

HEISSER DRAHT

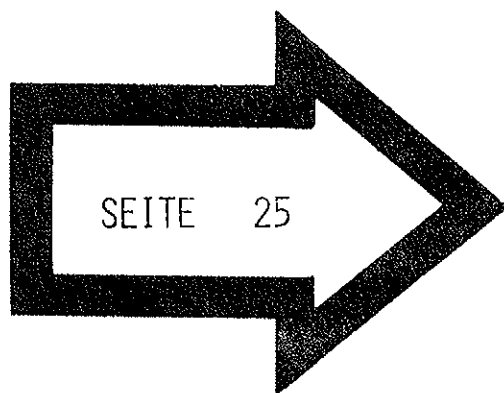
RECHENZENTREN
UNI-TH WIEN
CYBER 73-74

NUMMER 12

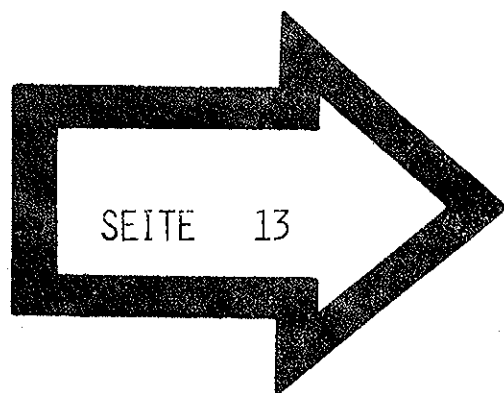
MAI 1975



SCHWERPUNKTE

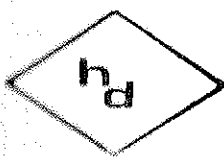


DIE PROGRAMMIERSPRACHE SIGMA



DAS NEUE BETRIEBSSYSTEM SCOPE 3.4.3

*Eigentümer, Herausgeber und Verleger:
Rechenzentren UNI - TH Wien, Cyber 73-74.
Für UNI-Wien: Universitätsstrasse 7, 1010 Wien
Für TH-Wien: Gußhausstraße 27-29, 1040 Wien
Für den Inhalt verantwortlich: Manfred Weiss.
Druck: Österr. Hochschülerschaft der TH-Wien.*



EINGABE

Aufgrund von Fehlern in der letzten Nummer des HEISZEN DRAHTES
möchten wir die Betriebszeiten wiederholen:

BETRIEBSZEITEN DES INTERFAKULTÄREN RECHENZENTRUMS DER UNIVERSITÄT WIEN

ÖFFNUNGSZEITEN MO - FR: 07.00 - 20.00 Uhr
(Locherraum, Jobannahme, SA: 07.00 - 14.00 Uhr
Jobausgabe)

BATCH-BETRIEB MO, MI, FR: 08.00 - 22.00 Uhr
 DI, DO: 11.00 - 22.00 Uhr
 SA: 08.00 - 15.00 Uhr

INTERCOM-BETRIEB MO - FR: 12.00 - 21.00 Uhr

Fallweise werden am Freitag von 07.00 - 11.00 Uhr System-
arbeiten durchgeführt. In diesem Falle entfällt der Produktions-
betrieb.

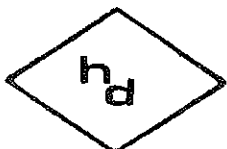
PROGRAMMBERATUNG:

MO - FR: 09.30 - 12.00 Uhr
 14.00 - 17.00 Uhr.

SEKRETARIAT:

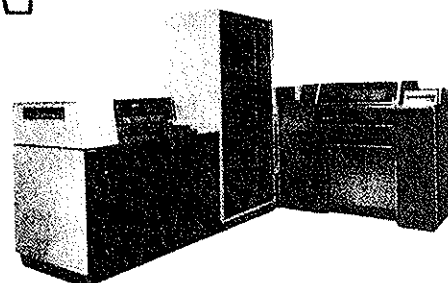
MO - FR: 09.00 - 12.00 Uhr

AUSGABE



MAI 1975

DATENSTATIONEN



D A T E N S T A T I O N CD-1700

Für alle Kunden, die es noch nicht wissen:

Die Datenstation 1700 befindet sich im Hauptgebäude am Karlsplatz, und zwar in den ehemaligen Räumen des IBM-7040 Rechners.

Die Datenstation 1700 ist besonders für jene Kunden von Vorteil, die ihren Arbeitsplatz im Hauptgebäude haben.

Die Datenstation 1700 steht aber natürlich auch allen anderen Kunden zur Verfügung.

Die Datenstation 1700 wird zur Zeit nur von wenigen Kunden benutzt, obwohl die Turn-around-Zeiten wesentlich kürzer sind als beim Großrechner CYBER 74.

Die Datenstation 1700 wird derzeit hauptsächlich von Informatik- und Einführungspraktikanten benutzt. Die Kapazität der Praktikajobs hat sich gegenüber dem Wintersemester 1974/75 um 50 Prozent verringert. Die Einführungspraktikanten haben nur an bestimmten, festgesetzten Tagen Übungsbetrieb.

Die Datenstation 1700 bietet genügend Platz zum Programmieren (Bibliothek) und Lochen (Locherraum). Den Kunden stehen drei 026- u. zwei 029- Locher zur Verfügung. Ein dritter 029- Locher wird in absehbarer Zeit installiert.

Die Datenstation 1700 bietet weiters die Möglichkeit der Aufbewahrung von Kartenpaketen, da in den beiden am Gang stehenden Lochkartenkästen ausreichend Platz vorhanden ist. Diesbezügliche Wünsche leiten Sie bitte an den diensthabenden Operator weiter.

BENUTZEN UND GENIEßEN DAHER AUCH SIE DIE VORTEILE DER DATENSTATION!

ROZA

MAI 1975



DATENSTATIONEN

AUTOMATISCHES DISPOSE AN DER CYBER 73

Da längere Ausdrücke an der Datenstation die Rückgabe kürzerer Jobs empfindlich verzögern, wurden im Betriebssystem Maßnahmen gesetzt, die Jobs mit einem Ausdruck von mehr als ca 50 Seiten automatisch in die DISPOSE-Klasse 01 versetzen.

Solche Jobs werden in der Regel parallel zum Datenstationsbetrieb am Zentraldrucker ausgedruckt und mehrmals täglich vom Operator im Datenstationsraum wie üblich abgelegt.

GRAFENDORFER



MAI 1975



PAPIERSPARMASSNAHMEN AN DER CYBER 73

Um die Papierkosten zu senken, wurde an der CYBER 73 generell auf eine Schreibdichte von 8 Zeilen/Zoll (bisher 6 Zeilen/Zoll) umgestellt, sowie die Seitenvorschübe beim FORTRAN-Compiler-Output eliminiert. Selbstverständlich kann in Sonderfällen der Seitenvorschub reaktiviert werden, sowie eine Schreibdichte von 6 Zeilen/Zoll durch Formatsteuerung erreicht werden. Nähere Auskünfte erteilt die Programmberatung.

Wir bitten um Verständnis für diese Maßnahme, die uns jährlich ca S 200.000.- Kosteneinsparung bringt.

GRAFENDORFER



MAI 1975

Die neue EOF Karte hat den Vorteil, daß beide linke Ecken abgeschnitten sind. Beschädigungen sind daher nicht so häufig. Das Fehlen des runden Abschnittes auf der linken Seite haben wir durch zwei Ecken auf der rechten Seite der EOF Karte ersetzt. Es ist dadurch nach wie vor möglich, den Jobstapel in der Lochkartenlade leicht zu finden.

ROZA



MAI 1975

UP TO DATE UP T

GILT NUR FUER CYBER 74

B E T R I E B S M I T T E L Z U T E I L U N G

AB 1975 04 14 WERDEN ALLE BESTAETIGUNGEN UEBER
BETRIEBSMITTELZUTEILUNGEN AN DEN VERANTWORTLICHEN
BEARBEITER (TH-BENUTZER) BEZIEHUNGSWEISE AN DAS
ZUSTAENDIGE INTERFAKULTÄRE RECHENZENTRUM GESCHICKT,

DIE BESTAETIGUNGEN SIND ALSO NICHT MEHR, WIE BIS-
HER, IN DER PROGRAMMBERATUNG ABZUHOLEN, SONDERN
LIEGEN DANN IM JEWEILIGEN INSTITUT AUF,

GROHMANN

DATE UP TO DA



MAI 1975

FRAGEN AN RZ

FRAGE AN RZ-TH: Wie kann man Informationen von Magnetband für kurzfristige Arbeiten in Time-Sharing-Betrieb auf permanente Files legen, wenn Magnetband-Jobs erst am Abend, nach Ende des Time-Sharing-Betriebs ausgeführt werden?

ANTWORT: Eine Regelung, daß Sonderjobs, die Daten von Magnetband auf permanente Files legen, vor Betriebsbeginn durchgeführt werden, ist in Vorbereitung.

FRAGE AN RZ-TH: Das 'Benutzerhandbuch' ist auf einzelnen Seiten unlogisch numeriert.

ANTWORT: In Kürze wird ein eigenes Benutzerhandbuch (Stand April 1975) herausgegeben, das nach einem verbesserten System numeriert ist. Wir werden uns in Zukunft bemühen, daß die weiteren Seiten des Benutzerhandbuches einfach und eindeutig zuzuordnen sind.

FRAGE AN RZ-TH: Im ALGOL-Manual Appendix C-1, Table 1 ist angegeben, daß alle ALGOL-Symbole als Lochkombinationen darstellbar sind. Wo sind diese veröffentlicht?

ANTWORT: In den Unterlagen für den Kurs RD1-Einführung in das Programmieren ist eine Tabelle mit allen ALGOL-Symbolen und den entsprechenden Lochkombinationen, sowohl im Code 026 als auch 029. Über das Druckbild der Schnelldrucker berichten wir bei Bedarf (HD Okt. 74).



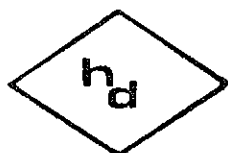
FRAGE AN RZ-TH: Ist es möglich, ALGOL-Programme, die auf anderen Rechenzentren gelocht sind und bei denen die Lochung der Wortsymbolbegrenzung (Apostroph) mit der an der CYBER 7⁴ nicht übereinstimmt, auf der CYBER 7⁴ zu verwenden?

ANTWORT: Es existiert in der Bibliothek THMISC ein Programm, das die Umcodierung des Apostroph durchführt (entspricht Lochcode (8-5) → (8-7)).
Programmbeschreibung MIX269, Nr. MSM001

BEISPIEL:

```
job,CM60000.  
ACCOUNT,xxxxxxxxx.  
ATTACH,THMISC.  
SLOAD,THMISC,MIX269.  
EXECUTE.  
ALGOL,I=COMP.  
LGO.  
eor  
    Programm  
eor  
    Daten  
eof
```

GROHMANN



FRAGE AN RZ zum Kartenleser-Problem an der UNI:

ANTWORT: Beim Kartenleser im Datenstationsraum wurde die Leseinheit erneuert (Phototransistoren anstelle von Photodioden). Seither ist das "Problem der verschwundenen Karten" gelöst. Wir danken den Benützern für die Mithilfe bei der Eingrenzung des Fehlers. Beim Kartenleser im Systemraum wird dieselbe Änderung durchgeführt.

BESCHWERDEN über Funktionsuntüchtigkeit der Kartenlocher:

ANTWORT: Die Kartenlocher werden jeden Tag in der Früh durch unser Personal überprüft und bei Mängel sofort der Technische Dienst verständigt. Ein relativ großer Teil der Fehler ist jedoch auf unsachgemäße Bedienung dieser Geräte zurückzuführen. So wird z.B. beim Herausnehmen der Programmtrommel häufig vergessen, die Kontakthebel anzuheben, wodurch diese Hebel und die Sternrädchen oft zerstört werden. Wir bitten deshalb die Benutzer, die Benützungsvorschriften genau zu beachten.

DATENSTATION an die CYBER 74

Diese Datenstation wurde uns von der CDC freundlicherweise vorübergehend kostenlos zur Verfügung gestellt. Es ist uns bekannt, daß mit dieser Datenstation viele Probleme auftreten. Wir sind aber bemüht, diese Datenstation funktionstüchtig zu halten.

Wir hoffen, daß das vordringlichste dieser Probleme, das Verschwinden einzelner über diese Datenstation eingelesener Jobs, in Scope 3.4.3 behoben sein wird.

GRAFENDORFER



MAI 1975

SOFTWARE

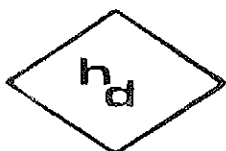
DAS NEUE BETRIEBSSYSTEM SCOPE 3.4.3

Die Software für die CYBER-Anlagen (Betriebssystem, Time-Sharing-System, Compiler) wird laufend verbessert und weiterentwickelt. Die Weitergabe der Änderungen von der Firma CDC an die Rechenzentren erfolgt in verschiedenen Stufen.

Als unterste Stufe (PSR-Level) liefert CDC etwa einmal im Monat Korrekturinformation über kleine, isolierte Fehler.

Die nächste Stufe ist die der neuen "Releases". Sie enthalten grundsätzliche Verbesserungen, die große Teile einzelner Software-Produkte betreffen und mit anderen Software-Produkten abgestimmt werden müssen. Solche Neufassungen enthalten naturgemäß Fehler und sind zumindest in der Anfangsphase meist wesentlich instabiler als die bereits laufenden Versionen, aus denen im Laufe ihrer Einsatzzeit gefundene Fehler bereits entfernt wurden. Diese Situation besteht weltweit und bei allen Herstellern: Die Einsatzplanung neuer Software-Systeme ist ein Optimierungsproblem zwischen den Vorteilen der neuen Möglichkeiten und den Nachteilen der neuen Fehler.

Vor einiger Zeit wurde von CDC die Release SCOPE 3.4.3 freigegeben und von den Rechenzentren der UNI und TH soweit wie möglich getestet. Die dabei gemachten Erfahrungen waren durchaus positiv. Im Sinne einer benutzerorientierten Einsatzplanung wurde am 15. April 1975 ein die Benutzer einbeziehender umfassender Test in der neuen Version durchgeführt. Nach positivem Verlauf dieses Tests bzw. nach daraus resultierenden Verbesserungen ist ein Einsatz von SCOPE 3.4.3 anstelle der alten Release 3.4.1 geplant.



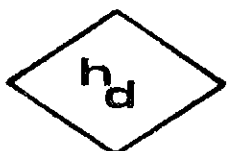
Die folgenden Beiträge beschreiben die wichtigsten, für den Benutzer bedeutungsvollen Neuerungen; genaue Unterlagen sind in Vorbereitung.

SCOPE 3.4.3

Wie die meisten Benutzer schon aus dem SYSBULL wissen, gibt es seit Ostern die neue Release 3.4.3 unseres Betriebssystems SCOPE. Sowohl die internen Tests als auch der große Benutzertest am 15. April zeigten sehr gute Ergebnisse.

Eine neue Betriebssystemrelease bringt immer die Behebung von alten Fehlern und/oder neue Features der Soft- oder Hardware. In der Tat hat L.A.Belady, ein Theoretiker auf dem Gebiet der Betriebssysteme, der auch an der Entwicklung von OS-VS mitgearbeitet hat, eine allgemeine Gesetzmäßigkeit in den Betriebssystemreleases festgestellt. Mit großer Regelmäßigkeit wechseln solche Releases, die neue Features bringen, aber instabil und fehlerhaft sind, mit solchen, die nur die Fehler der vorhergehenden Release beheben, und daher große Stabilität aufweisen. Die etwa eineinhalbjährige Erfahrung hat gezeigt, daß diese Zusammenhänge auch für unser SCOPE gelten, und daß die Releases SCOPE 3.4.x mit ungeradem x stabil und brauchbar sind.

SCOPE 3.4.2 brachte eine völlige Neukonzeption der Queues zusammen mit der Möglichkeit, mit dem ASCII 95-Zeichensatz zu drucken und zu stanzen. Dem Benutzer wird diese Möglichkeit mit der ROUTE-Steuerkarte gegeben, die eine Verallgemeinerung der DISPOSE-Karte und des BATCH-Commands darstellt. Analog dazu enthielt INTERCOM 4.2 die Verwendbarkeit des ASCII 64-, ASCII 95-, ASCII 256-Zeichensatzes im interaktiven Betrieb.

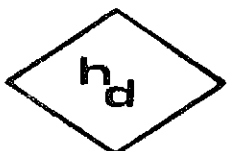


Schon die ersten Versuche mit SCOPE 3.4.2 im vergangenen Sommer zeigten unbefriedigende Ergebnisse. Es wurde daher entschieden, SCOPE 3.4.2 nicht zu installieren, dafür aber SCOPE 3.4.1 und die enthaltenen Produkte auf den jeweils höchstmöglichen PSR-Level zu bringen. SCOPE 3.4.3 enthält in konsolidierter Form die zahlreichen Neuerungen von SCOPE 3.4.2.

INTERCOM 4.3

Auch für INTERCOM gilt das schon für SCOPE gesagte, nämlich, daß die Release 4.3 die in 4.2 neuen Features in konsolidierter Form bringt. Das sind im einzelnen:

1. Die Wahl zwischen den ASCII 64-, ASCII 95- und ASCII 256-Zeichensätzen.
Das bedeutet, daß Benutzer, die intelligente Terminals besitzen, die Möglichkeiten dieser Terminals ausnutzen können; speziell sind das im ASCII 256 (transparenter Code) die Cursorprogrammierung und Steuerung von angeschlossenen Geräten (z.B. Plotter). Im nächsten Heißen Draht wird ein ausführlicher Artikel darüber erscheinen. Jetzt sei bloß gesagt, daß Ein- und Ausgabe im ASCII 256 nur von eigens dafür geschriebenen interaktiven Programmen erfolgen kann, im Command- und Editormode aber nach wie vor nur ASCII 64 verwendet werden kann.



2. Der SCREEN-Command.

Er ermöglicht die Abstimmung des Outputs auf die individuelle Bildschirmgröße.

Für Teletypes:

SCREEN,w	Defaultwert
w Zeilenlänge	72

Für 200UT's:

SCREEN, w,l	
w Zeilenlänge	50
l Zeilenanzahl	20

3. Die Unterscheidung zwischen Batch- und interaktiven Terminals.

Interaktive Terminals geben so wie bisher LOGIN, können je nach Konfiguration Batch- und interaktive Commands geben; sie haben INTERCOM-ID's beginnend mit B und aufwärts. Teletype-kompatible Terminals können nur als interaktive Terminals betrieben werden.

Batch-Terminals brauchen kein LOGIN geben, können aber nur BATCH-Commands verwenden; sie haben INTERCOM-ID's beginnend mit A.

4. Spezielle Eingabezeichenfolgen:

a) Programmabbruch: % A CR

Bei laufendem Output muß dieser zuerst mit CTRL-Z oder ESC unterbrochen werden.

b) interaktive Eingabe von End-of-File (End-of-Partition)

% E O F CR

c) interaktive Eingabe von End-of-Record (End-of-Section)

% E O R CR



DIE ROUTE-KARTE

SYNTAX:

ROUTE(lfn,DC=xx,DEF,EC=ec,FC=ww,FID=*ffffff,IC=ic,TID=yy)

<u>Parameter</u>	<u>Semantik</u>	<u>Default</u>
lfn	Das in eine Queue zu sendende File	-
DC=xx	Disposition-Code SC Scratch IN Input-Queue PR Output-Queue, bel. Drucker P2 Output-Queue, 512 Drucker PU Punch-Queue	SC
DEF	Deferred (d.h.am Jobende)	kein DEF
EC=ec	Externe Charakteristik Printfiles: A6 Central-Site u.alle ASCII-Term. B6 BCD200UT A9 ASCII 95-Zeichensatz f. 711,714 Punchfiles: 026 } Lochcode SB SCOPE 029 } 80COL 80-spaltig } binär ASCII }	A6 029
FC=ww	Formularcode (wie auf DISPOSE)	kein FC
FID=*ffffff	Externer Filename (Header) * kreiere 2 neue eindeut. Zeichen ffffff die ersten fünf Zeichen	Jobname
IC=ic	Interne Charakteristik DIS Display-Code ASCII ASCII-Code BIN Binär	DIS
TID=yy	Terminal-ID Intercom-ID oder C f. Central-Site	Ursprungs-ID des Jobs

Default-Werte für bestimmte lfn's:

	DC	EC	IC
OUTPUT	PR	A6	DIS
PUNCH	PU	029	DIS
PUNCHB	PU	SB	BIN
P80C	PU	80COL	BIN

alle übrigen lfn's haben DC=SC.

MASTAL



DIE NEUE VERSION DES FTN-COMPILERS

Mit dem neuen Betriebssystem SCOPE 3.4.3 wird auch ein neuer FTN-Compiler Version 4.3 installiert. Neben Fehlerbehebungen sind im neuen Compiler einige neue Möglichkeiten eingebaut:

1) CONNECT

In der CALL CONNECT-Anweisung kann durch einen 2. Parameter der Zeichensatz gewählt werden (Display, ASCII-95, ASCII-256).

2) SORT-MERGE-SYSTEM

Das SORTMERGE-System kann von FORTRAN aus durch Unterprogramme direkt aufgerufen werden.

3) RECORD MANAGER

Die Erweiterungen des CRM (Multiple Index Files, READ-ONLY IS) stehen den Benutzern auch in FORTRAN zur Verfügung.

4) FTN-Karte

Es sind nunmehr die neuen Formen der Parameter L und B möglich (wie in den letzten beiden Nummern des HD beschrieben).

5) NAMELIST, FTN-LIBRARY

Im Namelist und in vielen Unterprogrammen wurden Verbesserungen durchgeführt. Vor allem bei den Mathematischen Funktionen wurde die Überprüfung der Argumente weiter ausgebaut.

6) PAUSE

Die Pause-Anweisung kann (soweit sinnvoll) wieder verwendet werden.

7) Papiereinsparung

Die Seitenvorschübe zwischen Programmeinheit, Reference-map und Fehlermeldungen entfallen.



8) Auswertung arithmetischer Ausdrücke

In Übereinstimmung mit dem ANSI-FORTRAN wird zugunsten einer besseren Optimierung darauf verzichtet, die Auswertungsreihenfolge bei den kommutativen und assoziativen Operatoren (*,+) festzulegen.

$A*B*C$

kann daher nun als

$(A*B)*C$ wie bisher oder als
 $A*(B*C)$ oder $(A*C)*B$ oder $A*(C*B)$
ausgewertet werden.

Wird auf eine besondere Auswertungsreihenfolge Wert gelegt, so muß geklammert werden.

COBOL

In Cobol wurden hauptsächlich Fehler behoben und keine neuen Möglichkeiten eingebaut. Besonders wichtig ist, daß nun das Datum in der Schreibweise nach ÖNORM geliefert wird.

Die Erweiterungen des CRM:Multiple Index Files (RECORD KEY IS... und ALTERNATE KEY IS ...), READ-ONLY IS stehen den Benutzern auch in COBOL zur Verfügung.

SCHMITT

ALGOL

In der unter SCOPE 3.4.3 laufenden ALGOL-4-Version (PSR-Level 393) wurde folgende Erweiterung vorgenommen.

Die Fehlerbehandlung erstreckt sich während der Laufzeit auf alle Errormodes, d.h. bei Auftreten eines fatalen Laufzeitfehlers wird das Programm abgebrochen, alle Puffer geleert und ein normales Ende (Endrun) simuliert. Außerdem wird eine entsprechende Dayfile-Nachricht gegeben.

BRAUN



UP TO DATE UP T

PL / I

IM RECHENZENTRUM TH WERDEN DERZEIT UNTERSUCHUNGEN DURCHGEFUHRT, DURCH DIE FESTGESTELLT WERDEN SOLL, WELCHE ANFORDERUNGEN AN EINEN ZUKÜNFTIGEN PL / I - COMPILER AN DER CYBER 74 GESTELLT WERDEN SOLLTEN. UM WIRKLICHKEITSNÄHE TESTKRITERIEN AUFSTELLEN ZU KÖNNEN, ERSUCHEN WIR POTENTIELLE BENUTZER (MIT BEREITS VORHANDENEN PROGRAMMEN) SICH IM RZ-TH BEI HERRN SCHMITT ZU MELDEN.

SCHMITT

UPDATE UP TO DA



MAI 1975

SOFTWARE

SELCOM AN DER CYBER 74

Das Programmsystem SELCOM ist ein in Fortran geschriebenes Programmsystem zur statistischen Analyse von Daten. Seine Hauptfunktion besteht im Auswählen von Daten, Auszählen von Häufigkeitsverteilungen und Drucken von diesbezüglichen Tabellen. Typisches Anwendungsgebiet ist die Auswertung von Fragebögen. Mit diesem Anwendungsspektrum ist SELCOM eine willkommene Ergänzung zu Programmen wie SPSS.

Die wesentlichen Eigenschaften von SELCOM sind:

a) Einfache Verwendbarkeit

Die Auswertungswünsche werden in einer leicht verständlichen Sprache formuliert, die keine sonstigen Programmierkenntnisse voraussetzt. SELCOM führt eine sorgfältige Kontrolle aller möglichen Fehler durch.

b) Optimale Ausnutzung des Computers

SELCOM ist sowohl hinsichtlich Kernspeicher als auch hinsichtlich Rechenzeit außerordentlich sparsam. Da SELCOM grundsätzlich alle Speicherorganisation dynamisch und mit Paging durchführt, gibt es für den Benutzer keine Fragen der Kapazität von SELCOM.

c) Anschlußmöglichkeiten für benutzerspezifische Programme

Für die Ergänzung durch eigene, individuelle Programme sieht SELCOM ein bequemes Interface vor, sodaß auch spezifischen Auswertungswünschen außerhalb der Standardmöglichkeiten Rechnung getragen wird.

SELCOM wurde im Juli/August 1972 für die IBM/360-44 des Rechenzentrums der Universität geschrieben und wurde im Jänner 1975 auf



das Betriebssystem SCOPE 3.4 der CDC-CYBER 73 umgestellt.
Seit 1972 wurde SELCOM an etlichen Rechenzentren installiert
und läuft auf Anlagen IBM/360 und /370 (PS44 und DOS), IBM/1130,
UNIVAC 1106 und CDC CYBER 72, 73 und 74.

Ein Einführungsvortrag über SELCOM findet statt am

Donnerstag, dem 15.5.1975, 17.00 Uhr ct

im Seminarraum des Institutes für Technische Mathematik, Technische
Hochschule Wien, Gußhausstraße 27-29, 4.Stock.

Alle Einzelheiten über SELCOM können auch dem SELCOM-Manual
entnommen werden, das im Sekretariat (TH) bzw. in der Programm-
beratung (UNI,TH) zum Preis von S 10.- erhältlich ist.

RASTL



MAI 1975

Benutzerhandbuch/30.60.-40

NEUES INTEGRATIONSPROGRAMM

Ab sofort ist an der CYBER 74 ein neues Integrationsprogramm in FORTRAN verfügbar, das den neuesten Stand der Softwareentwicklungen auf diesem Gebiet repräsentiert. Der Aufruf dieses Programms (in der Standardform) geschieht folgendermaßen:

```
CALL AQUA1(F,A,B,RESULT,IER)
```

wobei F der Name eines FORTRAN-Function-Unterprogrammes ist. F(X) muß den Funktionswert der zu integrierenden Funktion an der Stelle X liefern, auf A und B sind die untere bzw. obere Integrationsgrenze anzugeben, RESULT enthält nach dem Aufruf den berechneten Näherungswert für das Integral. Der Fehlerindikator IER wird beim Auftreten von Schwierigkeiten bei der Integralberechnung auf einen Wert $\neq 0$ gesetzt.

Der Aufruf des neuen Programmes kann außer in der Standardform auch in drei Varianten erfolgen, die dem Benutzer gewisse zusätzliche Möglichkeiten (options) bieten: er kann eine gewünschte Genauigkeit vorschreiben und eine Maximalzahl von Funktionsauswertungen vorgeben und erhält auf Wunsch eine Genauigkeitsabschätzung des Resultats sowie (über einen Parameter gesteuert) das Resultat und verschiedene andere Informationen ausgedruckt. Von der Möglichkeit des gesteuerten Printouts wird man vor allem in der Testphase größerer Programmprojekte Gebrauch machen. Wenn das Teststadium beendet ist, kann der Ausdruck völlig unterdrückt werden.

Sämtliche Programmvarianten sind auch in source (FORTRAN-IV) verfügbar. Dem Benutzer wird also eine Anpassung an seine speziellen Probleme ermöglicht.

Wodurch zeichnet sich das neue Programm aus? Dieses Programm hat bei internationalen Vergleichsstudien immer als eines der besten verfügbaren Programme abgeschnitten. Es wurde auch auf der CYBER 74 umfangreichen Tests (mit 2130 Testläufen) unterworfen.

Spezialfälle (Sprungstellen, integrierbare Singularitäten) werden von diesem Programm erkannt, verarbeitet und gegebenenfalls dem Benutzer mitgeteilt (Varianten 1-3).

Im Gegensatz zu den bisher verfügbaren Programmen kann das neue Programm auch Integranden mit stark unterschiedlichem Verhalten im Integrationsintervall sehr gut verarbeiten (z.B. sehr steile und sehr flache Stellen oder Oszillationen und glatte Stellen wechseln ab).

Bezüglich Schnelligkeit (ca 50msec pro Integration mit einer Genauigkeit von 10^{-6}) und Anzahl der Funktionsauswertungen schneidet das neue Programm sehr gut ab. Allerdings benötigt das Programm etwa 3500 Kernspeicherworte.

Da die Erfahrungen mit diesem Softwareprodukt als Grundlage für die Implementierung weiterer benutzerfreundlicher Software



dienen soll, möchten wir Sie bitten, in Zukunft möglichst ausschließlich das neue Integrationsprogramm zu verwenden. Mitteilungen über Ihre Erfahrungen wären uns deshalb besonders willkommen. Sie sind an Herrn Dipl. Ing. Ch. Überhuber, Institut für Numerische Mathematik, zu richten.

Sie haben mit dem neuen Programm den Vorteil, daß Sie insbesondere in kritischen Fällen wesentlich zuverlässigere Resultate erhalten werden, und mit den Varianten eine flexible Anpassung an Ihre spezielle Situation erreichen können.

Die angeführten Unterprogramme sind in übersetzter Form in der Library THLIB1 gespeichert. Programmbeschreibungen sind in der Programmberatung erhältlich, wo auch eine Dokumentation der durchgeführten Tests aufliegt.

Chr. Überhuber
Institut f. Num. Mathematik

ACHTUNG PROGRAMMBIBLIOTHEK THLIB1

Die Bibliotheksprogramme ALKURV und APLOTT liegen ab 75 05 12 in der früher üblichen Version vor. Die Ausgabe der beiden Programme erfolgt nun direkt auf CHANNEL 61.

Damit sind die Anweisungen

CHANNEL,71=HELPOUT
REWIND,HELPOUT.
COPYBF,HELPOUT,OUTPUT.

nicht mehr notwendig.

Die sonstige Verwendung der Programme bleibt gleich.

Bei Umstellungsschwierigkeiten wenden Sie sich bitte an die Programmberatung.

H. Mayer



SOFTWARE

SIGMA

0. Einleitung

SIGMA ist ein interaktives, array-orientiertes System. Seine Anwendung reicht vom simplen Tischrechnermodus bis zur Erstellung und Ausführung von komplexen Programmen, wobei die graphische Darstellung von Ergebnissen besonders hervorzuheben ist.

SIGMA ist in der Notation FORTRAN-ähnlich, ist aber bezüglich Anweisungsform vollkommen formatfrei (Trennung von Sprachelementen durch Plank(s)).

SIGMA wurde am CERH entwickelt ¹⁾ und steht seit MÄRZ 1975 auch an der CYBER 74 allen ²⁾ INTERCOM-Benutzern zur Verfügung.

Im folgenden wird ein kurzer Überblick über SIGMA gegeben:

1. Aufruf

COMMAND-ATTACH,SIGMA.

COMMAND-ATTACH,EXCC.

COMMAND-EXCC,SIGMA.

Daraufhin meldet sich SIGMA mit Headerinformationen und der Zeilennummer

1.

betriebsbereit. Nun kann eine SIGMA-Session durch Eintippen von SIGMA-Statements begonnen werden.

1) R.HAGEDORN, J.REINFELDS, C.VANDONI und L. van HOVE
"SIGMA, A new language for interactive arrayoriented
computing".

2) CM 65000 erforderlich!



2. BEENDIGUNG EINER SIGMA-SESSION

Eingabe von `!STOP` Rückgabe der Kontrolle an INTERCOM.
Programme und Variable, die nach dem letzten `!SAVE` erzeugt wurden, gehen dabei verloren.

`!BYE` Ähnlich `!STOP`, vorher wird noch `!STATUS` und `!NAMES` (siehe SIGMA-Commandliste, Seite 11) ausgeführt.

3. FEHLEINGABEN

`CTRL u. glz. H` Rücksetzen des Cursors
um 1 Stelle nach links
`CTRL u. glz. X` Löschen der letzten Zeile

4. DIE SPRACHE SIGMA

SIGMA erlaubt eine voll dynamische Datendefinition:

4.1 NAMEN (max. 7 alphanumerische Zeichen beginnend mit einem Buchstaben)

`A = 2` A ist ein Skalar; Wert 2
`A = A+1` - " - ; Wert 3
`A = A&A` ist ein eindimensionales Feld;
Elemente 3 3

`A = ARRAY (200,Ø#PI)` A ist ein eindimensionales Feld
mit 200 Elementen; Werte von Ø
bis 3.14 äquidistant verteilt.

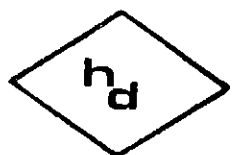
`A = ausdruck` A ist ein Skalar oder ein Feld,
definiert durch den allgemeinen
Ausdruck der rechten Seite.

`SUBROUTINE A` A ist der Name einer SUBROUTINE
(FORTRAN-ähnlich).

4.1.1 SCHUTZ VON NAMEN gegen Neudefinition:

`PROTECT name1,, name n`

`DEPROTECT name1,, name n` hebt den Schutz wieder auf



4.1.2 LÖSCHEN VON NAMEN: DELETE name1, ...,namen

4.1.3 GÜLTIGKEITSBEREICHE VON NAMEN

Namen von Programmen (FUNCTION, SUBROUTINE, MACRO) sind global auf allen Ausführungsebenen, Namen von Variablen (Feldern) sind lokal zur Definitionsebene. Durch

GLOBAL name1, ...,namen
können sie global gemacht werden.

4.2 NUMERISCHE GRÖSSEN

4.2.1 ZAHLEN

Form: $\pm ddd \dots d.dd..d$ max. 14 sign. Ziffern
oder $\pm dd \dots d.d..dE^{\pm ddd}$

z.B. 3.5 -28.3E28 4 Exponent < 240

Kein Unterschied zwischen Real- und Integerzahlen; sie werden intern gleich dargestellt.

Wahrheitswerte (wahr, falsch) werden durch 1 (wahr) bzw. 0 (falsch) dargestellt.

4.2.2 SKALARE

sind Zahlen; sie werden durch Zuweisungen erzeugt und unter Namen gespeichert:

name = ausdruck

|| Skalare werden von SIGMA wie eindimensionale Felder mit einem Element behandelt.

4.2.3 FELDER

Definition von Feldern:

(..) durch ARRAY-Operator

X = ARRAY (ausdruck) Alle ausdruck Komponenten
oder haben den Wert 1

X = ARRAY (ausdruck1,ausdruck2)
ausdruck1 gibt die Feldstruktur an
(=NCO<Number of Component>
Vektor)

ausdruck2 gibt die Wertbelegung an.



Bsp: X=ARRAY(4,0#PI)
Y=ARRAY(3&4,1#12)

3 x 4 Matrix mit Werten
von 1 bis 12

(..) durch Zuweisung
name = ausdruck

ausdruck hat als Ergebnis
ein Feld.

Bsp: Y=SIN (ARRAY(10,0#PI))

Y hat als Feldelemente
die 10 Sinuswerte von
0 bis π .

(..) durch Verkettung
x=ausdr1&ausdr2& &ausdr n

ausdr i ... Skalare, ein-
dimensionale Felder oder
Ausdrücke von Skalaren und
Feldern.

Bsp: X=1&2&3
Y=1&2*3 y=1&6
Z=(1&2)*3 z=3&6

Die Indexreihenfolge ergibt sich daraus, daß der am wei-
testen rechts stehende Index am raschesten variiert.

4.3 ARITHMETIK

4.3.1 ARITHMETISCHE OPERATIONEN

+ - * / **

Wirken komponentenweise auf Skalare bzw. Felder bzw. auf
zwei Felder mit demselben NCO-(Struktur)vektor,

Die konventionelle komponentenweise Ausführung von bi-
nären Operationen ist in SIGMA generalisiert und wird
"Topologische Arithmetik" genannt.



4.3.2 VERGLEICHS-UND LOGISCHE OPERATOREN

EQ NE GT GE LT LE

Operanden können wieder Skalare oder Felder sein (komponentenweise Operationen).

4.4 BENUTZER-DEFINIERTER PROGRAMMEINHEITEN

Ähnlich FORTRAN-Unterprogrammen, wobei Rekursionen möglich sind. Erweiterung um sogenannte MACRO's.

4.4.1 SUBROUTINEN

Mit oder ohne Parameterliste (keine RETURNS-Liste wie in FORTRAN-EXTENDED).

Esp: SUBROUTINE INTEGER(A)

 \$ druckt alle positiven Integerzahlen von A

 PRINT A*(A-ABS(INT(A))) EQ 0

 RETURN

Sei Z=ARRAY(9,-2#2)

Aufruf: CALL INTEGER(Z)

bewirkt die Ausgabe von

 0 0 0 0 0 0 1 0 2

4.4.2 FUNKTIONEN

Müssen mindestens einen formalen Parameter haben

Esp: FUNCTION GAUSS(SIGMA,X,X0)

 GAUSS=EXP(-((X-X0)**2/2/SIGMA**2))

 GAUSS=GAUSS/SQRT(2*PI*SIGMA**2)

 RETURN

 END

4.4.3 MACROS

haben keine formalen Parameter

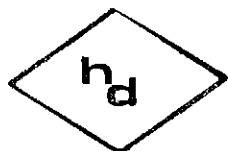
 MACRO NULL

 X=0

 Y=0

 Z=0

 END



Aufruf: CALL NULL

bewirkt, daß der Code an seine Stelle eingesetzt wird.

Namen in MACRO's stehen automatisch auf der rufenden Ebene.

Namen können durch

GLOBAL name1,, name_r in SUPROUTINEN und FUNKTIONEN global definiert werden.

4.5 KONTROLLSTRUKTUREN

4.5.1 BEDINGTE VERZWEIGUNG

IF (B) Anweisung

B muß ein logischer Ausdruck sein. Ist B ein Feld, so ist die Bedingung wahr, wenn alle Komponenten 1 sind.

IF(ANY(X**2 LT 0.01)) X=Ø * X

Bei

ANY (\sim) genügt schon eine Komponente, z.B. wenn ein $X_i^2 < 0.01$

4.5.2 MARKEN

Positive, ganze Zahl, max. 5-stellig

$1 \leq \text{Marke} \leq 99999$

Esp: 17 X=EXP(Y*Z)

2 Y(J)=X(J-1)*U(J)

Achtung: Zeilennummern sind keine Marken

4.5.3 UNBEDINGTE VERZWEIGUNG

GO TO n

oder GOTO n

n Marke, Name eines Skalars (mit dem Wert der Marke) o. ein skalarer Ausdruck

4.5.4 SCHLEIFEN

DO n J=S1,S2,S3

:

n Statement

Parameter und Wirkung wie bei FORTRAN.



4.5.5 SICHERHEITS-STATEMENTZÄHLER

Begrenzt die Zahl der seit dem letzten !SAVE ausgeführten Statements

!LIMIT ausdruck

Standardwert: 4000

4.6 EIN/AUSGABE

4.6.1 EINGABE von Werten mit Hilfe von Zuweisungen:

z.B. X=5.3E5

A=ARRAY(2&2,3.5&780.05&3.7E-3)

Z=ARRAY(100,1#100)

B=ARRAY(4&5&5,Z)

4.6.2 AUSGABE

PRINT ausdr1, ausdr2, ...,ausdr n

listet die Werte aller Elemente (=Ergebnisse der ausdr i)

Z.B. PRINT 4,I-PI/2,MAX,'UEBERSCHRIFT' bewirkt folgenden Ausdruck:

4.000000

I=10.00000

-1.5707963

1.SUBROUTINE MAX(X,Y)

2.IF(X LT Y) GO TO 10

3.PRINT X

4.RETURN

5.10 PRINT Y

6.RETURN

7.END

UEBERSCHRIFT

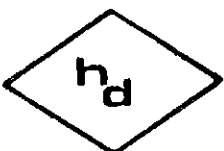
4.6.3 GRAPHISCHE AUSGABE

DISPLAY Y%X

zeichnet die Funktionswerte Y über den Argumentwerten (unabh. Variable) X

Y vertikal

X horizontal



Zwischenpunkte werden linear Interpoliert.

Druckzeichen für Kurvendarstellungen:

DISPLAY [zeichen] Y

Bsp: DISPLAY [*] Y%X, [Ø] Z%A, [-] B%X

4.6.4

DRUCKKONTROLLKOMMANDS

- !DIGITS n Anzahl der Ziffern für Werte, die mit PRINT ausgegeben werden ($1 \leq n \leq 14$)
- !LENGTH m $1 \leq m \leq 72$
Länge einer Ausgabezeile
- !PRINT }
!NOPRINT } Bei jedem Wertwechsel eines Skalars o. Array soll der neue Wert (nicht) ausgegeben werden.
(!PRINT für Tischrechnermodus empfohlen)
- !CHAIN }
!NOCHAIN } Jede Verkettung von Aufrufen bzw. jede Rückkehr von Subroutinen und Funktionen soll (nicht) ausgedruckt werden.

Standardannahmen:

!NOPRINT und !NOCHAIN

4.6.5

ZEICHENKONTROLLKOMMANDS

- !SCALE }
!NOSCALE } Automatische Skalierung auf optimale Zeichenflächenausnutzung
- !LINX } lineare }
!LOGX } logarithm. } Skalierung der { Abszisse
!LINY } lineare }
!LOGY } logarithm. } { Ordinate
- !GRID !FRAME !AXES }
!NOGRID !NOFRAME !NOAX } Zeichnen eines Netzes über, eines Rahmens um bzw. von Koordinatenachsen im Kurvenbild
- !NORMAL Alle Standardannahmen obiger Kommands werden rückgesetzt:
!LINX !LINY !NOGRID !FRAME !AXES !SCALE



4.6.6 SYSTEMPARAMETER

!LIMIT	ausdr	Maximalzahl der nach einem !SAVE auszuführenden Anweisungen
!LENGTH	ausdr	Anzahl der Druckpositionen pro Zeile
!DIGITS	ausdr	$1 \leq \text{ausdr} \leq 14$ Anzahl der auszugebenden Ziffern pro Wert
!XLENGTH	ausdr	Länge der $\left\{ \begin{array}{l} \text{Abszisse} \\ \text{Ordinate} \end{array} \right\}$ in Druckpositionen
!YLENGTH	ausdr	

Standardannahmen:

!LIMIT	4000
!LENGTH	70
!DIGITS	8
!XLENGTH	55
!YLENGTH	34

4.7 ARRAY-FUNKTIONEN

R = MULT (array1,array2)	Matrizenprodukt
R = INV(array)	Matrixinversion
D = DET (array)	Determinantenberechnung
R = EIGVAL(array)	Eigenwerte bzw.-vektor
R = EIGVEC(array)	
R = NCO(array)	Strukturvektor von array
R = ANY(array)	array muß Wahrheitswerte (=1 oder 0) haben
	$R = \begin{cases} 1 & \dots \text{mind.ein Element ist } 1 \\ 0 & \dots \text{alle Elemente } 0 \end{cases}$

5. (ARBEITS)SPEICHER-COMMANDS

!SAVE	pfname, ^{PROT} accnr, ^{OWR}	Abspeicherung des Arbeitsspeicherinhalts unter dem Permanentfilenamen pfn.
	^{OWPROT}	

- accnr ... Accountnummer
- PROT ... File ist gegen Überschreiben geschützt
- OWR ... Überschreiben des Files mit demselben Namen, auch wenn es geschützt ist.
- OWPROT .. Zusätzlich zu OWR soll das neue File geschützt werden.
- fehlend.. File wird ungeschützt gespeichert.



Fehlen alle Parameter, so werden sie interaktiv angefordert.

!LOAD pfn,accnr Ladet pfn in den Arbeitsspeicher
!CLEAR Löscht den Arbeitsspeicher
!COPY pfn,name1,...,namen
 Kopiert von pfn die Variablen o.
 Programme name 1, ...,namen

!EDIT progname
Listet das Programm progname mit Zeilennummern
Danach gibt das System die Meldung
TYPE C.L(COPY),D.L(DELETE) OR NEW STATEM.
Angabe von
C.L : Kopiert die Programmzeilen bis zur Zeile L
 exclusive.
D.L : Löscht die Programmzeilen bis zur Zeile L
 exclusive.

Neue Anweisung: Die neue Anweisung wird ins Programm
eingefügt.

Zur Prüfung des neuen Textes wird PRINT progname empfohlen.

6. ANDERE COMMANDS

!ERASE	Löschen des Bildschirmes
!WAIT	Bewirkt im Programm ein Halten der Ausführung, bis GO eingegeben wird.
!NEWS	Auflisten von SIGMA-Neuigkeiten
!STATUS	Statistikausgabe über Systemstatus
!NAMES !LABELS } !LIBRARY	Listen aller Namen und Marken der laufenden Programmebene Listen von Bibliotheksfunktionen

BRAUN



SOFTWARE

PASCAL

8. ZEICHEN FIN/AUSGABE

8.1 AUFBAU VON TEXTFILES

PASCAL verwendet für Textfiles standardmäßig Z-Records (zero-byte-Record), d.h. jeder Record (= Zeile, Karte) ist durch 12 Binärnullen abgeschlossen. Diese End-of-line-Marke, die durch die PASCAL-Prozeduren

writeln,readln,eoln

erzeugt, übersprungen bzw. erkannt werden kann, befindet sich rechtsbündig im letzten Datenwort. Kann das letzte Datenwort diese Marke (12 Binärnullen entspr. ::) nicht mehr aufnehmen, so wird dazu ein neues Datenwort verwendet, wobei die unbesetzten Wortteile - ein Wort enthält maximal 10 Zeichen - mit Flanks aufgefüllt werden. Es kann also die tatsächliche Zeichenzahl eines Records (=Zeile,Karte) um diese Füllblanks größer werden.

8.2 BESONDERHEITEN DES FILES INPUT

Liest ein PASCAL-Programm vom File INPUT (Kartenleser) Datenkarten, so beziehen sich die

read- bzw. get- Anweisungen

auf jene Z-Records, die das Betriebssystem aus den Datenkarten erzeugte, die sich zwischen zwei entsprechenden EOR-Karten (7/8/9-Lochung in Spalte 1) befanden.

Ist dabei eine Karte nur bis zur Spalte i gelocht, so besteht der entsprechende Z-Record nur aus den i Zeichen + End-of-line Marke (+ev. Füllblanks).

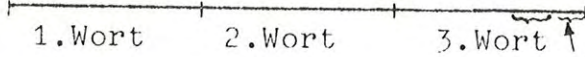
Esp.:

DAS IST EINE UEBERSCHRIFT <div style="text-align: right; margin-right: 20px;"> ↑ 26 </div>
--

Der entsprechende Z-Record hat dann folgende Form bzw. Länge:



DASLIST EINELUEBFERSCHRIFTL::



EOL-Marke (12 Binärnullen entspr.:)

zwei Füllblanks

Liest man also bis zum Ende der "Karte", z.B. durch

```
read(ch);i:=1;
while not eoln(input)and(i<=80)do
    begin
    :
    Verarbeitung des nächsten
    Zeichens
    :
    i:=i+1;
    read(ch)
    end;
```

so hat i nach Beendigung der while-Schleife den Wert 29 und ch wird definitions gemäß Blank gesetzt. (Entspricht der End-of-line-Marke).

Beim nächsten

```
read(ch)
```

wird ch das 1. Zeichen der nächsten "Karte" (=nächster Z-Record) zugewiesen.

Wird beim Lesen die EOF-Karte oder FOR-Karte erreicht, so bedeutet dies File-Ende; ch wird undefiniert.

BRAUN



tips

der Programm beratung

UNTERDRUECKEN DES PROGRAMMABBRUCHS BEI DATENFEHLERN IN FORTRAN

Beim Versuch, Daten einzulesen, die auf Grund der FORMAT-Anweisung illegal sind (z.B. alphabetische Daten in einem numerischen Feld, oder Zahlenwerte außerhalb des erlaubten Zahlenbereichs), wird ein FORTRAN-Programm mit der Fehlermeldung

```
ERROR NUMBER 78      ILLEGAL DATA IN FIELD  
oder ERROR NUMBER 79  DATA OVERFLOW
```

abgebrochen. Dies kann verhindert werden, indem man das Systemunterprogramm ERRSET vor dem betreffenden Einlesebefehl aufruft:

```
CALL ERRSET (num,lim)
```

Dieses Systemunterprogramm steht natürlich nur im CDC-FORTRAN zur Verfügung und kann auf anderen Rechenanlagen nicht verwendet werden.

Die Parameter sind

num ganzzahlige Variable, in der die gesamte Anzahl der festgestellten Fehler mitgezählt wird;

lim ganzzahlige Konstante oder Variable, die angibt, nach wievielen Datenfehlern der Abbruch erfolgen soll;

ERRSET kann für die Eingabe mit formatiertem, listengesteuertem oder NAMELIST-READ verwendet werden. Beim Auffinden eines Datenfehlers wird eine Fehlermeldung geschrieben, der Wert von "num" um 1 erhöht, und die Programmausführung fortgesetzt. Der Wert der betreffenden Variablen bleibt unverändert. Bei formatiertem oder NAMELIST-READ wird nur das fehlerhafte Datenfeld übergangen, die übrigen Variablen werden korrekt eingelesen.

Wenn ERRSET ein zweitesmal aufgerufen wird, kann das Fehler-Limit abgeändert und der Zähler "num" neu initialisiert werden. Sobald "num" den Wert der Grenze "lim" erreicht hat, bricht das Programm ab.

PECHTER

MAI 1975

Benutzerhandbuch/30.60.-55

PROGRAM EXAMPL(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
DIMENSION ZAHL(5)

C
C SETZEN DES FEHLERLIMITS DURCH AUFRUF VON ERRSET
C

CALL ERRSET(NUM,5)

C
1 READ(5,1) ZAHL
10 1 FORMAT(F4.1)
WRITE(6,2) NUM
2 FORMAT("0 ANZAHL DER DATENFEHLER...",I2)
WRITE(6,3) ZAHL
3 FORMAT("0 DIE EINGELESENEN ZAHLEN SIND..."(6X,F5.1))
15 STOP
END

47.1
25./
48.3
24,1
91.2

RECORD NO. 2 25./
...'......12345678901234567890123456789012345678901234567

* ERROR DATA INPUT * ILLEGAL DATA IN FIELD ***
ERROR NUMBER 0078 DETECTED BY INCOM= AT ADDRESS 000200
CALLED FROM KRAKER= AT ADDRESS 000353
CALLED FROM INPC= AT ADDRESS 000075
CALLED FROM EXAMPL AT LINE 0008

RECORD NO. 4 24,1
...'......12345678901234567890123456789012345678901234567

* ERROR DATA INPUT * ILLEGAL DATA IN FIELD ***
ERROR NUMBER 0078 DETECTED BY INCOM= AT ADDRESS 000200
CALLED FROM KRAKER= AT ADDRESS 000353
CALLED FROM INPC= AT ADDRESS 000075
CALLED FROM EXAMPL AT LINE 0008

ANZAHL DER DATENFEHLER... 2

DIE EINGELESENEN ZAHLEN SIND...

47.1
0.0
48.3
0.0
91.2



tips

der
Programm
beratung

UNTERSCHIEDEN VON NULL UND BLANK BEIM EINLESEN IN

I-FORMAT IM FORTRAN

Bei der Eingabe von Leerzeichen im I-Format wird der Wert Null gelesen; gelegentlich ist es wünschenswert, zwischen Blanks und Nullen zu unterscheiden. Dies kann im CDC-FORTRAN in einfacher Weise mit Hilfe der eingebauten Funktion

ISIGN (n, m)

geschehen, die eine Vorzeichenübertragung vom Argument m auf das Argument n bewirkt.

Blanks werden nämlich bei der Eingabe im I-Format als negative Null (oktal 777...77) verschlüsselt, während Nullen +0 (oktal 00...00) ergeben. In den meisten anderen Fällen verhalten sich -0 und +0 gleich, sodaß man bei der Rechnung darauf nicht mehr Rücksicht zu nehmen braucht, eine Unterscheidung ist jedoch mit der Funktion ISIGN möglich.

Beispiel:

```
      READ(5,1)M
1  FORMAT(I1)
      IF(M.EQ.0.AND.ISIGN(1,M).EQ.+1)GOTO 2
      IF(M.EQ.0.AND.ISIGN(1,M).EQ.-1)GOTO 3
      .
      .
      .
```

Es erfolgt Sprung nach 2, wenn eine Null, und Sprung nach 3, wenn ein Blank eingelesen wurde.

Wird ein im I-Format eingelesenes Blank wieder im Format I1 ausgegeben(als-0), so wird ein Stern gedruckt (wegen der Formatüberschreitung).

Diese Methode zur Unterscheidung von Null und Blank ist natürlich nur auf CDC-Anlagen möglich, und daher völlig systemabhängig.



tips

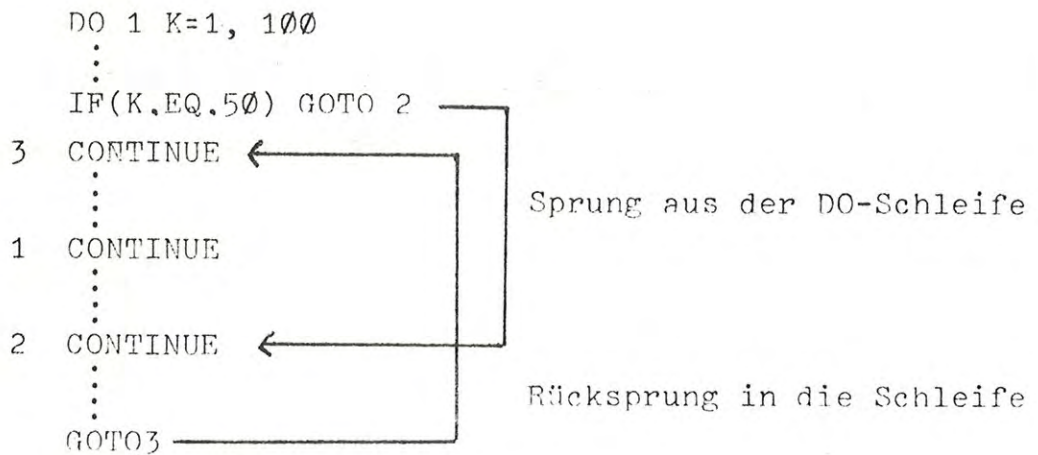
der Programm beratung

FEHLER DES FTN-COMPILERS IM ZUSAMMENHANG MIT ERWEITERTEM

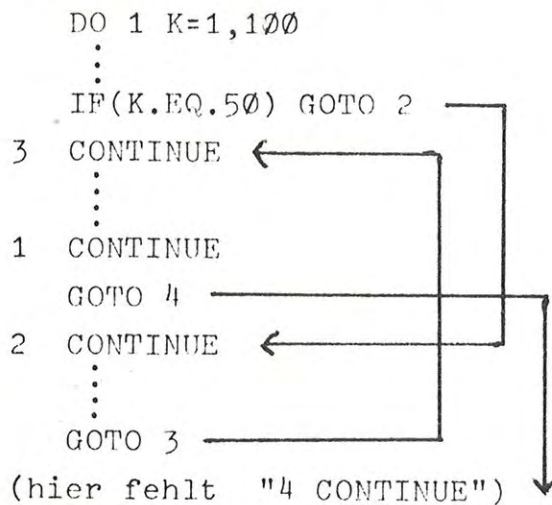
DO-SCHLEIFENDEREICH

Ein Sprung aus einer DO-Schleife mit anschließendem Rücksprung in die Schleife ist in FORTRAN erlaubt.

Beispiel:



In dieser Situation ist der FTN-Compiler unter Umständen nicht in der Lage, fehlende Anweisungsnummern, die in Sprungbefehlen angesprochen werden, zu erkennen. Im folgenden Beispiel fehlt das Sprungziel der Anweisung GOTO4



Man erhält keine Fehlermeldung vom Typ FE ("Fatal Error"), obwohl man "Undefined Statement Number" erwarten würde. Das Programm wird auch ausgeführt, jedoch ist der Programmablauf undefiniert, und es tritt eine Run-time-Fehlermeldung auf, die nicht leicht zu verstehen ist.

In den durchgeführten Testläufen traten die Fehlermeldungen

```
ERROR MODE=Ø. ADDRESS=000001
```

bzw.

```
PP CALL ERROR
```

auf.

Ein Hinweis auf den Fehler ist der Reference-Map (FTN,R=3) zu entnehmen; darin ist die betreffende Anweisungsnummer mit dem Zusatz * UNDEF versehen.

Falls im Programm kein Rücksprung in die DO-Schleife erfolgt, oder falls der erweiterte Schleifenbereich vor der DO-Schleife selbst liegt, ist die Fehlerdiagnostik wieder korrekt.

PECHTER







ANSCHAFFUNG VON APEX ?

Das Institut für Betriebswirtschaftslehre der Universität Wien hat Interesse, das CDC-Programmprodukt APEX verwenden zu können. APEX ist ein leistungsfähiges Programmsystem zur linearen Programmierung. Da die Anschaffung dieses Programmsystems allerdings mit erheblichen Kosten verbunden ist, ersucht das Rechenzentrum alle daran interessierten Benutzer, sich mit Dr. Rastl (Rechenzentrum UNI) oder Dr. Knolmayer (Institut für Betriebswirtschaftslehre, Wien 1090, Türkenstraße 23/11, Telefon: 34-74-79) in Verbindung zu setzen. Die diesbezüglichen CDC-Manuals können bei Dr. Rastl (UNI) sowie Dipl.Ing.Welser (TH) eingesehen werden.

RASTL



MAI 1975

k u r s e

Das Interfakultäre Rechenzentrum der Universität Wien veranstaltet im Sommersemester 1975 folgende Kurse:

1. EINFÜHRUNG IN DIE FORTRAN-PROGRAMMIERUNG

Kursleitung: R. Wytek

Zeit: 1. Termin: Di 13.5.-Fr 16.5.,
Mi 21.5.-Fr 23.5.,
Mo 26.5.-Mi 28.5.1975

2. Termin: Mo 2.6.-Fr 6.6.,
Mo 9.6.-Fr 13.6.1975

jeweils 17 bis 20 Uhr
(siehe auch Anmerkung)

Ort: Hörsaal 3 der evangelisch-theologischen Fakultät,
1010 Wien, Liebiggasse 5, 3.Stock

Anmeldung: Sekretariat des Rechenzentrums (UNI)

Kursgebühr: S 200.- (siehe auch Anmerkung)

Inhalt: Einführung in die Programmierung in der Programmiersprache FORTRAN. Vorkenntnisse sind nicht erforderlich. Dieser Kurs soll Anfänger mit den nötigen Kenntnissen ausstatten, für einfache Probleme Fortran-Programme zu erstellen und von dieser Grundlage aus selbständig weitergehende Programmiererfahrung zu erwerben. Bei diesem Kurs werden regelmäßig Übungen an den Einrichtungen des Rechenzentrums durchgeführt.

Anmerkung: Die beiden Kurse finden in Zusammenarbeit mit dem Psychologischen Institut statt. Jene Kursteilnehmer, die den Kurs regelmäßig besuchen und am Ende einen Test über die Wirkung des Unterrichts absolvieren, erhalten die halbe Kursgebühr rückvergütet.

Es besteht zusätzlich die Möglichkeit, den Kurs im Selbststudium mittels einer "Programmierten Unterweisung" (erhältlich am ersten Kursabend) zu absolvieren.



MAI 1975

2. SCOPE-SYSTEMSOFTWARE

Zeit: 27.5.-18.6.1975, jeweils 9.00 bis 12.00 Uhr
Programm: (Änderungen vorbehalten)

Di 27.5.:	Allgemeine Einführung in SCOPE	(K.Pechter)
Mi 28.5.:	SCOPE Job Control	(K.Pechter)
Fr 30.5.:	FTN-Compiler, Debug	(R.Kofler)
Di 3.6.		
Mi 4.6.:	Intercom	(W.Altfahrt)
Do 5.6.		
Fr 6.6.:	Loader, insb.Segmentierung	(W.Weisz)
Di 10.6.		
Mi 11.6.:	Bänder, diesbez.Record Manager	(W.Weisz)
Do 12.6.		
Fr 13.6.:	Libraries, Update	(W.Altfahrt)
Di 17.6.:	Sort/Merge	(R.Kofler)
Mi 18.6.:	Utilities	(K.Pechter)

Ort: Seminarraum des Institutes für Mineralogie und Kristallographie, Hauptgebäude der Universität, Erdgeschoß

Anmeldung: Sekretariat des Rechenzentrums (UNI)

Kursgebühr: keine

Beschränkte Teilnehmerzahl. Auch der Besuch von Einzelvorträgen ist möglich.

3. COMPASS

Zeit: Di 23.6. bis Fr 27.6.1975, jeweils 9.00 bis 12.00 Uhr

Ort: Seminarraum des Institutes für Mineralogie und Kristallographie, Hauptgebäude der Universität, Erdgeschoß

Anmeldung: Sekretariat des Rechenzentrums (UNI)

Kursgebühr: keine

Beschränkte Teilnehmerzahl

Inhalt: Einführung in COMPASS, die Assemblersprache der CDC-CYBER.

Der Kurs wird nicht PP-COMPASS behandeln.



MAI 1975

4. BENUTZUNG DES RECHENZENTRUMS

Es wird künftig regelmäßig einmal monatlich eine Einführung in die Benutzung des Rechenzentrums gegeben:

Termine: Freitag, 23.5.1975 9.00 bis 12.00 Uhr
 Freitag, 20.6.1975 9.00 bis 12.00 Uhr

Ort: Seminarraum des Institutes für Mineralogie
 und Kristallographie, Hauptgebäude der Uni-
 versität, Erdgeschoß.

Anmeldung: keine.

RASTL



MAI 1975

HANDBÜCHER

Die Informationsschriften und Handbücher der Rechenzentren UNI und TH sind ab sofort an beiden Rechenzentren erhältlich (UNI: Programmberatung, TH: Sekretariat und Programmberatung).

Informationsschriften UNI

Preis 10.-

INTERCOM (erweiterte Neuauflage)
LOCHERBESCHREIBUNG (Neuauflage)
SELCOM (3.Aufl., Version CDC)
ALGOL
UPDATE
SERIE BYTE
SCOPE
FORTRAN-UMSTELLUNG

Informationsschriften TH

SCOPE 3.4	Handbuch	(92 Seiten)	30.-
SCOPE 3.4	Kurzfassung	(22 Seiten)	10.-
RD1	Kursunterlage (als ALGOL-Unterlage verwendbar; 52 Seiten)		15.-
BENUTZUNG DER PROGRAMM- BIBLIOTHEK	Kurzfassung (2.Aufl. in Vorbereitung)	(23 Seiten)	20.-
ANLEITUNG ZUR BENUTZUNG VON MAGNETBÄNDERN	Kurzfassung	(5 Seiten)	5.-
ALGOL UMSTELLUNGSKURS	Kursunterlage	(30 Seiten)	10.-
COBOL UMSTELLUNGSKURS	Kursunterlage	(22 Seiten)	10.-
FORTRAN UMSTELLUNGSKURS	Kursunterlage	(30 Seiten)	10.-
INTERCOM	Handbuch	(ca.40 Seiten)	20.-
ANLEITUNG ZUR BENUTZUNG VON LOCHSTREIFEN	Kurzfassung	(6 Seiten)	5.-
FORTRAN	Handbuch (2.Aufl.)	(120 Seiten)	45.-
FORTRAN	Fehlermeldungen	(40 Seiten)	15.-
<u>NEU</u> PASCAL	Handbuch	(50 Seiten)	20.-



MAI 1975

Benutzerhandbuch/20.60-1

HANDBÜCHER

An den Rechenzentren UNI und TH liegen sämtliche relevanten CDC-Manuals in der Programmberatung zur Einsicht auf.

Die folgenden CDC-Manuals liegen auch zum Verkauf gegen Lieferschein auf (UNI: Programmberatung, TH: Sekretariat). Die Rechnung erhält der Käufer von der Fa. CDC zugeschickt.

60307200 SCOPE VERSION 3.4 REF MAN
60358700 SCOPE VERSION 3.4 USER'S GUIDE
60342500 UPDATE REF MAN
60360200 UPDATE INSTANT
60344200 LOADER REF MAN
60372200 LOADER INSTANT
60360900 COMPASS VERSION 3 REF MAN
60361000 COMPASS VERSION 3 INSTANT
60361700 COMPASS VERSION 3 Instruction Card
60305600 FORTRAN EXTENDED VERSION 4 REF MAN
60357900 FORTRAN EXTENDED VERSION 4 INSTANT
60359400 8-BIT SUBROUTINES VERSION 1.0 REF MAN
60384100 COBOL VERSION 4 REF MAN
60328400 COBOL VERSION 4 INSTANT
60343900 SORT/MERGE VERSIONS 4 AND 1 REF MAN
60328800 SYMPL VERSION 1 REF MAN
19980300 BASIC LANGUAGE VERSION 2 REF MAN
60384700 ALGOL VERSION 4 REF MAN
60307300 RECORD MANAGER VERSION 1 REF MAN
60359600 RECORD MANAGER VERSION 1 FILE ORGANIZATION USER'S GUIDE
60385200 RECORD MANAGER GUIDE FOR USERS OF FORTRAN EXTENDED
60385300 RECORD MANAGER GUIDE FOR USERS OF COBOL
60307100 INTERCOM VERSION 4 REF MAN



MAI 1975

Benutzerhandbuch/20.60.-2

60359700 INTERCOM INTERACTIVE GUIDE FOR USERS OF FORTRAN EXTENDED
60385700 INTERCOM INTERACTIVE GUIDE FOR USERS OF COBOL
60384600 INTERCOM INTERACTIVE COMMAND SUMMARY
60307000 FORM VERSION 1 REF MAN
60384900 QUERY UPDATE VERSION 2 REF MAN (VOLS 1 & 2)
60385500 QUERY UPDATE VERSION 2 MINI USER GUIDE
60359000 DATA DESCRIPTION LANGUAGE (DDL) VERSION 1 REF MAN
60384300 DATA DESCRIPTION LANGUAGE (DDL) VERSION 1 USER'S GUIDE
60359200 DATA DESCRIPTION LANGUAGE FOR THE QUERY UPDATE SUB-SCHEMA
(QU DDL) REF MAN
60358500 SIMSCRIPT VERSION 3 REF MAN
60234800 SIMULA REF MAN
60235100 SIMULA INSTANT
60328900 MATH.SCIENCE LIBRARY GENERAL INFORMATION MANUAL



