

Vergleichs- und Test-Operationen

Zur Erledigung komplexer Aufgaben mittels Shell-Skripten, ist es oftmals notwendig Zahlenwerte oder Zeichenketten miteinander zu vergleichen. Für diese Aufgabenstellung existiert im Verzeichnis `/usr/bin` das Kommando `test`. Mit `test` lassen sich vielfältige Vergleichs- und Test-Operationen durchführen¹.

Das Kommando `test` ist im eigentlichen Sinne eine **Funktion**, die **zwei Rückgabewerte** kennt:

1. Die Vergleichs- oder Test-Operation ist **TRUE**: `test` liefert den Wert 0
2. Die Vergleichs- oder Test-Operation ist **FALSE**: `test` liefert den Wert 1

Vermutung meinerseits: Üblicherweise liefern Programme den `Errorlevel 0`, wenn während der Programmausführung **kein Fehler** aufgetreten ist. Daher der Rückgabewert 0 bei **TRUE**.

Vergleich von Zahlenwerten

Es lassen sich Ganzzahlwerte (Integer) mit fünf Vergleichsoperationen vergleichen.

Aufgabe 1

Ergänzen Sie die Tabelle mit den möglichen Vergleichsoperatoren, Beispielen und Rückgabewerten. Nutzen Sie hierzu das Buch "UNIX Shellprogrammierung" Seite 74 f.

V-Operation	V-Operator	Beispiel	Rückgabewert
gleich		<code>test 15 20</code>	
größer		<code>test 15 20</code>	
kleiner		<code>test 15 20</code>	
größer gleich		<code>test 25 20</code>	
kleiner gleich		<code>test 20 30</code>	

Überprüfen Sie den Rückgabewert für alle Zeilen der Tabelle mittels Rechner durch Eingabe in der Kommandozeile. Nutzen sie hierzu folgende Befehlskonstruktion (**als Beispiel!**):

```
uhomm@obelix:~>test 100 -eq 200; echo $?
```

Die Vergleichsoperation. Das nachfolgende Komma trennt die beiden Befehle `test` und `echo` voneinander.

Die Variable `$?` enthält den Rückgabewert des letzten Befehls

Zur besseren Lesbarkeit gibt es eine **weitere Schreibweise** für das Kommando `test`, die vor allem in Shell-Skripten verwendet wird. Die beiden nachfolgende Beispiele sind identisch:

```
test 100 -eq 200 ist identisch mit [ 100 -eq 200 ]
```

Die Vergleichsoperation wird einfach in eckigen Klammern geschrieben! Die eckigen Klammern rufen `test` auf.

Wichtig: Hinter der ersten und vor der zweiten eckigen Klammer muss ein Leerzeichen stehen!!!

Aufgabe 2

Geben Sie alle Beispiele aus der obigen Tabelle in der Alternativ-Schreibweise in der Kommandozeile ein!

¹ Krienke, Rainer: UNIX Shell-Programmierung, München Wien, S. 74f.
 © Uwe Homm Version vom 8. Dezember 2004 D:\Schule\Lehrgang_NeuBetriebssysteme Neu\33 Shell-Skript Programmierung Vergleichs- und Test-Operationen.sdw

Vergleich von Zeichenketten

Häufig wird der Vergleich von zwei Zeichenketten oder der Test einer Zeichenkette auf deren Größe benötigt. Insgesamt sind vier Operationen möglich.

Aufgabe 3

Ergänzen Sie die Tabelle mit den möglichen Operatoren, Beispielen und Rückgabewerten.

Operation	Operator	Beispiel	Rückgabewert
gleich		["BEISPIEL" "Beispiel"]	
ungleich		["BEISPIEL" "Beispiel"]	
Länge ist Null		["STRING"]	
Länge ist ungleich Null		["STRING"]	

Innerhalb von Skripten wird man bei Vergleichsoperationen (gleich/ungleich) allerdings nicht mit zwei festen Zeichenketten arbeiten; normalerweise wird eine der beiden Zeichenketten entweder eine Shell-Variable oder ein Übergabeparameter sein.

Aufgabe 4

Formulieren Sie eine Bedingung, die auswertet, ob der angemeldete User der Superuser root ist.

Formulieren Sie eine Bedingung, die auswertet, ob ein erster Parameter des Shell-Skriptes existiert

Geben Sie das Test-Ergebnis mit der zuvor verwendeten Befehlskonstruktion mit `echo $?` aus

Test-Operationen

Gerade im Zusammenhang mit Dateien werden sehr häufig Test-Operationen benötigt. Zum Beispiel, ob eine bestimmte Datei oder ein bestimmtes Verzeichnis existiert.

Aufgabe 5

Formulieren Sie eine Bedingung, die auswertet, ob die Datei `passwd` im Verzeichnis `/etc` existiert

Formulieren Sie eine Bedingung, die auswertet, ob das Verzeichnis `public_html` im Home-Verzeichnis existiert

Geben Sie auch hier das Ergebnis des Testes mit der Befehlskonstruktion mit `echo $?` aus