

RRZE

BENUTZERINFORMATION

BI 13 - ERLANGEN - 3. MÄRZ 1978

R R Z E

REGIONALES RECHENZENTRUM

MARTENSSTRASSE 1

8520 ERLANGEN

TEL: 09131 / 85 70 31 - 85 70 32

BETEILIGTE EINRICHTUNGEN :

UNIVERSITÄT ERLANGEN - NÜRNBERG MIT

RECHENZENTRUM

AUSSENSTELLE ERLANGEN INNENSTADT

AUSSENSTELLE ERLANGEN SÜDGELÄNDE

AUSSENSTELLE NÜRNBERG TUCHERGELÄNDE

AUSSENSTELLE NÜRNBERG FINDELGASSE

UNIVERSITÄT BAYREUTH

GESAMTHOCHSCHULE BAMBERG

FACHHOCHSCHULE COBURG

FACHHOCHSCHULE NÜRNBERG

HERAUSGEgeben VOM REGIONALEN RECHENZENTRUM ERLANGEN

INHALT:

1.	Aktuelle Information	2
1.1	Termine	2
1.2	Batch User's Guide	2
1.3	LFD-Engpaß	2
1.4	Magnetbandanforderung am TR440	2
1.5	Verhalten bei ungewollter Leitungsunterbrechung am TR440	2
1.6	TR440-Kommando RERUNSTART/Stuetzpunktschreiben	3
1.7	Protokoll des Benutzerkolloquiums v. 21.2.78	3
2.	Neues zum Stand der Software	5
2.1	Programmbibliothek	5
2.1.1	Umstellung der TR440-Kommandos der RRZE-Programmbibliothek	5
2.1.2	Neue Programmbibliotheken	5
2.1.3	STARG-TR440	6
2.1.4	UNRZPB - Programmbibliothek	6
2.1.5	CERN-Hinweis	7
2.1.6	STARG-Organisation	7
2.1.7	PASCAL, LISP	7
2.1.8	Statistik	7
2.1.9	ELAN	7
2.1.10	Editoren	7
2.1.11	ALGOL68C - TR440	7
2.2	Hinweise zur Benutzung von SIMULA	8
2.2.1	Schnelle Ein-/Ausgabe für Dialogprogramme	8
2.2.2	Fehler bei der Dateibearbeitung	8
2.3	Neue Grafik-Software	9

ANHANG:

- Software-Katalog
- ALGOL68C (Beschreibung)
- Programmiersprache ELAN (Beschreibung)
- Programm des Einführungskurses (Anmeldeformular)

1. Aktuelle Information

1.1 Termine

Es werden folgende Kurse ganztägig durchgeführt:

17.4. - 21.4. (HS 2.03)	Einführung in die Benutzung der Rechner TR440 und CYBER (Genaueres Programm im Anhang)
24.4. - 28.4.	Einführung in die Programmier- sprache APL.

1.2 Batch User's Guide

In der Aufsicht kann die Einführungsschrift

NOS Version 1 Batch User's Guide

zum Preis von 2,-- DM erworben werden.

1.3 LFD-Engpaß

Der Speicherplatz für LFD-Dateien an der TR440 ist zu einem Engpaß geworden. Deshalb werden ähnlich wie an der CYBER Dateien, auf die längere Zeit (2 Monate) nicht zugegriffen wurde, ausgelagert. Beachten Sie bitte die in der BI 12 (4.1.2.5) aufgezählten Hinweise zur Einsparung von Speicherplatz und löschen Sie nicht mehr benötigte Dateien in eigener Initiative.

1.4 Magnetbandanforderung am TR440

Die Anforderung des Betriebsmittels Magnetband sollte im XBA- (bzw. XBG-) Kommando über die Spezifikation B60 erfolgen. Die frühere Spezifikation BGB wird entfallen.

Entsprechend sollte in Programmiersystemkommandos ein Magnetband als Wert einer Trägerspezifikation mit B60N (800 bpi, Normalfall) oder B60H (1600 bpi) angegeben werden, also etwa QUELLTRAEGER=B60N(ARBEIT)

1.5 Verhalten bei ungewollter Leitungsunterbrechung am TR440

Durch verschiedene Benutzerfehler und Systemrestarts werden Wähldialogverbindungen zwischen Benutzer und TR440 aufgelöst, ohne das entsprechende Gespräch beendet zu haben. Ein betroffener Benutzer muß sich dann neu anwählen, um sein Gespräch fortzusetzen. Um wieder an sein "altes" Gespräch heranzukommen, muß sich der Benutzer seine Software-Identifikation merken, die nach dem XBG -Kommando ausgegeben wird. Dieser Software-Identifikation ist eindeutig eine Telefonnummer zugeordnet, die der folgenden Tafel entnommen werden kann.

Zuordnung von Softwareidentifikationen zu Telefonnummern an der TR440

Softwareidentifikation	Telefonnummer	Kaskade
F 58-13	7991	
F 59-13	7992	
F 60-13	7993	
F 61-13	7994	
F 62-13	7995	
F 63-13	7961	
F 64-13	7962	
F 65-13	7963	
F 66-13	7964	
F 67-13	7965	
		Kaskade 1
		7991
		Kaskade 2
		7961

Nach dem Wählen dieser Nummern können 2 Situationen auftreten:

- Der Benutzer erreicht sein altes Gespräch wieder. Er kann sein unterbrochenes Gespräch fortsetzen.
- Der Benutzer erhält, bedingt durch die Wählkaskade, einen neuen Anschluß oder das Besetztzeichen: dann hat ein anderer Benutzer den Anschluß des alten Gesprächs belegt (dieser hat laut Regelung durch das RRZE das für ihn fremde Gespräch mit #XEN beendet und ein eigenes begonnen).
Der betroffene Benutzer kann also in dieser Situation sein altes Gespräch nicht fortsetzen, er muß neu beginnen.

1.6 TR440-Kommando RERUNSTART/Stuetzpunktschreiben

Da in letzter Zeit im Zusammenhang mit Stuetzpunktschreiben bzw. - lesen unerklärliche Fehler aufgetreten sind, werden die Benutzer gebeten, das Kommando RERUNSTART und die Subroutine RERUN nicht mehr zu verwenden.

1.7 Protokoll des Benutzerkolloquiums vom 21.2.1978

Die Tagesordnung umfaßte 5 Punkte:

0. Bericht des Leiters des Rechenzentrums
1. Bericht über den Betrieb der zentralen Anlagen
2. Bericht über den Betrieb der Datenstationen
3. Termin
4. Sonstiges

TOP 0. Herr Dr. Wolf berichtete über den derzeitigen Ausbauzustand des Rechenzentrums

Bereits nach einem Jahr Produktion zeigen sich Engpässe. Die beiden Rechner CYBER172 und TR440 laufen z.Z. noch völlig unabhängig voneinander. Ab 1979 soll in einer Ausbauphase mit einem Knotenrechner versucht werden, einen technischen Verbund zu erreichen. Schwierigkeiten ergeben sich dadurch, daß Produkte der Firmen CDC, CGK, Siemens und Dietz gekoppelt werden müssen.

TOP 1. Herr Thomas berichtete über den Betrieb der beiden Rechner
Der Rechenbetrieb an der TR440 war Ende Dezember 1977/Anfang Januar 1978 weitgehend lahmgelegt. Nur durch den massiven Einsatz von Spezialisten und Ersatzteilen konnten die Fehler behoben werden. Probleme gibt es mit der LFD, die nahezu voll ist.

An der CYBER172 waren keine längeren Ausfallzeiten zu verzeichnen. Der post mortem dump wurde verbessert. In der Aufsicht wurde ein Anrufbeantworter unter der Nummer 7039 eingerichtet.

TOP 2. Datenfernverarbeitung

Die Fehlerhäufigkeiten sind gleich geblieben.

TOP 3. Termine

7.3.1978	16,00 Uhr Kolloquium mit Film (Siemens)
17. - 21.4.1978	Konzentrierte Ausbildungswöche
27.6.1978	14,00 Uhr Treffen der Mitarbeiter der Außenstationen
	16,00 Uhr Benutzerkolloquium

TOP 4. Sonstiges

Programmbibliothek wurde erweitert, neue Dokumentation bis Ende März.

Es gibt Schwierigkeiten mit dem Kommando ERZEUGE, Fehler sollen der Aufsicht gemeldet werden.

GRAFIK ist einsatzbereit.

2. Neues zum Stand der Software

2.1 Programmbibliothek (Abel, Cramer)

Die Anwendersoftware des RRZE wurde um einige Programmpakete und Einzelprogramme erweitert, ebenso wurden zu einigen Programmen (SPSS, BMDP, EISPACK, PASCAL, MACLISP) Manuals gekauft. Wegen der Fülle der Programmdokumentation war eine Umorganisation der Programmbibliotheksände erforderlich. Inhaltlich wurden die Bibliotheken nur geringfügig geändert (z.B. Programmbibliotheksgedächtnis am TR440).

Im Anhang finden Sie einen Überblick über die am RRZE installierten Programmpakete, die Installation weiterer hat begonnen bzw. ist geplant (Statistik: CLUSTAN IC, NPAR, ALSCAL; finite Elemente: SAP IV, SAP V, NONSAP, PLOTSAP; Simulation: GASP IV).

2.1.1 Umstellung der TR440-Kommandos der RRZE-Programmbibliothek

Durch die Vergrößerung des Kommando umfanges der RRZE-Programmbibliothek wurde eine Umstellung von der bisherigen globalen Bereitstellung aller Kommandos über das Gedächtnis UNRZPB.GED zu einer selektiven Bereitstellung erforderlich. Der damit eventuell verbundene Mehraufwand wird durch die Vereinheitlichung des Zugriffs auf Kommandos, Dokumente, Systeme und Quellen sowie durch die Möglichkeit der Benutzung von benutzereigenen Gedächtnissen aufgewogen.

Ab sofort werden alle Objekte der Programmbibliothek mit dem Kommando ERZEUGE bereitgestellt. Für Kommandos hat es die Form

ERZEUGE, KOMMANDO.Kommandoname

dabei können mehrere Kommandos gleichzeitig angegeben werden. Abkürzungen sind, soweit eindeutig, erlaubt; so sind die beiden folgenden Kommandos gleichwertig:

ERZEUGE, KOMMANDO.OBJEKTINFORM' KOMMANDO.SAM

und

ERZEUGE, K.OBJ' K.SAM

Die Datei UNRZPB.GED (23.0) vom 7.2.1978 wird ab sofort nicht mehr gewartet.

(Beschreibung der Kommandos siehe Band 61 und 63)

2.1.2 Neue Programmbibliotheken

NAG (Numerical Algorithmus Group)/CYBER

IMSL (International Mathematical & Statistical Libraries)/TR440

NAG und IMSL sind FORTRAN-Unterprogrammbibliotheken, zur Lösung numerischer Probleme. Beide haben sich seit vielen Jahren weltweit bewährt, es wird Wert auf die Stabilität der Algorithmen, die einfache Handhabung der Programme und natürlich auf Fehlerfreiheit gelegt. Beide Bibliotheken wurden für 1 Jahr gemietet, bei Bewährung wird die Miete verlängert.

Die Dokumentation von NAG und IMSL ist in der Beratung, den Benutzerräumen und an den RJE-Stationen verfügbar.

Wir empfehlen unseren Benutzern anstelle der alten Bibliotheken wie z.B. SSP die Verwendung von NAG und IMSL, da diese, wie die Erfahrung an anderen Universitäten zeigt, besser sind.

2.1.3 STARG-TR440 (Band 61)

Folgende Programme und Kommandos der STARG wurden neu am RRZE installiert.

DRDRUCKE:	Ausgabe des Ablaufprotokolls als Teilauftrag
DRLOESCHE:	Löschen des Ablaufprotokolls und Neuerstellen des "Jobnamens".
ERZEUGE:	Bearbeiten einer registerverwalteten Steuerkarten-datei
GRAUDRUCK:	Ausdruck eines Rasters in Grautönen
OBJEKTINFORM:	Information über Operatoren und Montageobjekte
STAGE2:	Makroprozessor von Waite

2.1.4 UNRZPB

Die Bibliothek UNRZPB enthält Programme und Kommandoprozeduren, die keinem geschlossenen Programm paket angehören. Hier werden die Einzelprogramme und Kommandoprozeduren gesammelt, die am RRZE erstellt oder adaptiert werden.

Neu in UNRZPB-TR440 (Band 61):

AOKOPIERE:	Kopieren einer Druckdatei (A-Datei) in eine Texthaltungsdatei (O-Datei)
CATLIST:	Inhaltsverzeichnis beliebiger Träger in alphabetischer Reihenfolge
GENERZDATEI:	Generieren einer Erzeugedatei
PROT:	Abfrage und Änderung des Ablaufprotokollzustands
MCS80:	Simulationsprogramm für INTEL8080
MODNEW:	Optimierung von Fehlerquadratsummen
MUCH:	Multiple choice Analyse

Neu in UNRZPB-CYBER (Band 61):

CATSAVE:	Sichern aller permanenten Files eines Benutzers auf Magnetband
GETPROG:	Kopieren von Quellendecks aus der Programmbibliothek
GET440:	Übernahme von Daten vom TR440 zur CYBER
KC4P:	Ausstanzen von Karten im Kartencode KC4
KC4R:	Einlesen von Karten im Kartencode KC4
LIBCALL:	Aufruf von Programmen von benutzereigenen Bibliotheken
LIBMOD:	Generierung und Wartung von benutzereigenen Bibliotheken
PTREAD:	Einlesen von Lochstreifen im ASCII-Code (8-Kanal)
PTWRITE:	Ausstanzen von Lochstreifen im ASCII-Code (8-Kanal)
SAVE440:	Übernahme von Daten von der CYBER zum TR440

2.1.5 CERN-Hinweis (Band 62)

Zu TIDY (Reinigen von Fortran-Programmen) liegt die ausführliche Beschreibung vor.

2.1.6 STARG-Organisation (Band 63)

Mit dem Informationssystem (SYSTEM.STARG) kann auf die STARG-Dokumentation zugegriffen werden, die STARG-Quellen sind mit ERZEUGE einfacher zugänglich geworden. Welche Programme in der Bibliothek &STARG enthalten sind, erfährt man mit OBJEKTINFORM. Näheres siehe Band 61 und 63.

2.1.7 PASCAL, LISP (Band 66)

Die Dokumentation wurde um PASCAL- User Manual and Report und MACLISP-Manual erweitert.

2.1.8 Statistik (Band 67)

Dieser Band ist neu, hier ist die am RRZE verfügbare Statistik-Software dokumentiert:

BMD/BMDP Biomedical Computer Programs
SPSS Statistical Package for the Social Sciences
SFZ Programmbibliothek des Sozialwissenschaftlichen
 Forschungszentrums Nürnberg

Zu BMDP und SPSS wurden bzw. werden ausführliche Manuals beschafft.

2.1.9 ELAN (Band 73)

ELAN (Education Language) wird neben PASCAL als Programmiersprache empfohlen, die sich besonders für den Programmierunterricht an Schulen eignet.

2.1.10 Editoren (Band 68)

NEDIERE wurde EDIERE als TR440-Texteditor, SAM (TR440-Editor zur Wartung von Programmbibliotheken) wurde schneller gemacht, die Quelldateien erhielten eine andere Struktur.

2.1.11 ALGOL 68 C - TR440

Ab sofort steht am TR440 eine ALGOL68C-Testversion zur Verfügung (ausführliche Beschreibung siehe Anhang)

2.2 Hinweise zur Benutzung von SIMULA

2.2.1 Schnelle Ein-/Ausgabe für Dialogprogramme

Zur Beschleunigung der Ein-/Ausgabe bei Dialogprogrammen in SIMULA wird empfohlen, eine neue Version dieses Teils des Laufzeitpaketes zu verwenden. Diese steht zur Verfügung, wenn vor dem Übersetzen auf LGO (andernfalls ist die betr. Zielfile im folgenden Kommando anstelle von LGO einzusetzen) das Kommando

GET, LGO=NEWSIM/UN=UNRZST.

ausgeführt wird.

Hierdurch wird verhindert, daß das Programm nach jedem Ausgabevor-
gang aus dem Arbeitsspeicher hinausgerollt und nach der Eingabe
wieder hineingerollt wird.

2.2.2 Fehler bei der Dateibearbeitung

Bei der Verwendung sequentieller Dateien in Simula (infile, outfile) unter NOS treten häufig Fehler auf, die mit der Setzung des BL (buffer length) - Parameters in der OPEN-Anweisung im Zusammenhang stehen.

Mit dem BL-Parameter wird der betreffenden Datei ein bestimmter Arbeitsspeicherbereich zugewiesen (Puffer). Häufig sind die Dateien, die von einem SIMULA-Programm bearbeitet werden sollen, umfangreicher als der mit dem BL-Parameter vereinbarte Bereich aufnehmen kann.

In den Puffer wird dann an der durch den BL-Parameter vorgegebenen Grenze eine 'end-of-file'-Marke eingetragen, auf die die endfile-Abfrage im SIMULA-Programm anspricht. Die restlichen Daten auf der Originaldatei werden ignoriert.

Nach den bisher getesteten Fällen ist es möglich, diese Daten dennoch zu bearbeiten. Dazu wird bei Ansprechen der 'endfile'-Abfrage die Datei geschlossen, nicht zurückgespult und erneut geöffnet. Danach kann weiter eingelesen oder ausgegeben werden.

Anweisungsfolge:

```
if inp.endfile then
begin
  inp.close("")"norewind");
  inp.open(blanks(imagelength),b1);
end;
inp.inimage;
```

Dabei ist inp die betreffende sequentielle Datei, b1 und image-length müssen vorher im Programm Werte zugewiesen bekommen haben; im Regelfall sollten hier die Werte des ersten OPEN-Aufrufs eingesetzt werden.

Der Standardwert für BL ist 512 und gibt in etwa die Länge des mit der Datei verbundenen Puffers in Maschinenworten an. Ist b1 kleiner 65, so wird es automatisch auf 512 gesetzt.

Man beachte, daß es normalerweise nicht möglich ist, von vorneherein genau zu sagen, wo die 'end-of-file'-Marke eingetragen wird. Man kann also die Anzahl der Wörter nicht nach der Image-Länge und der Anzahl der Zeilen in der Datei berechnen!

Der Wert von BL kann selbstverständlich bis zur maximal zulässigen 'field length' erhöht werden, so daß eventuell die ganze Originaldatei in den Pufferbereich paßt, ohne daß die CLOSE/OPEN-Operationen durchzuführen sind. Dies bringt aber verlängerte Wartezeiten mit sich und der Aufwand erhöht sich, daher wird dieses Vorgehen bei größeren Dateien nicht empfohlen (über 20000g Worte). Es sei noch darauf hingewiesen, daß bei der vorgeschlagenen Anweisungsfolge natürlich bei Durchlaufen der gesamten Originaldatei der Lauf mit dem Laufzeitfehler "Endfile encountered" abgebrochen wird, wenn versucht wird, hinter der 'end-of-file'-Marke der Originaldatei weiterzulesen, wenn also die Originaldatei tatsächlich abgearbeitet ist. Es empfiehlt sich dazu, sich selbst eine 'end-of-file'-Markierung zu vereinbaren, die dann nach dem inimage-Aufruf abzufragen wäre,

zum Beispiel:

```
if inp. image.strip="END OF FILE" then ...;
```

Bei "random access files" (directfile) wird der BL-Parameter ganz anders eingesetzt. Die Aufteilung der Originaldatei auf verschiedene, nacheinander zu bearbeitende Bereiche entfällt insbesondere dann, wenn die Datei ursprünglich schon von einem SIMULA-Programm erstellt wurde. Es ist nämlich bei 'direct-files' nicht erlaubt, die Image-Länge oder den Wert von BL zu verändern. Beide müssen bei jeder Verwendung der Datei die gleichen Werte haben. Die gesamte Datei muß also sofort und vollständig übernommen werden, d.h. der Wert von BL muß von vorneherein groß genug sein.

Leider sind die genannten Schwierigkeiten nicht auf einfache Weise im SIMULA-Laufzeitsystem zu beheben, da die dortige Ein-/Ausgabe-Verwaltung nichtstandardisierte Schnittstellen zum Betriebssystem benutzt. Da der Compiler unter SCOPE bzw. NOS/BE implementiert wurde, haben wir in Norwegen um eine Anpassung an NOS nachgesucht.

2.3 Neue Grafik-Software (z.Z. nur auf Cyber)

Die für beide Rechenanlagen konzipierte Grafik ist jetzt als erstes auf der CYBER verfügbar.

Sie erhalten eine Kurzbeschreibung durch folgende Steuerkarten:

```
GET,GRAKURZ/UN=UNRZPB,NA.  
COPY,GRAKURZ,OUTPUT.
```

Beschreibung und geplottete Zeichnungen mit der Verschlüsselung der Zeichen und Symbole erhält man mit folgender Steuerkartensequenz:

```
GET,GRABESC/UN=UNRZPL,NA.  
CALL,GRABSC.
```

Der Grafik-File für die Zeichnungen ist durch folgende Steuerkarte erhältlich:

```
GET,GRAFIK/UN=UNRZPB,NA.
```

Hinweis:

Gegenwärtig ist der Plotter-Characterset noch nicht vollständig identisch mit den auf den Druckern ausgegebenen Zeichen. Dies betrifft die Zeichen mit den Nummern 48,52 bis 57 und 60 bis 62; es ist für künftige Versionen vorgesehen, daß die Zeichensätze der Grafik auch an den Drucker angepaßt werden; es empfiehlt sich deshalb, diese Zeichen in neu zu schreibenden Programmen möglichst noch nicht (oder schon in der neuen Bedeutung) zu verwenden.


```
+++++
+          +          R  R  Z  E          +  SOFTWARE-KATALOG  +
+  REGIONALES  +          +          +          +  BEARB.: H.CRAMER  +
+          +          +          +          +  +
+  RECHENZENTRUM  +          PROGRAMMBIBLIOTHEK  +  +
+          +          +          +          +  +
+  ERLANGEN  +          CYBER / TR440  +  DATUM : FEBR.78  +
+          +          +          +          +
+++++
```

AUF DEN FOLGENDEN SEITEN WIRD EIN UEBERBLICK UEBER DIE AM RRZE VERFUEGBARE SOFTWARE GEgeben. DIE PROGRAMMPAKETE SIND NACH DEM ACM-INDEX GEORDNET, EIN VERZEICHNIS DER ALLGEMEINEN MATHEMATISCHEN PROGRAMMBIBLIOTHEKEN (ACM-INDEX:VO) IST IN BAND 60 UND IN DEN UNTER DO ANGEgebenEN BAENDEN ZU FINDEN.

BEDEUTUNG DER SPALTEN :

ACM	KLASSIFIKATIONSCODE (ACM-INDEX)
PROGRAMMNAME	NAME DES PROGRAMMPAKETS ODER EINZELPROGRAMMS
SPR	SPRACHE, IN DER DAS PROGRAMM AUFGERUFEN WIRD BZW. GESCHRIEBEN IST
	FTN: FORTRAN A60: ALGOL60 COB: COBOL KEINE ANGABE: MEHRERE SPRACHEN ODER FUER DEN AUFRUF BEDEUTUNGSLOS
T	TYP DES PROGRAMMS
	H: HAUPTPROGRAMM U: UNTERPROGRAMM KEINE ANGABE: HAUPT- UND UNTERPROGRAMME ODER PROGRAMMSYSTEM
RA	RECHENANLAGE
	C: CYBER T: TR440
BIBLIOTHEK	BIBLIOTHEKSSAMMLUNG, ZU DER DAS PROGRAMM GEHOERT
DO	DOKUMENTATION
	<ZAHL>: BANDNUMMER DER RRZE-DOKUMENTATION BB : RRZE-BENUTZERBERATUNG
KURZBESCHREIBUNG KURZE BESCHREIBUNG DES PAKETS BZW. PROGRAMMS	

#F. MATRIZEN- UND VEKTOROPERATIONEN, LINEARE GLEICHUNGSSYSTEME

F1. MATRIZENOPERATIONEN
FORMAT440 FTN U T FORMAT440 27 UNTERPROGRAMMPAKET FUER MATRIXBERECHNUNGEN

F2. EIGENWERTE UND EIGENVEKTOREN
EISPACK FTN U CT EISPACK 65 EIGENSYSTEM SUBROUTINE PACKAGE

#G. STATISTIK UND WAHRSCHEINLICHKEITSTHEORIE

G0. ALLGEMEINES
BMD FTN H CT BMD 67 BIOMEDICAL COMPUTER PROGRAMS
BMDP FTN H CT BMDP 67 BIOMEDICAL COMPUTER PROGRAMS (P-SERIE)
SFZ FTN H CT SFZ 67 PROG.BIBL. D.SOZIALWISSENSCH.FORSCHUNGSZENTRUMS NUERNBERG
SPSS FTN H T SPSS 67 STATISTICAL PACKAGE FOR THE SOCIAL SCIENCES

#H. OPERATION RESEARCH, SIMULATION UND MANAGEMENT SCIENCE

H4. SIMULATIONS-VERFAHREN
GPSS T SYSTEM 36 SPRACHE ZUR SIMULATION DISKRETER SYSTEME

H6. NETZPLANTECHNIK
BKN T SYSTEM 29 NETZPLANPROGRAMMSYSTEM
PERT C SYSTEM BB PERT/TIME NETZPLANTECHNIK FUER ZEITABHANG. UEBERWACHUNGEN

#J. AUSGABE

J5. GRAFISCHE AUSGABE UEBER DRUCKER ODER SPEZIELLE ZEICHENGERAETE
GRAFIK A60 U CT GRAFIK 69 RRZE-GRAFIK-SOFTWARE
GRAFIK FTN U CT GRAFIK 69 RRZE-GRAFIK-SOFTWARE

#K. INFORMATIONS-TRANSFER

K0.	ALLGEMEIN	C	SYSTEM	41	FILE ORGANIZER AND RECORD MANAGER (DATENMANIPULATION)
	FORM	FTN	U C	41	RECORD MANAGER ROUTINEN FUER FORTRAN
	RM/FORTRAN	COB	U C	43	RECORD MANAGER ROUTINEN FUER COBOL

#L. AUSFUEHRUNGROUTINEN

L 1.	ASSEMBLER	C	SYSTEM	46	CYBER ASSEMBLER
	COMPASS	T	SYSTEM	32	TELEFUNKEN ASSEMBLER
	TAS	H	UNRZPB	61	INTEL 8080 SIMULATION AM TR440
	MCS80				
L 2.	COMPILER	C	SYSTEM	44	CYBER ALGOL 60 -COMPILER
	ALGOL60	T	SYSTEM	24	TR440 ALGOL 60 -COMPILER
	ALGOL60	C	SYSTEM	43	CYBER COBOL 5 -COMPILER
	COBOL5	T	SYSTEM	23	TR440 COBOL COMPILER
	COBOL	C	SYSTEM	41	CYBER FORTRAN -COMPILER
	FORTRAN	C	SYSTEM	41	CYBER FORTRAN -COMPILER MIT POST MORTEM DUMP
	PMDCOMP	T	SYSTEM	21	TR440 FORTRAN -COMPILER
	FORTRAN	T	UNRZPB	73	TR440 ALGOL 68 COMPILER (TESTVERSION)
	ALGOL68	C	SYSTEM	45	A PROGRAMMING LANGUAGE
	APL	C	SYSTEM	45	CYBER BASIC -COMPILER
	BASIC	C	SYSTEM	21	TR440 BASIC -COMPILER
	BASIC	T	SYSTEM	36	PROGRAMMIERSPRACHE ZUR LOESUNG NICHT NUMERISCHER PROBLEME
	BCPL	T	UNRZPB	73	EDUCATION LANGUAGE (PROGRAMMERSPRACHE F .D. SCHULUNTERRICHT)
	ELAN	T	PASCAL	66	PROGRAMMIERSPRACHE VON N. WIRTH
	PASCAL	T	SYSTEM	25	TR440 PL / 1 -COMPILER
	PL / 1	T	SYSTEM	37	TR440 -SYSTEMPROGRAMMIERSPRACHE
	PS440	T	SYSTEM	32	REPORT PROGRAM GENERATOR
	RPG	T	SYSTEM	44	SIMULA COMPILER (NORWEGEN)
	SIMULA	C	SYSTEM	57	CYBER -SYSTEMPROGRAMMIERSPRACHE
	SYMPLE	C	SYSTEM		

L4. PREPROCESSING (VORLAUFPROGRAMME)
PFTP 73 SYNTAXCHECKER FUER PORTABLES FORTRAN
STAGE2 61 MACROPROCESSOR VON WAITE
UPDATE 47 VERWALTEN VON PROGRAMMQUELLEN IM BATCH
SAM 68 VERWALTEN VON PROGRAMMQUELLEN/-BIBLIOTHEKEN IM BATCH

#M. DATENVERARBEITUNG

MO. ALLGEMEINES
EDIT C SYSTEM 42 CYBER TEXTEDITOR
EDIERE T STARG 68 TR440 TEXTEDITOR
EDITOR H T STARG 68 DRUCKAUFBEREITUNG VON TEXTEN

M1. SORTIEREN
SORT C SYSTEM 43 CYBER SORT/MERGE
SORT T SYSTEM 23 TR440 SORT

#R. NICHT-NUMERISCHE MATHEMATIK

R2. SYMBOL-MANIPULATION
REDUCE H T UNRZPB 61 MANIPULATION ALGEBRAISCHER AUSDRUECKE

R3. LISTEN - UND ZEICHENKETEN VERARBEITUNG
LISP T UNRZPB
UTLISP C UNRZPB 66 LISPLISP
66 UTLISP4.0

#S. INFORMATION RETRIEVAL

SO. ALLGEMEINES
TOTAL C SYSTEM 55 CYBER DATENBANKSYSTEM
ATHENA C SYSTEM 55 ABFRAGESYSTEM ZU TOTAL
DBS T SYSTEM 28 TR440 DATENBANKSYSTEM
TELDOCK T SYSTEM 28 TELEFUNKEN DOKUMENTATIONSSYSTEM (DBS-ABFRAGESYSTEM)

ACM PROGRAMMNAME SPR T RA BIBLIOTHEK DO KURZBESCHREIBUNG

#T. ANWENDUNGSPROGRAMME

T4.	TECHNIK	H	T	UNRZPB	72 ANALYTIC NETWORK PROGRAM 3
	ANP3	H	T	UNRZPB	72 NONLINEAR ANALYSIS PROGRAM FOR ELECTRONIC CIRCUITS
T8.	SOZIALWISSENSCHAFT, PAEDAGOGIK UND PSYCHOLOGIE				

T8. PLANIT T SYSTEM 36 PROGRAMMING LANGUAGE FOR INTERACTIVE TEACHING

#V. VERSCHIEDENE KATEGORIEN

VO.	ALLGEMEINES	FTN	C	CERN	62 EUROPAEISCHE ATOMFORSCHUNGSZENTRALE GENF
	CERN	A60	U	T SYSTEM	24 PROGRAMME DER COMPUTER GESELLSCHAFT KONSTANZ
	CGK	FTN	U	T SYSTEM	27 PROGRAMME DER COMPUTER GESELLSCHAFT KONSTANZ
	CGK	FTN	U	C SYSTEM	40 FORTRAN COMMON LIBRARY MATHEMATICAL ROUTINES
	CDC	FTN	U	C SYSTEM	71 INTERNATIONAL MATHEMATICAL&STATISTICAL LIBRARIES
	IMSL	FTN	U	T IMSL	70 NUMERICAL ALGORITHMS GROUP
	NAG	FTN	U	C NAG	64 SCIENTIFIC SUBROUTINE PACKAGE (IBM)
	SSP	FTN	U	CT SSP	63 STAENDIGE ARBEITSGRUPPE DER TR440 RECHENZENTREN
	STARG		T	STARG	61 RRZE-PROGRAMME UND -KOMMANDOPROZEDUREN
	UNRZPB		CT	UNRZPB	

Regionales Rechenzentrum Erlangen	RRZE Programmbibliothek TR44Ø	ALGOL68C Bearb.: Abel/Görz Datum: Febr. 1978
---	-------------------------------------	--

Ab sofort steht an der TR44Ø eine Vorabversion des ALGOL68C-Compilers der Universität Cambridge zur Verfügung.

ALGOL68C weicht in einigen Punkten von revised ALGOL68 ab.

Die Abweichungen sind im Anhang a) beschrieben. Eine genaue Beschreibung findet man in

S.R. Bourne, A.D. Birrell, I.Walker
ALGOL68C Reference Manual

Diese Unterlage ist bestellt und wird in Kürze in der Aufsicht einzusehen sein.

Die derzeit vorhandene Compilerversion ist, wie schon oben erwähnt, eine Vorabversion, deshalb tritt ein erhöhter Bedarf an Kernspeicher und Rechenzeit auf. Für Trivialprogramme mit wenigen Zeilen, z.B.

KSB \sim 60 K

RZS \sim 20 sec.

Außerdem können noch Fehler im Compiler auftreten.

Der Compiler wird mit dem Kommando

GERZEUGE, SYSTEM.ALG68

bereitgestellt. Anschließend kann das Kommando UEBERSETZE mit den im Anhang c) beschriebenen Abweichungen verwendet werden.

Anhang A) Unterschiede zwischen ALGOL68C und ALGOL68

Anhang B) Darstellung von ALGOL68-Programmen am TR44Ø

Anhang C) Sprachspezifische Abweichungen im UEBERSETZE-Kommando

Unterschiede zwischen
ALGOL68C und ALGOL68

Restrictions

- No parallel clauses.
- No flexible names.
- No formatted transput.
- No vacuums.
- Use of indicants.
- Round brackets are not available in row-declarers.
- Colon-symbol must not be present in virtual rowers.
- Both bounds must be specified in an actual rower.
- :=and=: are not available in operators.
- Widening of BITS and BYTES is not provided.
- The form '(declarer):' in conformity cases-clauses is not available.

Extensions

- Labels in enquiry-clauses.
- UPTO and DOWNTO in loop-clauses.
- Until-part in loop-clauses.
- Operator priorities.
- Row-symbol in row-declarers.
- Monadic-formula is a secondary.
- Displacements.
- Assign-formulas.
- Predicates.
- Handles.
- Code-sections.
- Escaped-characters in string-denotations.
- Use of square brackets in calls.
- The f-symbol in conditional-clauses.
- :~=: as a representation of the is-not-symbol.

Darstellung von ALGOL68-Programmen
am TR440

a) Zeichenvorrat:

In Kommentaren und Zeichenketten sind alle TR440-Zeichen zulässig, ansonsten gilt:

- In Bezeichnungen können nur die Buchstaben A...Z, a...z verwendet werden, nicht die Umlaute und ß.
- In real- und compl-Literalen muß statt ₁₀ e oder E verwendet werden.
- Die Kurzform von skip ist ?, nicht ~.
- Eine Kurzdarstellung für nil gibt es nicht.
- ! ist zulässig als Ersatzdarstellung für !.
- Nur die folgenden Sonderzeichen können für Operatoren verwendet werden:

∨ ^ & % + -

⟨ ⟩ / = *

Für die übrigen ALGOL68-Operator-Bezeichnungen gibt es Ersatzdarstellungen:

Report	*	≤	≥	÷	□	L	Γ	⊥	¬	↑	x
TR440	/=	⟨ = ⟩ =	%							**	*
	ne	le	ge	over	elem	lwb	upb	i	not		

b) Darstellung von Wortsymbolen

- Voreingestellt ist „IFIP point stropping“, d.h. alle Wortsymbole werden mit einem Punkt eingeleitet, sie enden beim nächsten Leerkästchen oder Sonderzeichen:

.BEGIN .MODE .R1=.REAL;.R1 R=PI; PRINT (R).END

In Identifikationen sind Leerkästchen zulässig).

- Beginnt der Programmtext mit .UPPER , wird „IFIP upper stropping“ eingestellt, d.h. alle Wortsymbole werden mit großen, alle Identifikatoren mit kleinen Buchstaben dargestellt:

```
BEGIN MODE R1=REAL;R1 r=pi;print(r) END
```

(In Identifikationen sind Leerkästchen zulässig).

c) Darstellung von Zeichenketten (strings) und Zeichen-Literalen

- Für " gibt es keine Ersatzdarstellung; auf Geräten, deren Zeichenvorrat " nicht enthält, gibt man $\text{\texttt{`096}}$ ein.
- Innerhalb von Zeichenketten wird
 - " durch " " und
 - ' durch ' ' dargestellt.

Will man Leerkästchen sichtbar darstellen, kann man 'S oder 's (statt dem im Report angegebenen $\text{\texttt{`}}$) verwenden.

- Paßt eine Zeichenkette nicht in eine Zeile, muß sie in der ersten Zeile mit " abgeschlossen und auf der nächsten wieder mit " begonnen werden. Dazwischen dürfen höchstens Leerzeichen stehen.

Beispiel:

```
print ("ab ""1'S2"
      "3"4")
      ergibt ab "1 23'4
```

Für char-literale gelten die gleichen Regeln.

d) Programmbeispiele

```
.BEGIN
  .PROC NEUE ZEILE=.VOID:PRINT(NEWLINE);
  .OP .FAC = (.INT N).INT:
    (N < 0 ! PRINT ("Fehler!");?
     !:N=0 ! 1
       ! N*.FAC (N-1);
    . FOR I .TO 10 .DO PRINT (.FAC I); NEUE ZEILE .CD
    . END
```

(Statt ! wäre auch | zulässig).

e) Sprachumfang

Gegenüber dem revidierten Report bestehen folgende Abweichungen:

- par, flex, long, short, format
und alles, was dazu gehört, gibt es nicht.
- string ist eine primitive Art, nicht flex [1:0] char.
Einzelne Zeichen von strings erhält man mit n elem s statt s[n]. Alle übrigen string-Operationen sind verfügbar.
- Für Arten, die mit [...] beginnen, gibt es noch keine mode-Deklaration:
mode a = [1:3] real ist nicht möglich (wohl aber mode b = ref[]real).
- Zur Ein- und Ausgabe gibt es derzeit nur die Prozeduren read und print. Die derzeitigen Versionen erwarten ausnahmsweise keine doppelten Klammern:
read (x,y,z);print (pi,"abc",x 3.5, true)
- In Zahlen sind keine Leerzeichen zulässig.
- Operatoren, deren Berechnung := oder =: enthält (z.B. +:=) kann der Benutzer nicht selbst deklarieren.

f) Ein- und Ausgabe von Daten

Derzeit steht ein vorläufiges E/A-System zur Verfügung, das gegenüber dem im Report beschriebenen einige Einschränkungen aufweist. Diese Beschreibung befaßt sich nur mit dem jetzigen Zustand.

Ein völlig neues E/A-System steht vor der Fertigstellung; es umfaßt alle im Report beschriebenen Prozeduren, mit Ausnahme der für formatierte E/A.

Beschreibung der vorläufigen E/A:

- Gleichzeitig können bis zu 10 Dateien bearbeitet werden.
- Jede Bearbeitung erfolgt über einen der beiden folgenden Kanäle:
channel stand in channel = & nur Lesen möglich &,
channel stand out channel = & nur Schreiben möglich &,
channel parameter channel = & nur Lesen möglich & *)
- Für jede Datei muß zunächst eine file-Variable deklariert werden, in der auch das System alles nötige über den Zustand der Dateibearbeitung merkt. Der Identifikator dieser file-Variablen ist beliebig.

*) parameter channel gestattet Zugriff auf *PARAMETER(STARTE)

Beispiel:

```
loc file daten; loc file ergebnis;
```

Zwei file-Variablen sind bereits deklariert:

```
loc file stand in, stand out
```

- Vor dem ersten Lesen/Schreiben muß dem file eine bestimmte Datei und ein passender Kanal zugeordnet werden:

```
open (daten, "DATEI1", stand in channel);
```

```
open (ergebnis, "DATEI2", stand out channel);
```

```
open (steuer, "", parameter channel).
```

Der zweite Parameter von open ist ein string, der die Dateibezeichnung angibt. Hierzu existiert ein Hilfsoperator

```
op datei = (int sgnr)string : # Verweis auf die Datei,  
# die im STARTE-Kommando  
# unter SGNR zugeordnet  
# wurde #
```

Will man bei open keine Dateinamen, sondern symbolische Gerätenummern angeben, erfolgt dies also in Form

```
open (daten, datei 10, stand in channel)
```

stand in und stand out sind bereits eröffnet:

```
open (stand in, datei 5, stand in channel);
```

```
open (stand out, datei 4, stand out channel)
```

Wurde im STARTE-Kommando nichts anderes zugeordnet, bezieht sich also

```
stand in auf DATEN(STARTE) bzw. SGNR=5
```

stand out auf Ablauf- und Konsolprotokoll bzw. SGNR=4
DATEI (STARTE) und DNUMMER (STARTE) werden wie gewohnt ausgewertet.

- Nach dem letzten Zugriff muß die Datei wieder geschlossen werden:

```
close (daten); close (ergebnis).
```

stand in und stand out werden automatisch abgeschlossen.

- Wechsel zwischen Schreiben und Lesen erfordert ein close und ein open mit passendem channel.

- Anmerkung: Wird open nicht mit datei **«sgnr»**, sondern mit einem Dateinamen, z.B. "DATEN", aufgerufen, wird intern eine der Gerätenummern 40,41,... verwendet. Diese Nummern sollten dann bei STARTE nicht anderweitig besetzt sein.

- Schreiben ist grundsätzlich nur mit print möglich. Im Gegensatz zum Report hat print variable Parameterzahl:

```
print (3,a,"abc",newline);
print (x)
```

- Nur Worte der Arten int, real, bool, string, char, bits, bytes, compl, lassen sich ausgeben.

Entgegen dem Report werden int-Zahlen stets auf kleinstmöglichem Raum ohne Leerzeichen ausgegeben.

- print wirkt immer auf stand out. Soll ein anderes file bearbeitet werden, ist wie folgt zu verfahren:

```
loc file stand res := stand out #Stand out - Zustand
                                wird unter Stand res
                                hinterlegt#;
stand out := ergebnis #Zuweisung des gewünschten file
                    auf Stand out#;
print (...) #wirkt auf Stand out, also auf ergebnis#;
stand out := stand res #Wiederherstellung des ursprünglichen Zustands#;
print (...) #wirkt wieder auf Normalausgabe, also
Protokoll.#
```

- Lesen ist nur mit read möglich. Für read und stand in gilt das unter print/stand out beschriebene entsprechend.
- Nur Werte der Arten int, real, bool, string, char, compl lassen sich eingeben.
Im Falle string wird bis zum Zeilenende gelesen, falls nicht make term vorher aufgerufen wurde.
- Die im Report beschriebenen on-Prozeduren stehen fast alle zur Verfügung.

Beispiel:

```
loc file f;
open (f, datei 2, stand out channel);
loc file stand out res := stand out;
loc int i; loc char x;
read (i,newline,x);
stand out :=f;
print (x,newline,i+1);...
```

Sprachspezifische Abweichungen
im UEBERSETZE-Kommando

- 1 QUELLE : Auch SEQ-Dateien sind zulässig
- 2 SPRACHE : ALG68
- 3 NUMERIERUNG: Nur -STD- ist zulässig
- 4 MO : (keine Abweichungen)
- 5 VARIANTE : Die folgenden Angaben werden ausgewertet:
 - DG ("Delimiter groß")
Die Quelle liegt im "IFIP Upper Stropping" vor, z.B. BEGIN REAL y ; read(y) END
 - Im Normalfall ist "IFIP Point Stropping" eingestellt, z.B. .BEGIN .REAL Y ; READ (Y) .END
 - &10 "Cambridge Casestrop"
 - &11 "Cambridge Quotestrop"
(&10 und &11 sollten nur bei alten, von anderen Installationen übernommenen Programmen nach Cambridger Konventionen verwendet werden).
 - &13 Nach der Übersetzung mit MO = <m> bleiben folgende Dateien erhalten:
 - &CODE.<m> : ZCODE-Version des Montageobjektes
 - &TAS.<m> : TAS-Version des Montageobjektes
 - &XREF.<m> : interne Form der Referenzlisten
- 6 PROTOKOLL : A und KW werden derzeit nicht ausgewertet, ansonsten keine Abweichungen.
Bei &13 wird die Quelle nicht protokolliert.
FEHLERMELDUNGEN bitte immer mit PROTOKOLL=0
- 7 DYNKON : Wird erst demnächst ausgewertet.
- 8 TRACE : Wird erst demnächst ausgewertet.
- 9 MV : (Keine Abweichungen)
- 10 KE : Wird nicht ausgewertet.
- 11 TRANSFER : Wird nicht ausgewertet.

12 VERSION : Bei getrennter Übersetzung von Programmteilen wird hier der Name des "umfassenden" Montageobjektes angegeben.

13 BEREICH : Wird zur Zeit nur für das Quellprotokoll ausgewertet.

REGIONALES
RECHENZENTRUM
ERLANGEN

RRZE
PROGRAMMBIBLIOTHEK
TR 440-KOMMANDO

ELAN

ELAN

Übersetzung und Ausführung von ELAN-Programmen

Spezifikation :

1 QUELLE	Quellprogramm
2 PROTOKÖLL	Protokollangabe
3 ZEIT	Rechenzeitschranke für die Programmausführung
4 BEREICH	Bereichsangabe für die Übersetzung
5 TEST	Testparameter für Compilerwartung
6 DATEN	Daten für die Programmausführung

Kommando der RRZE-Programmbibliothek

anlagenspezifische
Voreinstellung :

Einschränkung :

Kommandobereitstellung:

ERZEUGE, KOMMANDO.ELAN

Wirkung :

Mit dem Kommando ELAN können Programme, die in der Schulsprache ELAN abgefaßt sind übersetzt und ausgeführt werden.
Die anschließend aufgeführte Literatur ist in den Ordner der Programmbibliothek enthalten

G. Hommel, S. Jähnichen, K. Kleine, W. Koch
Beschreibung der Schulsprache ELAN
TU Berlin Forschungsprojekt Schulsprache 1976

K. Kleine
Standardpaket der Schulsprache ELAN
TU Berlin Forschungsprojekt Schulsprache

P. Heyderhoff
Einführung in ELAN
GMD Informatik-Kolleg, Bonn, 1976

Bartling, Hahn, Liedke
SLAN3B-Benutzerhandbuch
RU Universität Bielefeld

R. Hahn, K. Leweling
Einführung in die Programmiersprache SLAN/ELAN
RZ Universität Bielefeld

Bemerkungen:

1. Vorübersetzen

Mit Hilfe der Compiler-Option INSERT können Pakete vorübersetzt werden. Sie stehen dann bei weiteren Übersetzungen im gleichen Auftrag zur Verfügung.

2. Vorübersetzen und fest eintragen

a) Erstellen einer eigenen E&&VSI-Datei

KOPIERE, DATEI=E&&VSI, QUELLE=LFD(&ELAN&)
, ZIEL=LFD(elan user)

Von jetzt an muß natürlich stets die eigene VSI-Datei anstelle der System-VSI angemeldet werden mit

LFABMELDE, E&&VSI (Abmelden der System VSI)
 LFANMELDE, SCHREIBEN=elan user.E&&VSI

b) Eintragen

Die Pakete müssen wie in (1.) beschrieben vorübersetzt werden. Mit Hilfe der Kommandos

LFANMELDE, SCHREIBEN=elan user.E&&VSI
 STARTE, ELANSAVE

wird der augenblickliche Zustand des virtuellen Tabellenspeichers in die VSI-Datei geschrieben, so daß die vorübersetzten Pakete auch in späteren Aufträgen zur Verfügung stehen.

1 QUELLE

QUELLE

ELAN-Quellprogramm

Spezifikation :

Dateiname: Quelle liegt in einer Datei vor

/Fremdstring: Quelle liegt auf Fremdstringposition vor

obligate Spezifikation

anlagenspezifische
Voreinstellung : -

Einschränkung :

Wirkung :

Durch die Spezifikation QUELLE wird das zu rechnende ELAN-Programm
gekennzeichnet.

Datum :

PROTOKOLL

Spezifikation :

- : kein Protokoll
- STD- : Standard-Protokoll
- KO : zusätzliches Protokoll auf der Konsole

optionale Spezifikation

anlagenspezifische
Voreinstellung : -

Einschränkung :

Wirkung :

3 ZEIT

ZEIT

Rechenzeitschranke für Programmausführung

Spezifikation :

n: Zeitangabe in Sekunden

optionale Spezifikation

anlagenspezifische
Voreinstellung : 3

Einschränkung :

Wirkung :

Angabe einer Rechenzeitschranke für die Ausführung des
ELAN-Programmes

Datum :

4 BEREICH

BEREICH

Spezifikation :

- : Übersetzung der gesamten Quelle

n-m : Der Zeilenbereich n-m der Quelle wird übersetzt

optionale Spezifikation

anlagenspezifische
Voreinstellung : -

Einschränkung :

Wirkung :

TEST

Testparameter für Compilerwartung

Spezifikation :

- : kein Test
4711 : Test

optionale Spezifikation

anlagenspezifische
Voreinstellung : -

Einschränkung :

Wirkung : Compiler Wartung

Bei vermuteten Compiler-Fehlern kann für die Diagnose und Korrektur im HRZ Bielefeld notwendige Information auf zwei verschiedene Weisen beschafft werden:

1. TEST-Parameter im ELAN-Kommando:

#elean,quelle,test=4711

bewirkt, daß bei Alarmen und SSR-Fehlern während der Compilation ein Binärdump ausgegeben wird.

2. Compiler Optionen (in ELAN-Programmen):

COMPILER DUMP OF PASS 1

bewirkt Dump der Hash-Namens-Symboltabelle und des Zwischenstrings.

SYMBOL TABLE DUMP

bewirkt Dump der Symboltabelle.

COMPILER DUMP OF PASS 2

bewirkt Druck des von Pass 2 geparsten Zwischenstring und des erzeugten Objekt-Codes.

Im allgemeinen sind im Fehlerfall meistens nur der COMPILER DUMP OF PASS 2 und bei unerklärlichen Laufzeitfehlern oder Compilationsabbruch der Binärdump notwendig.

6 DATEN

DATEN

Daten für Programmausführung

Spezifikation :

/Fremdstring: Daten liegen als Fremdstring vor

- : Dateneingabe im Gespräch über die Konsole

optionale Spezifikation

anlagenspezifische
Voreinstellung :

Einschränkung :

Wirkung :

An das
 Regionale Rechenzentrum
 der Universität
 Erlangen-Nürnberg
 Martensstr. 1
 8520 Erlangen

EINFÜHRUNG IN DIE BENUTZUNG DER RECHNER
 TR440 UND CYBER

Raum 2.03/Rechenzentrum

		Personen- zahl
Mo., 17.4.	Einführung in die Benutzung des TR440 Steuersprache des TR440 (Fortsetzung vom Vormittag)	9 - 12 Uhr <input type="radio"/> 14 - 17 Uhr <input type="radio"/>
Di., 18.4.	Dialogmöglichkeiten am TR440 Spezielle Themen für den TR440 z.B. Programmpakete: IMSL, Programm- bibliothek besondere Wünsche:.....	9 - 12 Uhr <input type="radio"/> 14 - 17 Uhr <input type="radio"/>
Mi., 19.4.	Einführung in die Benutzung der CYBER Steuersprache der CYBER (Fortsetzung vom Vormittag)	9 - 12 Uhr <input type="radio"/> 14 - 17 Uhr <input type="radio"/>
Do., 20.4.	Dialogmöglichkeiten an der CYBER Spezielle Themen für die CYBER z.B. Lader (Segmentierung), Update, Programmbibliothek besondere Wünsche:.....	9 - 12 Uhr <input type="radio"/> 14 - 17 Uhr <input type="radio"/>
Fr., 21.4.	Einführung in die Benutzung des Grafik-Systems	9 - 12 Uhr und 14 - 17 Uhr <input type="radio"/>

Bei allen Themen dieser Ausbildungswoche wird Gelegenheit zu praktischen Übungen gegeben. Kreuzen Sie bitte die für Sie interessanten Themen an und schreiben Sie dahinter, wieviel Personen aus Ihrem Institut zu dieser Veranstaltung kommen werden.

Für die Termine Dienstag nachmittag und Donnerstag nachmittag geben Sie uns bitte besondere Wünsche an, die wir (notfalls mit Vereinbarung eines späteren Termins) auf jeden Fall berücksichtigen wollen.

Wir erwarten Ihre Anmeldung bis zum 31. März 1978.

An das
 Regionale Rechenzentrum
 der Universität
 Erlangen-Nürnberg
 Martensstr. 1
 8520 Erlangen

**EINFÜHRUNG IN DIE BENUTZUNG DER RECHNER
 TR440 UND CYBER**

Raum 2.03/Rechenzentrum

		Personen- zahl
Mo., 17.4.	Einführung in die Benutzung des TR440 Steuersprache des TR440 (Fortsetzung vom Vormittag)	9 - 12 Uhr <input type="radio"/> 14 - 17 Uhr <input type="radio"/>
Di., 18.4.	Dialogmöglichkeiten am TR440 Spezielle Themen für den TR440 z.B. Programmpakete: IMSL, Programm- bibliothek besondere Wünsche:.....	9 - 12 Uhr <input type="radio"/> 14 - 17 Uhr <input type="radio"/>
Mi., 19.4.	Einführung in die Benutzung der CYBER Steuersprache der CYBER (Fortsetzung vom Vormittag)	9 - 12 Uhr <input type="radio"/> 14 - 17 Uhr <input type="radio"/>
Do., 20.4.	Dialogmöglichkeiten an der CYBER Spezielle Themen für die CYBER z.B. Lader (Segmentierung), Update, Programmbibliothek besondere Wünsche:.....	9 - 12 Uhr <input type="radio"/> 14 - 17 Uhr <input type="radio"/>
Fr., 21.4.	Einführung in die Benutzung des Grafik-Systems	9 - 12 Uhr und 14 - 17 Uhr <input type="radio"/>

Bei allen Themen dieser Ausbildungswoche wird Gelegenheit zu praktischen Übungen gegeben. Kreuzen Sie bitte die für Sie interessanten Themen an und schreiben Sie dahinter, wieviel Personen aus Ihrem Institut zu dieser Veranstaltung kommen werden.

Für die Termine Dienstag nachmittag und Donnerstag nachmittag geben Sie uns bitte besondere Wünsche an, die wir (notfalls mit Vereinbarung eines späteren Termins) auf jeden Fall berücksichtigen wollen.

Wir erwarten Ihre Anmeldung bis zum 31. März 1978.

