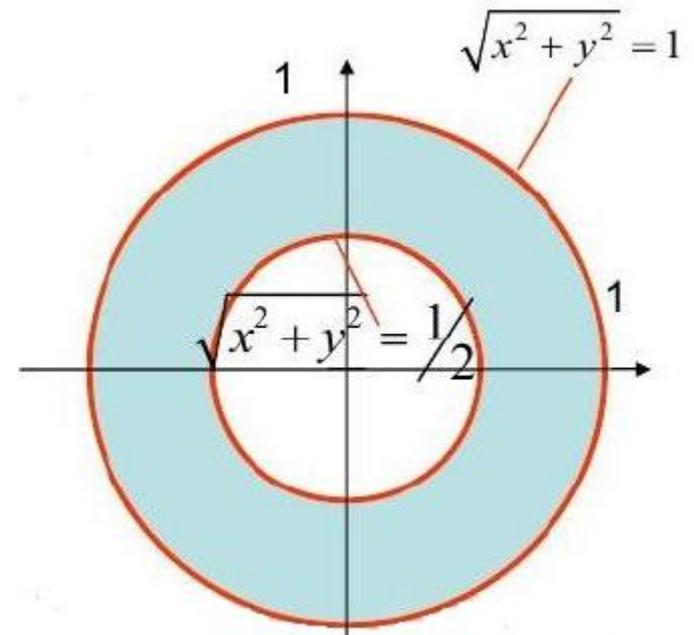


3

4

- Zweidimensionales Beispiel

- $\text{sqrt}(x^2 + y^2) > 0.5 \ \& \ \text{sqrt}(x^2 + y^2) < 1$



Erweiterte Operatorprioritäten

- Operatoren können miteinander vermischt werden, deshalb müssen die **Operatorprioritäten** dringend berücksichtigt werden:

1.	Klammern	(...)
2.	Potenzieren	\wedge
3.	Unäre Operatoren	+ - ~
4.	Punktoperatoren	* /
5.	Strichoperatoren	+ -
6.	Vergleichsoperatoren	< > ==
7.	Und-Operator	&
8.	Oder-Operator	

- Beispiele:
 - $(2^3 > 1 \& 3/2 < 4)$ ergibt 1
 - $2^{\left((3 > 1) \& (3 / (2 < 4) < 4)\right)}$ ergibt 2

Befehle mit Nebeneffekten

- **Manche Funktionen** bewirken außer der Berechnung des Ergebnisses auch noch etwas anderes (**Nebeneffekte**). Beispiel: $p=plot(1,1)$
 - Nebeneffekt: Es öffnet sich ein Fenster und der Punkt (1,1) wird in ein Koordinatensystem eingezeichnet.
 - „Rechenergebnis“: p ist die Nummer des geöffneten Fensters.
- Solche Befehle sollte man **nicht direkt in logischen Ausdrücken verwenden**. Beispiel: $x=(1>2) \& (plot(1,1)>0)$
 - Grund: Wenn der erste Teil schon falsch ist, wird in vielen Programmiersprachen der zweite Teil nicht mehr ausgewertet.
 - Abhilfe durch Aufspalten in zwei Befehle mit Zwischenspeicherung:
 $p=plot(1,1);$
 $x=(1>2) \& (p>0);$
- MATLAB unterscheidet $\&$ und $\&\&$ für **volle Auswertung** des Ausdrucks und **Teilauswertung** (Shortcut „AND“). Beide sind möglich, wenn keine Nebeneffekte vorkommen.
- Analoges gilt für $|$ und $||$ (Shortcut „OR“).