

ANWENDER- DOKUMENTATION	Teil 3	MOS
11/87	Tabellenformatierung mit tbl	MUTOS 1700

Programmtechnische
Beschreibung

Teil 3

Tabellenformatierung mit tbl

AC A 7100/7150

VEB Robotron-Vertrieb Berlin

Ausgabe: 11/87

Die Ausarbeitung dieser Dokumentation erfolgte durch ein Kollektiv des VEB Robotron-Elektronik Dresden Stammbetrieb des VEB Kombinat Robotron.

Nachdruck und jegliche Vervielfältigung, auch auszugsweise, sind nur mit Genehmigung des Herausgebers zulässig.

Im Interesse einer ständigen Weiterentwicklung werden alle Leser gebeten, Hinweise zur Verbesserung der Dokumentation dem Herausgeber mitzuteilen.

Herausgeber:

VEB Robotron-Vertrieb Berlin
Mohrenstr. 62
Berlin 1086

(C) VEB Kombinat Robotron

Kurzreferat

Das Tabellenformatierungsprogramm tbl ist ein Preprozessor für das Textformatierungsprogramm nroff. Es übersetzt die mittels einer einfachen Formatierungssprache beschriebenen Tabellen in entsprechende nroff-verständliche Kommandos. Der übrige außerhalb der Tabellen liegende Text wird nicht bearbeitet.

1. EINLEITUNG
2. OPTIONS
3. FORMAT
4. DATEN
5. FORMATWECHSEL INNERHALB EINER TABELLE
6. BENUTZUNGSHINWEISE
7. BEISPIELE

1. Einleitung

~~~~~

Das Programm tbl ist ein Preprozessor für das Textformatierungsprogramm nroff. Seine Aufgabe ist es, das mit wenigen einfachen Anweisungen beschriebene Format der gewünschten Tabellen in die notwendigen nroff-Kommandos und -Steuerfolgen zu übersetzen.

Der von tbl zu formatierende Text ist in die Kommandos .TS und .TE (Tabellenstart bzw. Tabellenende) einzuschließen. Diese Kommandos müssen auch nroff als Makros zur Verfügung stehen.

Das allgemeine Format ist:

```
.TS  
Options;  
Format.  
Daten  
.TE
```

## 2. Options

~~~~~

Mit den Options werden allgemeine Rahmenbedingungen für die Formatierung der Tabelle eingestellt. Sie haben für die gesamte Tabelle Gültigkeit. Die Options müssen sofort dem .TS-Kommando folgen und enthalten eine Liste von Namen, die durch Leerzeichen oder Kommas getrennt und mit einem Semikolon ";" abgeschlossen werden.

Folgende Options sind zugelassen:

- center
Die Tabelle wird zentriert, d.h. sie wird in die Mitte der Seite positioniert. Standard ist linksausgerichtet.
- expand
Die Tabelle wird auf die laufende für nroff gültige Zeilenlänge erweitert.
- box
Die Tabelle wird durch Linien eingerahmt.
- allbox
Die Tabelle und jedes Feld der Tabelle werden eingerahmt.
- tab(x)
Verwendung von x statt TAB zum Trennen der Felder bei der Texteingabe.
- delim(x,y)
Die Zeichen x und y werden als Begrenzungszeichen für neqn-Anweisungen erkannt.

3. Format

Die Formatangaben legen die Art der Ausgabe der einzelnen Felder fest. Jede Zeile dieser Formatangaben korrespondiert mit einer Zeile der Tabelle, wobei eine Tabellenzeile nicht identisch mit einer Druckzeile sein muß. Die letzte Zeile der Formatangaben bestimmt das Format der entsprechenden Zeile der Daten und aller weiteren, falls mehr angegeben sind.

Die Formatspezifikationen innerhalb einer Zeile sind zur besseren Übersichtlichkeit durch Leerzeichen oder Tabulator voneinander zu trennen, die letzte Zeile ist mit einem Punkt (.) abzuschließen.

Die Schlüsselbuchstaben für die Feldformatierung sind:

- l Der Inhalt des Feldes wird linksausgerichtet.
- r Der Inhalt des Feldes wird rechtsausgerichtet.
- c Der Text wird innerhalb des Feldes zentriert.
- n Kennzeichnung eines Feldes mit numerischem Inhalt.
Die Zahlenwerte werden spaltengerecht rechtsausgerichtet untereinander formatiert. Enthalten die numerischen Werte einen Dezimalpunkt, so werden die Zahlen so ausgerichtet, daß der Dezimalpunkt in der gleichen Zeichenposition untereinander steht.
- a Unterspalte, gefüllt mit Zeichenketten, die linksausgerichtet und eingerückt dargestellt werden.
- s Kennzeichnung, daß eine Tabellenüberschrift dieses Feld bzw. diese Spalte mit überdecken soll.
- ^ Kennzeichnung, daß dieses Feld von dem darüberstehenden mit vereinnahmt wird (senkrechte Überschrift).

Zusätzlich zu den Schlüsselzeichen können jeweils weitere Zeichen nachgestellt werden, die die einzelnen Felder weiter spezifizieren; die Reihenfolge ist willkürlich.

- Horizontallinien:
Ein Schlüsselbuchstabe kann durch "_" (underscore) ersetzt werden, um anzuzeigen, daß dieses Feld durch eine waagerechte Linie ersetzt werden soll. Gibt es angrenzende waagerechte oder senkrechte Linien, so wird die Linie bis zu diesen durchgezogen.
- Vertikallinien:
Die Angabe eines senkrechten Striches zwischen den Formatspezifikationen kennzeichnet, daß die angrenzenden Felder durch einen senkrechten Strich zu trennen sind.
- Leerzeichen zwischen den Spalten:
Mit einer Zahl nach dem Schlüsselbuchstaben kann die Anzahl der Leerzeichen zwischen dieser und der nächsten Spalte festgelegt werden. Standard sind 3

Leerzeichen, die auch mindestens ausgegeben werden.

- Senkrecht mehrere Unterzeilen überspannende Überschrift:
Mit einem nachgestellten t (top) wird angezeigt, daß der Inhalt dieses Hauptfeldes nicht über den gesamten Bereich senkrecht zentriert wird, sondern ab Anfang des Feldes plaziert wird.
- Spaltenbreitenspezifizierung:
Ein w nach dem Schlüsselbuchstaben und eine in Klammern gesetzte Integer-Zahl spezifiziert die minimale Spaltenbreite; nützlich vor allem für Spalten, die Textblöcke enthalten. Normalerweise wird die notwendige Spaltenbreite automatisch berechnet.
- Gleich weite Spalten:
Alle Spalten, die mit einem zusätzlichen "e" gekennzeichnet werden, erhalten die gleiche Spaltenbreite.

4. Daten

Die Daten sind der eigentliche Tabelleninhalt. Sie werden nach der Formatspezifikation angegeben. Im Normalfall entspricht jeder Tabellenzeile eine Eingabezeile. Die einzelnen Spalten werden durch Tabulatoren oder durch ein anderes mit der Option `tab` definiertes Zeichen getrennt. Sehr lange Eingabezeilen können durch einen inversen Schrägstrich `\` unterbrochen werden. Folgende Spezialfälle sind zugelassen:

- `nroff`-Kommandos:
Eine Eingabezeile, beginnend mit einem Punkt ".", gefolgt von einem Buchstaben, wird als `nroff`-Kommando und nicht als Text interpretiert.
- Spaltenbreite waagerechte Trennlinien:
Eingabe eines "_" (underscore) bewirkt eine waagerechte Trennlinie in dem angegebenen Feld. Soll dieses Zeichen explizit in einem Feld erscheinen, muß ihm "&" vorgestellt oder ein Leerzeichen nachgestellt werden.
- Leerfeld:
Soll ein Feld leer bleiben, so wird das nächste Trennzeichen (TAB) bzw. das Zeilenendezeichen, falls es sich um das letzte Feld in der Zeile handelt, eingegeben.
- Tabellenbreite horizontale Trennlinien:
Eine Eingabezeile, die nur aus dem Zeichen "_" besteht, wird als Trennlinie über die gesamte Tabellenbreite interpretiert.
- Kurze waagerechte Linien:
Die Eingabe von "_" bewirkt eine waagerechte Linie innerhalb des angegebenen Feldes von der Länge des längsten Spalteninhaltes.
- Wiederholzeichen:
Durch Eingabe von `\Rx`, wobei `x` ein beliebiges Zeichen ist, wird erreicht, daß `x` so oft in das angegebene Feld eingetragen wird, wie der längste Spalteninhalt lang ist.
- Senkrecht mehrere Unterzeilen überspannende Hauptzeile:
Die Eingabe `^` bewirkt, daß dieses Feld von den Daten eines darüber stehenden Feldes mit vereinnahmt wird.
- Textblöcke:
Ein Textblock ist zusammenhängender Text innerhalb eines Tabellenfeldes. Er kann sich über mehrere Zeilen erstrecken. Jeder Textblock wird mit "T{" eröffnet und mit "T}" abgeschlossen.
Es ist folgende Form einzuhalten:

```
...T{  
Text.....  
.....  
T}...
```

Es ist darauf zu achten, daß das öffnende Symbol "{ " eine Zeile abschließt und das schließende Symbol "}" am Anfang einer neuen Zeile steht. Die Spaltenbreite wird berechnet, sie ist jedoch nicht von der Länge des Textes abhängig. Es empfiehlt sich deshalb, insbesondere bei unterschiedlich breit gewünschten Spalten, eine Weite anzugeben, entweder bei der Formatspezifikation w oder mit dem entsprechenden nroff-Kommando (.ll). Diese zweite Möglichkeit muß dann verwendet werden, wenn sich der Textblock als Überschrift über mehrere Spalten erstrecken soll (Formatspezifikation s). Die Formatierung des Textblockes wird von nroff übernommen, es stehen daher die dort zugelassenen Möglichkeiten zur Verfügung. nroff-Kommandos innerhalb des Textblockes haben nur für diesen Gültigkeit. nroff-Kommandos außerhalb des Textblockes haben keine Gültigkeit für dessen Formatierung.

5. Formatwechsel innerhalb einer Tabelle

Soll das Format innerhalb einer Tabelle gewechselt werden, z.B. bei Zwischenüberschriften, so ist das Kommando `.T&` einzugeben.
Der prinzipielle Aufbau einer solchen Tabelleneingabe hat folgendes Aussehen:

```
.TS
Optionen;
Format.
Daten
....
....
.T&
Format.
Daten
....
....
.T&
Format.
Daten
....
....
.TE
```

Es können keine neuen Optionen eingegeben werden und die maximale Anzahl und Weite der Spalten ist nicht mehr zu ändern.

6. Benutzungshinweise

Sind Gleichungen innerhalb von Tabellen angegeben, die mit dem Preprozessor neqn formatiert werden sollen, muß tbl vor neqn abgearbeitet werden.

Werden Gleichungen in mit dem Schlüsselbuchstaben n formatierten Spalten angegeben, so sind die mit der Option delim definierten Begrenzer zu verwenden, um ein Aufspalten der enthaltenen Zahlen zu vermeiden.

Zum Einordnen der Tabellen in die Seitengestaltung empfiehlt es sich, die Tabelle in die Makros .KS (Beginn eines zusammenhängenden Textteils) und .KE (Ende eines zusammengehörigen Textteils) einzuschließen. Statt .KS kann auch der Makro .KF (zusammengehörigen Text fließend in den laufenden Text einordnen) verwendet werden, um den sonst eventuell entstehenden Freiraum zu vermeiden. Diese Makros, wie auch die Makros .TS und .TE sind im Standardmakropaket enthalten.

Die Größe der Tabelle sollte auf eine Ausgabertextseite beschränkt werden. Größere Tabellen sind in mehrere Einzeltabellen zu unterteilen. Dies ist insbesondere dann erforderlich, wenn bei der Ausgabe das Kommando col verwendet wird. Das Kommando col ist bei gerahmten Tabellen und Tabellen mit Textblöcken der Formatierung durch nroff nachzuschalten, falls das Ausgabegerät keine vertikale Rückpositionierungen erlaubt.

7. Beispiele

Eingabe:

```
.TS
allbox expand;
c s c c
c c ^ ^
l c l l.
T{
.ll 20
Ergebnisse vor dem
Runden
T}T{
.ll 10
Rundungsmodus
TE}rgebnis
MantisseVorz.
normalisiert+aufrunden+Infinity
unnormalisiert+aufrundenT{
größter negativer Exponent, Mantisse unverändert
T}
.TE
```

Ausgabe:

.TS

Ergebnisse vor dem Runden		Rundungsmodus	Ergebnis
Mantisse	Vorz.		
normalisiert	+	aufrunden	+Infinity
unnormalisiert	+	aufrunden	größter negativer Exponent, Mantisse unverändert

.TE

Eingabe:

```
.TS
box;
lw(18)|lw(18)|lw(18)
l|l s
^|l|l.
1.Spalte      2.Spalte      3.Spalte

T{
.na
Senkrecht Tabellenhauptfeld,
nichtzentriert, als Textblock
T}          T{
.na
.ll 36
Waagrecht mehrere Unterspalten
überspannende Hauptspalte
mit Angabe der Zeilenlänge
T}

-          T{
Textblock in Unterspalte A
T}          T{
Textblock in Unterspalte B
T}
.TE
```

Ausgabe:

```
.TS
```

1.Spalte	2.Spalte	3.Spalte
Senkrecht Tabellenhauptfeld, nichtzentriert, als Textblock	Waagrecht mehrere Unterspalten überspannende Hauptspalte mit Angabe der Zeilenlänge	
	Textblock in Unterspalte A	Textblock in Unterspalte B

```
.TE
```

MUTOS 1700

Eingabe:

```
.TS
box;
cw(14) cw(14)|lw(28)
c|c|^
c s s
l s r
c s|c.
JA NEIN T{
Senkrecht mehrere Zeilen
überdeckendes Hauptfeld
T}

1.Feld 2.Feld

Hauptzeile einzeilig alle Spalten überdeckend

JA NEIN

T{
linkes Unterfeld als Textblock
T} T{
.ll 15
rechtes Unterfeld als Textblock
T}
.TE
```

Ausgabe:

```
.TS
```

JA	NEIN	
1.Feld	2.Feld	Senkrecht mehrere Zeilen überdeckendes Hauptfeld
Hauptzeile einzeilig alle Spalten überdeckend		
JA		NEIN
linkes Unterfeld als Textblock		rechtes Unterfeld als Textblock

```
.TE
```

Eingabe:

```
.TS
box;
l | c | l | c
l | lw(26) |c |lw(20) 2.
          Funktion           Argumente
```

```
/CR T{
Erzeugen einer Bibliothek
T} A :Länge:EPT:MNT:TYP
```

```
/CO T{
.na
Verdichten einer Bibliothek
T} A :Länge:EPT:MNT
```

```
/DE T{
Löschen von Moduln und ihrer Eintrittspunkte
T} A :Modulname
```

```
/DF T{
.na
Standardtyp der Bibliothek festlegen
T} A :Dateityp
.TE
```

Ausgabe:

.TS

	Funktion		Argumente
/CR	Erzeugen einer Bibliothek	A	:Länge:EPT:MNT:TYP
/CO	Verdichten einer Bibliothek	A	:Länge:EPT:MNT
/DE	Löschen von Moduln und ihrer Eintrittspunkte	A	:Modulname
/DF	Standardtyp der Bibliothek festlegen	A	:Dateityp

.TE

MUTOS 1700

Eingabe:

```
.TS
box;
cw(10) | cw(10) | cw(4) | cw(30)
c s s s.
Code      Standard      für      Bedeutung
```

1. CONTR --- beeinflusst die Taskbuilder-Ausführung

```
.T&
lw(10) | lw(10) | lw(4) | lw(30).
abort = N      --      mf      T{
Der Taskbuilder bricht das Bilden der Task ab.
.br
Wird verwendet, wenn durch frühere Fehler in der
Terminalfolge eine nichtnutzbare Task erzeugt wird.
.br
Taskbuilder stoppt das Akzeptieren der Eingabe und
erwartet eine erneute Eingabe.
T}
```

```
.T&
c s s s.
2. IDENT --- identifiziert die Taskeigenschaften
```

```
.T&
l | l | l | l.
Task = TN      T{
.na
Name der Taskabblidddatei
T}      mf      T{
deklariert den Namen der Task.
.br
TN --- Taskname (1...6 Radix50-Zeichen)
T}
```

```
T{
.nf
PAR = PN
[:basis
:lang]
T}      PAR = GEN      mf      T{
.na
Identifiziert die Partition, für die die Task gebildet wird.
.br
In Systemen mit Adreßzuweisung kann die Task in beliebiger
Partition installiert werden, die groß genug ist.
.br
In Systemen ohne Adreßzuweisung muß die Task in der Partition
installiert werden, in der sie laufen soll.
T}
.TE
```

Ausgabe:

.TS

Code	Standard	für	Bedeutung
1. CONTR --- beeinflusst die Taskbuilder-Ausführung			
abort = N	--	mf	Der Taskbuilder bricht das Bilden der Task ab. Wird verwendet, wenn durch frühere Fehler in der Terminalfolge eine nichtnutzbare Task erzeugt wird. Taskbuilder stoppt das Akzeptieren der Eingabe und erwartet eine erneute Eingabe.
2. IDENT --- identifiziert die Taskeigenschaften			
Task = TN	Name der Taskab- bilddatei	mf	deklariert den Namen der Task. TN --- Taskname (1...6 Radix50-Zeichen)
PAR = PN [:basis :lang]	PAR = GEN	mf	Identifiziert die Partition, für die die Task gebildet wird. In Systemen mit Adreßzuweisung kann die Task in beliebiger Partition installiert werden, die groß genug ist. In Systemen ohne Adreß- zuweisung muß die Task in der Partition installiert werden, in der sie laufen soll.

.TE

Eingabe:

```
.TS
box;
l|l|l
ltw(14) 3|cw(5) 3|lw(37).
Anweisung Form Operation
```

```
T{
.na
A
.br
(KOI7-Folge) Auch nicht druckbare Zeichen werden
in den Ausgabeblock gebracht.
T} %A T{
Das KOI7-Zeichen wird von der im Block der Task
stehenden Adresse in den Ausgabeblock gebracht.
T}
```

```
- %NA T{
Die nächsten n KOI7-Zeichen werden entsprechend in den
Ausgabeblock gebracht.
T}
```

```
- %VA T{
Der Wert des im Block der Task stehenden folgenden
Wortes wird als Wiederholungszähler benutzt und es
wird wie oben übertragen.
T}
```

```
.T&
lw(14) 3|cw(5) 3|lw(37).
T{
.na
B
.br
Byteweise Konvertierung von binär in oktal
T} %B T{
Das folgende Binär-Byte wird von der im Block der
Task stehenden Adresse in den Ausgabeblock übertragen,
wobei die entsprechende Konvertierung ausgeführt wird.
T}
```

```
- %NB T{
Es werden n Binär-Bytes entsprechend konvertiert und
übertragen. Zwischen den Zahlen sind Leerzeichen eingefügt.
T}
```

```
- %VB T{
übertragung wie bei %VA, Konvertierung von Binär-Byte
in oktal. Zwischen den Zahlen sind Leerzeichen eingefügt.
T}
.TE
```

|

Ausgabe:

.TS

Anweisung	Form	Operation
A (K0I7-Folge) Auch nicht druckbare Zei- chen werden in den Ausgabe- block gebracht.	%A	Das K0I7-Zeichen wird von der im Block der Task stehenden Adresse in den Ausgabeblock gebracht.
	%NA	Die nächsten n K0I7-Zeichen werden entsprechend in den Ausgabeblock gebracht.
	%VA	Der Wert des im Block der Task stehenden folgenden Wortes wird als Wiederholungszähler benutzt und es wird wie oben übertragen.
B Byteweise Kon- vertierung von binär in oktal	%B	Das folgende Binär-Byte wird von der im Block der Task stehenden Adresse in den Ausgabeblock übertragen, wobei die entsprechende Konvertierung ausgeführt wird.
	%NB	Es werden n Binär-Bytes entsprechend konvertiert und übertragen. Zwischen den Zahlen sind Leerzeichen eingefügt.
	%VB	Übertragung wie bei %VA, Konvertierung von Binär-Byte in oktal. Zwischen den Zahlen sind Leerzeichen eingefügt.

.TE