

Mit der Funktions-Prozedur *F_ELEMENT.PRO* lassen sich unter IDL [1, 2] aus einer Zeichenkette (string) Teil-Zeichenketten ausschneiden.

Dieses Programm ist eine erweiterte Nachbildung der DCL Lexical Function

F\$ELEMENT [3] des VAX/VMS-Betriebssystems der Firma Digital Equipment Corporation (DEC).

Das Programm steht im Internet zur Verfügung. Diese Beschreibung gibt den Stand des Programms in der Version V 01.5

(= V 1.0 – Edit 5) wieder. Es wurde von der Abt. Wiss. Datenverarbeitung in der „Interactive Data Language“ IDL [1] angefertigt und kann ggf. vom jeweiligen Anwender selbst modifiziert bzw. ergänzt werden.

1. Beschreibung:

Die neue IDL-Funktion *F_ELEMENT* extrahiert aus einer (längeren) Zeichenkette (string), die aus mehreren abgrenzbaren Elementen (substrings) besteht, ein bestimmtes Element oder auch alle Elemente. Ein zu extrahierendes Element (Teil-Zeichenkette) wird dabei durch seine Positionsnummer innerhalb der langen Zeichenkette gekennzeichnet.

Um mit *F_ELEMENT* aus einer gegebenen Zeichenkette (string) Teil-Zeichenketten (substrings) ausschneiden zu können, muß die Zeichenkette eindeutige Trennzeichen (delimiter) enthalten. Diese Trennzeichen können beliebige Zeichen sein. Bei *F_ELEMENT* kann als Trennzeichen auch eine kurze Zeichenkette verwendet werden.

2. Aufruf:

F_ELEMENT(elementNumber, delimiter,
string [, /ALL])

2.1 Variablen (VAR):

elementNumber — Die Nummer des Elements, das extrahiert werden soll. Die Zählung der Elemente beginnt mit Null. Falls die angegebene Nummer die in der Zeichenkette string vorhandene Anzahl von Elementen übersteigt, wird das Trennzeichen selbst übergeben.

delimiter — Das Zeichen, das zur Trennung der Zeichenkette verwendet werden soll. Als Trennzeichen läßt sich auch eine (kurze) Zeichenkette verwenden.

string — Die Zeichenkette, die durch die Operation zerlegt werden soll bzw. aus der Teil-Zeichenketten extrahiert werden sollen.

2.2 Keywords:

Durch die Angabe folgender Keywords kann der Ablauf des Programms modifiziert werden.

/ALL — Diese Möglichkeit ist gegenüber der DCL Funktion *F\$ELEMENT* [3] eine Erweiterung. Durch Angabe des Keywords */ALL* werden alle getrennten Elemente in einem String-Feld übergeben. Für die Variable „elementNumber“ kann dabei ein beliebiger (gültiger) Wert angegeben werden.

2.3 Beispiel(e):

Zum besseren Verständnis der Verwendung des Funktions-Programms *F_ELEMENT* werden im folgenden zwei Anwendungsbeispiele gegeben.

1. In einer Datei enthält die erste Zeile mehrere Kennwerte, die für die weitere Verarbeitung einzeln als Variablen benötigt werden (Aus: Programm *GET_EEG_DATA.PRO* – P 476/34).

Nachdem die vollständige Zeile auf die Variable „Kennung“ eingelesen ist, können aus dieser die einzelnen Kennwerte mit *F_ELEMENT* herausgetrennt werden. Dabei wird zunächst „:“ und in einem zweiten Schritt „ “ (Zwischenraum) als Trennzeichen verwendet. Die Umwandlung von Zeichenketten in Zahlenwerte erfolgt mit den IDL-Funktionen *Fix* bzw. *Float*.

```
Kennung = "TV:    200 TN:    800 AI:    5."
z = F_Element(1, ":", Kennung, /All)

tvor = Fix(F_Element(1, " ", z(1))) ; tvor = 200
tnach= Fix(F_Element(1, " ", z(2))) ; tnach= 800
tsamp= Float(F_Element(1, " ", z(3))) ; tsamp= 5.00
```

2. Mit der IDL-Funktion „*SysTime*“ soll die aktuelle System-Zeit ermittelt werden. Die einzelnen Zeitelemente sollen dann verschiedenen Variablen zugewiesen werden.

```
time = SysTime(0)
; z.B.: time = "Fri Jul 31 21:39:48 1992"
x = F_Element(1, " ", time, /All)

Print, x, Format='(A10)'
      Fri           ; = x(0)
      Jul           ; = x(1)
      31           ; = x(2)
      21:40:04     ; = x(3)
      1992         ; = x(4)

WoTag = x(0)
Tag = x(2)
Monat = x(1)
Jahr = x(4)
Uhr = F_Element(0, ":", x(3)) + "." + $
      F_Element(1, ":", x(3))
```

Beim ersten Aufruf von *F_ELEMENT* werden mit dem Trennzeichen „ “ (Zwischenraum) alle Elemente in das String-Feld *x* separiert. Zur Ermittlung der Uhrzeit (Stunden.Minuten) werden aus den Teil-Zeichenketten in *x(3)* mit dem Trennzeichen „:“ die Stunden und Minuten extrahiert und zur Zeitangabe zusammengesetzt.

3. Erforderliche Programme:

Dieses Programm verwendet ausschließlich die folgenden IDL-Funktionen: *StrCompress*, *StrLen*, *StrPos*, *StrMID*.

4. Sonstige Hinweise:

■ Zur Zeit keine.

5. Literatur:

- [1] Research Systems, Inc.: IDL – Interactive Data Language. Version 2.1. Boulder (USA): RSI 1991 (Edition vom 2.4.1991). E-Mail: idl@boulder.colorado.edu.
- [2] Dittberner, K.-H.: Interactive Data Language IDL: Eine Einführung. FU Berlin (IfP): wdv-notes Nr. 85, 1991.
- [3] Digital Equipment Corp.: VMS General User's Manual. VMS Version 5.0. Maynard (USA): DEC 1988. – Order No.: AA-LA98A-TE. – [CHB 554/2].

6. Copyright-Notiz:

© 1990-1992 – K.-H. Dittberner c/o Freie Universität Berlin. All rights reserved. No part of this software package may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any form by any means without the written permission of Karl-Heinz Dittberner. Permission will be granted for non-profit redistribution.