

8520 Erlangen

# RRZE

# BENUTZER INFORMATION

A circular stamp with the text "ERLANGEN-NÜRNBERG UNIVERSITÄT REGIONALES RECHT" around the top and "FACHHOCHSCHULE NÜRNBERG" around the bottom. The center contains the text "ERLANGEN-NÜRNBERG UNIVERSITÄT" and "FACHHOCHSCHULE NÜRNBERG".

BI 23 -ERLANGEN- 27.MÄRZ 1981

HERAUSGEBER: F. WOLF

REDAKTION: H. HENKE  
M. ABEL

R R Z E  
REGIONALES RECHENZENTRUM ERLANGEN  
MARTENSSTRASSE 1  
8520 ERLANGEN

TEL: 09131 / 85-7031 ODER 85-7032

KOLLEGIALE LEITUNG

Prof. Dr. F. Hofmann  
Prof. Dr. F. Lempio  
Prof. Dr. P. Mertens

TECHNISCHER DIREKTOR

Dr. F. Wolf

BETEILIGTE EINRICHTUNGEN:

UNIVERSITÄT ERLANGEN - NÜRNBERG MIT

RECHENZENTRUM

Aufsicht Tel.: 09131/85-7039  
Beratung Tel.: 09131/85-7040

AUSSENSTELLE ERLANGEN INNENSTADT

Mathematisches Institut  
Bismarckstr. 1 1/2  
Tel.: 09131/85-2560

AUSSENSTELLE ERLANGEN SÜDGELÄNDE

Physikalisches Institut  
Erwin-Rommel-Str. 1  
Tel.: 09131/85-7405

AUSSENSTELLE NÜRNBERG WISO

Lange Gasse 20  
Tel.: 0911/5302-272

AUSSENSTELLE NÜRNBERG FINDELGASSE

Sozialwissenschaftliches  
Forschungszentrum  
Tel.: 0911/5302-652

UNIVERSITÄT BAMBERG,

Prof. Dr. F. Vogel  
Feldkirchenstr. 21  
Tel.: 0951/402-219

UNIVERSITÄT BAYREUTH,

Dr. Siller/Dr. Wunderatsch  
Universitätsstr. 30  
Tel.: 0921/55-2316

FACHHOCHSCHULE COBURG,

Prof. Dr. H. Meyer  
Friedrich-Streib-Str. 2  
Tel.: 09561/36150

FACHHOCHSCHULE NÜRNBERG,

Prof. Dr. W. Baumann  
Keßlerstr. 40  
Tel.: 0911/533138



## Inhaltsverzeichnis

1	Aktuelle Informationen	2
1.1	Termine	2
1.2	Benutzerkolloquium am 24.02.1981	2
1.3	Plotterfile	5
1.4	Plotterpapier	5
1.5	Magnetbänder bei Nacht:	5
2	Neues von der Software	6
2.1	Neue Betriebssystemversion (Level 531)	6
2.1.1	Änderungen im Dialog	6
2.1.2	Änderungen bei UPDATE	7
2.1.3	Neuer BASIC-Compiler 3.5	8
2.1.4	Entfallende Kommandos	8
2.2	SUBMIT-Kommando und Output-Listen im Dialog	8
2.2.1	SUBMIT-Kommando	8
2.2.2	Holen der Liste	9
2.2.3	Löschen/Abbrechen	10
2.2.4	Information über festgehaltene Listen	10
2.2.5	Zusammenfassung	11

### Anhang:

- A) Die Programmiersprache ADA
- B) "Permanent-File" - Unterprogramme für FORTRAN
- C) Lehrveranstaltungen im Sommersemester 1981

# 1 Aktuelle Informationen

## 1.1 Termine

02.06.81 DEC-Benutzertreffen am RRZE. (getrennte Einladung folgt)

23.06.81 Benutzerkolloquium

Lehrveranstaltungen des Sommersemesters 1981 siehe Anhang C

## 1.2 Benutzerkolloquium am 24.02.1981

Protokoll: H.-W. Hein

### Bericht des RRZE

#### a) Betriebsumfang 1980

	TR440	CYBER
Jobs	: 165000	215000
davon im Dialog	: 35%	37%
Terminalstunden	: 47000	53000
CPU-Zeit	: 15200	6170
davon für Dialog:	9%	7%

Daran hatten Anteil:

Naturwissenschaftliche Fakultät II	68%
Naturwissenschaftliche Fakultät I	15%
Technische Fakultät	6%
ZUV und RRZE	6%
Region	2%
andere	3%

#### b) Allgemeine Situation

Nach der großen Reinigungsaktion lief der Betrieb im Jahre 1980 an der TR440 befriedigend, mit Einbruch des Winters ergaben sich kleinere Schwierigkeiten mit den Rechnerkernen.

An der CYBER ist das Kommunikationsnetz zeitweise sehr stark belastet. Dies wird sich mit der Inbetriebnahme eines zweiten Kommunikationsrechners demnächst bessern.

Es läßt sich eindeutig feststellen, daß die Benutzung der TR440 ab-



nimmt und die der CYBER zunimmt. Das RRZE empfiehlt, kurzfristige Projekte an der TR440 abzuwickeln.

In der Kernarbeitszeit sind die CYBER-Terminals alle belegt; morgens und abends sind genügend Geräte frei. Das RRZE empfiehlt, dies auch auszunutzen.

#### c) Anschluß benutzereigener Terminals

Da es nicht garantiert ist, dass bei Bedarf auch Anschlüsse frei, oder bestimmte Fabrikate überhaupt anschließbar sind, bittet das RRZE die Benutzer in ihrem eigenen Interesse, nur nach Rücksprache Geräte zu kaufen. Das RRZE ist bei Beschaffungen gerne behilflich, insbesondere erhält es auch von einigen Herstellern erhebliche Rabatte.

#### d) Wähleingänge

Hier besteht ein Engpass an der CYBER, der durch einen Nummerntausch behoben werden soll:

##### alte Nummern

CYBER	7691 ... 7698
TR440	7961 ... 7965 und 7991 ... 7995

##### neue Nummern ab Anfang Mai 1981

CYBER	7691 ... 7695 und 7961 ... 7965
TR440	7991 ... 7995

#### e) Wartungszeiten

CYBER	Mo entfällt ab Anfang Mai Mi 8 bis 14 (!) Uhr
TR440	Di 6 bis 14 Uhr unverändert Do 8 bis 11 Uhr unverändert

### Betriebssysteme

a) TR440 keine Veränderungen

b) CYBER

- SUBMIT-Kommando

Im Zusammenhang mit dem SUBMIT-Kommando wurden bestimmte Voreinstellungen geändert (siehe Kap. 2). In der Folge blieben viele Listen in Ausgabewarteschlangen hängen, die von Benutzern explizit hätten freigegeben werden müssen. Mit dem Befehl STATUS,JN prüft man das Vorhandensein solcher Listen nach. Folgende Befehle werden in absehbarer Zeit nicht mehr existieren:

DISPOSE	(jetzt ROUTE)
SETID	(durch ROUTE abgedeckt)
RWF	(jetzt REWIND,*)
RTF	(jetzt RETURN,*)

- Neue Betriebssystemversion (Level 531) ab Anfang April (Siehe "Neues Betriebssystem CYBER" in Kap. 2)

### Textverarbeitung

Anfang Dezember 80 fand eine Besprechung mit interessierten Benutzern statt. Eine Kommission hat inzwischen die bereits existierenden TV-Makrodateien gesichtet und im Januar eine Wunschliste für eine Benutzerschnittstelle aufgestellt.

Folgende Verbesserungen sind zu erwarten:

- Groß-Klein-Drucker wird direkt verwendbar.
- Am Diablo werden verschiedene Schrifttypen für TV, sowie die Ausgabe von Exponenten und Indizes möglich.
- Es wird in drei Stufen eine neue und dokumentierte TV-Makrodatei erstellt. Die Dokumentation der ersten Stufe wird Ende März, die der zweiten Stufe Ende April vorliegen. Danach werden jeweils die Makros implementiert. Das gesamte System soll noch 1981 fertig werden.

### Verschiedenes

- a) Die Frage, wie man nach einem deadstart einen Job noch identifizieren kann, wurde wie folgt beantwortet:

Bei einem Level 0 deadstart erhält der Job einen neuen Namen und ist nicht mehr identifizierbar.

Bei einem Level 3 deadstart bleibt der Name eines Jobs der alte.

- b) Das RRZE hat eine Stelle für den Aufgabenbereich Kommunikationsperipherie ausgeschrieben (siehe Aushang).
- c) Am Dienstag 02.6.81 um 14 Uhr findet im H4 erstmals ein DEC-Benutzertreffen statt. Vertreter von DEC werden Anwendungsmöglich-



keiten vorstellen, es wird über Anschlußformen DEC<-->RRZE informiert und es können Erfahrungen ausgetauscht werden.

- d) Zum Stichwort COM nichts Neues.
- e) Ein Benutzer berichtet, daß dem CYBER-PL/1-Compiler wichtige Sprachelemente (bzgl. Matrizen) fehlen. Das RRZE sieht keine Möglichkeiten zur Abhilfe.
- f) Das nächste Benutzerkolloquium findet am 23.6.81 statt.

### 1.3 Plotterfile

Der Plotterfile an beiden Rechenanlagen ist recht gross, so gross, dass der Bensonplotter etwa 17 Stunden braucht, um einen vollen File-Inhalt zu bearbeiten. Trotzdem hat es ein Benutzer ganz allein geschafft, an einem Wochenende den File zum Überlaufen zu bringen.

Wir empfehlen deshalb hier noch einmal:

Heben Sie ihren eigenen Grafikfile auf, denn der öffentliche Plotterfile könnte überlaufen. Sie sparen sich dadurch teure Wiederholungsläufe.

### 1.4 Plotterpapier

Das RRZE gibt aus alten Beständen Plotter-Papier-Rollen (12 Zoll breit) an Institute kostenlos ab, soweit der Vorrat reicht. Bitte sprechen Sie Herrn Thomas an (Tel. 09131/857031).

### 1.5 Magnetbänder bei Nacht:

Es hat sich herausgestellt, dass es für Bänder und Geräte nicht gut ist, wenn ausserhalb der bedienten Zeiten Bänder z.T. stundenlang aufliegen. Deshalb bitten wir alle Benutzer, bei der Verarbeitung von Bändern die Möglichkeit wahrzunehmen, das Band auf Platte zu kopieren und dann als Plattenfile zu bearbeiten. Die Beratung (Tel. 85-7040) hilft Ihnen gerne dabei.

## 2 Neues von der Software

### 2.1 Neue Betriebssystemversion (Level 531)

An der CYBER ist eine neue Betriebssystemversion vorhanden.  
Diese wird ab

Montag, den 23. 3. 81  
im normalen Betrieb verwendet.

Das neue System unterscheidet sich von der alten Systemversion durch

- Korrekturen der Netzwerk-Software,
- Verbesserungen der Compiler,
- Einführung einer "type-ahead"-Möglichkeit im Dialog.

Vor allem die letztgenannte Erweiterung bedingt eine Änderung an der Dialog-Benutzer-Schnittstelle.

Die wesentlichen Änderungen des Systems sind:

#### 2.1.1 Änderungen im Dialog

- a) Es ist ein "type-ahead" möglich, d.h. während der Bearbeitung eines Kommandos kann bereits das nächste Kommando eingetippt werden, das dann nach Beendigung des ersten Kommandos automatisch anschliessend bearbeitet wird. Es können mehrere Kommandos im voraus eingegeben werden, deren maximale Anzahl von der Gesamtbelastung des Rechners abhängt.

Während der Eingabe eines Folgekommandos (vom Eintippen des ersten Buchstabens bis zum RETURN) ist die Ausgabe blockiert, d.h. die Ausgaben des zuerst eingegeben Kommandos, wozu auch eine evtl. erscheinende Eingabe-Aufforderung "?" gehört, werden erst nach vollständiger Eingabe eines Folge-Kommandos ausgegeben.

Wird ein Kommando mit ((CTRL)+T) abgebrochen oder mit ((CTRL)+P) unterbrochen, so werden die gespeicherten Kommandos gelöscht.

Auch bei Programmunterbrechung durch den Rechner wegen \*TIME LIMIT\* oder \*SRU LIMIT\* verschwinden die gespeicherten Kommandos.

- b) Wegen a) können die z.Z. während der Bearbeitung eines Kommandos möglichen Status-Abfragen (CR) bzw. STATUS (CR) nicht mehr verwendet werden, sie würden als nächste Eingabe abgespeichert. Statt dessen gibt es die folgenden Netz-Kommandos:

statt ENQUIRE (CR) oder

STATUS (CR)

jetzt: (ESC) E (CR)

statt (CR)

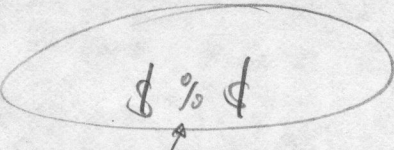
jetzt: (ESC) S (CR)

#### Statusabfrage im Dialog

Das Zeichen ESC hat bei manchen Geräten auch lokale Bedeutung (z.B. beim Teleray). Um unerwünschte Nebenwirkungen zu vermeiden, kann nach dem LOGIN mit dem Kommando TRMDEF dieses Zeichen undefiniert werden.

Beispiel: statt ESC soll % (Prozentzeichen) verwendet werden:




  
 TRMDEF,CT=%

Von da ab ist statt ESC E die Folge % E zu verwenden. Nach LOGOUT wird diese Umdefinition automatisch wieder zurückgesetzt.

- c) Es empfiehlt sich nicht mehr, eine laufende Ausgabe durch ein beliebiges Zeichen (z.B. Leertaste) anzuhalten und durch (CR) fortzusetzen, auch dies würde wieder eine ungewollte Abspeicherung dieses Zeichen als nächste Eingabe bewirken.  
Die richtige Methode ist:  
Anhalten der Ausgabe mit (BREAK) ! Danach entweder  
Fortsetzung mit (CR) oder  
Abbruch mit ((CTRL)+T) (CR) oder  
Unterbrechung mit ((CTRL)+P) (CR) wonach wiederum geht  
 (CR) zur Fortsetzung der Ausgabe  
 P (CR) zum Löschen des Ausgabepuffers und Programm-Fortsetzung  
 ein anderes Zeichen (CR) zum Abbruch
- d) Es ist nun möglich, im Dialog schrittweise eine LOADER-Sequence einzugeben, z.B.:  
 LDSET,LIB=EGSLIB  
 LGO
- e) Nach TEXT-Mode erfolgt jetzt automatisch ein PACK.
- f) Nach BYE bzw. LOGOFF werden evtl. verstellte Terminal-Parameter auf ihre Voreinstellungswerte zurückgesetzt; bei HELLO bzw. LOGIN bleiben sie bestehen.
- g) Das Kommando LIBRARY hat nun auch im Dialog die gleiche Bedeutung wie im Batch: Vereinbarung des Global-Library-Set.  
Das neue Dialog-Kommando LIB,lfn hat die Bedeutung OLD,lfn/UN=LIBRARY (die bisher im Dialog das LIBRARY-Kommando hatte).
- h) Mit X,... ist nun jederzeit jedes beliebige Batch-Kommando absetzbar, auch wenn noch nicht das Kommando BATCH gegeben worden ist, oder ein anderes Subsystem (FORTRAN oder BASIC) gewählt ist.

### 2.1.2 Änderungen bei UPDATE

UPDATE kann nun auch mit dem grossen Zeichensatz (gleicher Code wie der FCOPY-Output) arbeiten. Ausserdem sind längere Zeilen bearbeitbar. Im Dialog ist die Eingabe mehrerer Eingabe-Zeilen möglich. Genaue Beschreibung im Manual.

### 2.1.3 Neuer BASIC-Compiler 3.5

Das System enthält einen neuen, der ANSI-Norm angepassten BASIC-Compiler. Er ist teilweise inkompatibel zu BASIC 3.4. Einzelheiten im Manual oder über die Beratung.

### 2.1.4 Entfallende Kommandos

Die (undokumentierten) Kommandos

RWF

RTF

sind nicht mehr vorhanden.

Die Kommandos

SETID und

DISPOSE

sollten nicht mehr verwendet werden, da sie mit der nächsten Systemänderung entfallen. Statt dessen ist das Kommando

ROUTE

zu verwenden.

## 2.2 SUBMIT-Kommando und Output-Listen im Dialog

Zwei neue Kommandos QGET und DROP und Änderungen des SUBMIT-Kommandos erlauben jetzt

- beim Erstellen eines Batch-Jobs mit dem SUBMIT-Kommando anzugeben, daß dessen Output-Liste weder gedruckt werden noch verschwinden soll, sondern festgehalten werden soll.
- einen noch nicht gedruckten Output-File eines Batch-Jobs zum Local-File einer Dialog-Sitzung (des gleichen Benutzers) zu machen.
- einen laufenden Batch-Job vom Terminal aus abzuberechnen.
- einen Batch-Job, der sich noch in der Input-Queue befindet, vom Terminal aus zu löschen.
- im SUBMIT-Kommando die RJE-Station anzugeben, an der die Liste gedruckt werden soll.

### 2.2.1 SUBMIT-Kommando

Wie in der letzten BI beschrieben, kann mit einem im Job befindlichen ROUTE-Kommando angegeben werden, an welchem Drucker die Output-Liste (und der Dayfile) gedruckt werden soll. Enthält der Job-File kein ROUTE-Statement wird, je nach den im SUBMIT-Kommando angegebenen Parametern ein bestimmtes Routing angenommen:

SUBMIT-Kommando

implizites ROUTE-Kommando

SUBMIT, lfn, B

ROUTE, OUTPUT, DEF, DC=PR, ID=0.

die Liste wird im Rechnerraum gedruckt.

SUBMIT, lfn, N

ROUTE, OUTPUT, DEF, DC=SC.

die Liste verschwindet.



SUBMIT,lfn,T (Neu !)                   ROUTE,OUTPUT,DC=PR,ID=67.  
                   die Liste wird festgehalten.  
 SUBMIT,lfn,E=rjename (Neu !)   ROUTE,OUTPUT,DC=PR,UN=rjename.  
                   die Liste wird an der angegebenen RJE-Station gedruckt .  
 SUBMIT,lfn,E  
                   diese Form ist jetzt nicht mehr zulässig  
 SUBMIT,lfn       wirkt wie       (Änderung der Voreinstellung !)  
 SUBMIT,lfn,T  
                   die Liste wird festgehalten

Als Antwort des SUBMIT-Kommandos erscheint die Meldung:

hh.mm.ss.aaaabbb

dabei ist hh.mm.ss   die Uhrzeit  
                   aaaa       der sog. "User-Hash-Code" des erzeugten Batch-Jobs, dieser ist eindeutig einer Benutzernummer zugeordnet, hier derjenigen des (Dialog-)Jobs aus dem das SUBMIT-Kommando kommt.  
                   bbb       der "System-Job-Sequence-Name"

Die letzten drei Buchstaben bbb sind die Identifikation des soeben abgeschickten Jobs, die man angeben muss, wenn man ihn löschen, abbrechen oder auf seine Liste zugreifen will.

Die ersten vier Buchstaben aaaa stellen einen "Eigentümer" des Jobs dar, nur dieser darf etwas mit dem Job oder seiner Liste machen. Wird ein Job erzeugt, der in seinem Job-File einen anderen User-Name enthält als der Job, in dem das SUBMIT-Kommando gegeben wird, so gilt letzterer als Eigentümer.

Als rjename sind zulässig:

BAMRJE für Bamberg  
 BTHRJE für Bayreuth  
 COBRJE für Coburg  
 ERIRJE für Erlangen, Bismarckstrasse  
 ERSRJE für Erlangen, Tandemlabor  
 NFHRJE für FHS Nürnberg  
 NFIRJE für SFZ Nürnberg, Findelgasse  
 NTURJE für WISO-Nürnberg, Tuchergelände

Weitere Informationen zum ROUTE-Kommando finden Sie in der BI 22.

### 2.2.2 Holen der Liste

Mit dem folgenden Kommando kann eine Output-Liste (incl. Dayfile) aus der Drucker- (oder Stanzer-) Warteschlange entfernt und zu einem Local-File des Terminal-Dialogs gemacht werden. Voraussetzung ist, daß der User-Name des jetzigen Jobs als "Eigentümer" der Liste gilt, d.h. daß sie mit dem User-Name des Jobs, in dem das SUBMIT-Kommando gegeben wurde, bzw. dem des Batch-Jobs übereinstimmt.

QGET,bbb,DC=PR

QGET,bbb

dabei ist bbb

qq

Job-Sequence-Name (vom SUBMIT)

PR : Drucker-Warteschlange (Vorein-

stellung)

PU : Stanzer-Warteschlange

Der Local-File-Name, den der geholte Output-File bekommt, ist der vollständige 7-buchstabige System-Jobname aaaabbbb. Es empfiehlt sich, mit RENAME,xyz=aaaabbbb diesen File umzubenennen, wenn man mehr mit ihm machen will.

Wenn ein Job mehrere Drucker-Listen abschickt, die alle mit QGET geholt werden sollen, muss nach jedem QGET der Local-File entfernt oder umbenannt werden, da es sonst Namenskonflikte gibt.

### 2.2.3 Löschen/Abbrechen

Mit folgendem Kommando wird

- ein Input-File gelöscht
- ein laufender Job abgebrochen oder
- eine Output-Liste gelöscht.

Voraussetzung ist auch hier, daß der Terminal-User als "Eigentümer" gilt.

DROP,bbb,DC=qq

DROP,bbb

dabei ist bbb

qq

Job-Sequence-Name (vom SUBMIT)

ALL

Löschen oder Abbrechen, je nachdem, wo der Job gefunden wird.

EX

Abbrechen eines laufenden Jobs (Voreinstellung)

IN

Löschen eines Input-File

PR

Löschen eines Drucker-Files

PU

Löschen eines Stanz-Files

Wird ein mit SUBMIT,T abgeschickter Job mit DROP,bbb abgebrochen, kann sein bis dahin entstandenes Output-File und Dayfile mit QGET, bbb in den Dialog geholt werden.

### 2.2.4 Information über festgehaltene Listen

Das Kommando

STATUS,JN

informiert wie bisher über den Status von Jobs, die sich in der Eingabe-Warteschlange befinden, gerade rechnen oder deren Liste noch gedruckt werden soll.

Eine festgehaltenen Liste wird durch die Meldung

aaaabbbb IN TERMINAL QUEUE

gekennzeichnet.

Ein Benutzer, der das SUBMIT-Kommando verwendet, sollte sich ab und zu mit diesem Kommando vergewissern, daß keine festgehaltene Liste mehr vorhanden ist, zumal der Voreinstellungs-Wert im SUBMIT-Kom-



mando geändert ist.

Das RRZE ist aus Platzgründen gezwungen Listen, die älter als vier Tage sind, zu löschen.

### 2.2.5 Zusammenfassung

SUBMIT,lfn	oder	
SUBMIT,lfn,T		Liste wird festgehalten (Neu !)
SUBMIT,lfn,B		Liste wird im Rechnerraum gedruckt
SUBMIT,lfn,N		Liste verschwindet
SUBMIT,lfn,E=rjename		Liste wird an einer RJE-Station gedruckt (Neu !)
QGET,bbb	oder	
QGET,bbb,DC=PR		Output-File wird zum Local-File
QGET,bbb,DC=PU		Punch-File wird zum Local-File
DROP,bbb	oder	
DROP,bbb,DC=EX		laufender Job wird abgebrochen
DROP,bbb,DC=IN		Input-File wird gelöscht
DROP,bbb,DC=PR		Output-File wird gelöscht
DROP,bbb,DC=PU		Punch-File wird gelöscht
DROP,bbb,DC=ALL		Löschen oder Abbruch, je nach Jobzustand
STATUS,JN		Abfrage über Job- und Listen-Zustand

Kommerzielle Datenverarbeitung:

HERR/FRAU

RECHENZENTRUM UNIV  
MARTENSSTR 1  
8520 ERLANGEN

SEHR GEEHRTE FAMILIE RECHENZENTRUM, ERLANGEN!



ADA



# Die Programmiersprache ADA

Dieser Bericht wurde der BI108 des Regionalen Rechenzentrums Niedersachsen entnommen:

Wir haben Sie bereits in BI 77 und BI 82 über den jeweiligen Stand des DoD-Projektes (DoD= Department of Defense, U.S. Verteidigungsministerium) zur Entwicklung einer neuen problemorientierten Programmiersprache unterrichtet. Nach drei Jahre langen Bemühungen ist der Entwurf dieser Sprache, ADA genannt, abgeschlossen worden, so daß die Standardisierung durch ANSI und ISO inzwischen in die Wege geleitet werden konnte.

Die Programmiersprache ADA ist nicht in erster Linie für den kommerziellen Anwendungsbereich (wie etwa COBOL) oder speziell für den mathematisch-naturwissenschaftlichen Anwendungsbereich (wie etwa FORTRAN oder ALGOL 60 oder PASCAL oder - mit Einschränkung - etwa ALGOL 68) entwickelt worden. Vielmehr ist sie für die Programmierung sogenannter 'integrierter Systeme' oder 'eingebetteter Systeme' gedacht. Hierbei handelt es sich um Systeme, die aus Hardware, Software und technischen Prozessen bestehen und die in der Regel in größere Systeme eingebettet sind.

Beispiele: Ein Raumfahrzeug, das einen Rechner zur Überwachung der Navigation und des Fluges enthält; rechnergesteuerte Feuerleitsysteme, taktische hochkritische Waffensysteme; Kommunikationssysteme usw. Typisch ist der Einsatz von rechnerunterstützten eingebetteten Systemen im Zusammenhang mit Echtzeitanwendung.

Nun ist es für den RRZN-Benutzer sicher nicht so wichtig, daß er beispielsweise ein rechnerunterstütztes Raketenabwehrsystem mit ADA programmieren kann, als vielmehr, daß er mindestens alle diejenigen Probleme auch mit ADA lösen kann, die er bisher mit Hilfe von Sprachen wie ALGOL 60, PASCAL oder FORTRAN gelöst hat.

ADA zeichnet sich durch folgende Eigenschaften bzw. die Unterstützung folgender Konzepte aus, die zu einem großen Teil aus so bekannten Sprachen wie PASCAL, PL/1, ALGOL 68 u.a.m. stammen:

## Strenge Typisierung

Alle Datenelemente müssen vereinbart werden, so daß in jeder Situation, in der Datentypen miteinander in Beziehung treten (wie z.B. in arithmetischen Ausdrücken oder bei Parameteraustausch), die Gültigkeit dieser Beziehungen überprüft werden kann.

## Datentypen

- Datenstrukturen (Skalar, Feld, Struktur)
- Vom Programmierer definierte Datentypen
- Abgeleitete Datentypen

## Datenabstraktion

Private Datentypen, um Informationen über Implementationsdetails von den abstrakten Eigenschaften von Daten zu trennen (um sie zu verstecken). Diese Eigenschaft ist von großer Bedeutung für den Wiedergebrauch und Austausch von Software.

## Zeichendaten

ADA unterstützt den vollen ASCII-Zeichensatz (128 Zeichen) und Zeichenreihen.

## Zugriffstyp

ADA unterstützt die Handhabung von Referenzen und Pointer.

## Genauigkeit

Der Programmierer spezifiziert die gewünschte Genauigkeit beispielsweise eines Datentyps in maschinenunabhängiger Art und Weise (z.B. type REAL is digits 10; oder type LONG REAL is digits 14).

### Operatoren und Funktionen

Können "überladen" werden, d.h. sie können u. U. auf verschiedene Datentypen angewandt werden.

### Parameter

Unterprogramme können mit Standardparametern (Anfangswerten) versehen werden, die verwendet werden, wenn kein entsprechender Aktualparameter angegeben ist.

### Feldparameter

Formale Feldparameter dürfen beliebige Grenzen haben.

### Verallgemeinerte Programmeinheiten

Unterprogramme und 'Pakete' können hinsichtlich der Typen der Parameter parametrisiert werden. Diese Eigenschaft ist besonders wertvoll für Programmbibliotheken. Pakete ermöglichen die Zusammenfassung gemeinsamer Daten und Typen, die Sammlung zusammengehörender Unterprogramme oder die Zusammenfassung von Typen mit zugehörigen Operationen.

### Reservierte Wörter

Es gibt insgesamt 62 solcher Wörter, die reserviert sind und nicht redefiniert werden können.

### Ausnahmebehandlung

Der Programmierer hat Kontrolle über die Reaktion in Ausnahmesituationen.

### Parallelverarbeitung

ADA ist die erste Sprache für universelle Anwendungen, die ein Tasking-Konzept unterstützt.

### Ein-/Ausgabe

Einheitliche Ein-/Ausgabe und Dateiverwaltung. Die Ein-/Ausgabe ist nicht eigentlich Teil der Sprache, aber in ADA geschrieben und als Paket zur Verfügung gestellt.

### Compilation

Separate Compilation von Programmeinheiten. Compilationsdirektiven (Pragmas).

### Programmierungsmethodik

- Unterstützung moderner 'geklammerter' Kontrollstrukturen
- Datenabstraktion
- Programmstrukturierung mittels Programmeinheiten wie Unterprogramme und Pakete.

Zur Zeit gibt es weltweit eine Reihe von Aktivitäten im Zusammenhang mit der Realisierung von ADA. Dazu gehört der Bau von Übersetzern, von Cross-Compilern, von ADA-PASCAL-Übersetzern und von sogenannten Programmierungsumgebungen. Programmierungsumgebungen sind Softwarewerkzeugkisten für den sinnvollen Einsatz von ADA.

Die EG-Kommission hat beispielsweise gerade zwei Aufträge im Gesamtvolumen von fast 17 Mio. DM an europäische Firmen vergeben mit dem Ziel, ADA auf einer breiten Palette von Mainframe-Anlagen und Minicomputern bzw. in verteilten Systemen lauffähig zu machen. Die Firma Intel hat für Ende 1981 die Produktion eines 32-Bit-Mikro-Mainframe-Computers mit ADA als 'Maschinensprache' angekündigt. Die Firma CDC hat zur Zeit keinerlei Pläne hinsichtlich ADA (lt. Auskunft vom September 1980).

Die strikte Einhaltung der ADA-Spezifikation wird vom DoD überwacht.

ADA hat in den vergangenen Jahren eine lebhaft diskutierte Diskussion über die Brauchbarkeit und über den richtigen Weg zur endgültigen Definition dieser Sprache ausgelöst. Das Spektrum der Meinungen reicht etwa von "Irrweg", "ein zweites PL/I" oder "Monstrum" bis hin zu "die Sprache der Zukunft". In der Tat ist es so, daß ADA zwar wie keine andere Sprache die modernsten Forschungsergebnisse und Konzepte realisiert und dem Programmierer den Einsatz moderner Programmierungsmethoden erlaubt. Eine derartige Sprache, die wie ADA entstanden ist, muß aber zwangsläufig einen Um-



fang haben, der den FORTRAN-Programmierer erschrecken wird. Die ursprüngliche Idee einer schlichten und trotzdem wirkungsvollen Sprache (Vorbild PASCAL) ist völlig verlorengegangen. ADA ist auch in dieser Hinsicht eine Sprache für den professionellen Anwender.

Experten vermuten, daß etwa in der Mitte dieses Jahrzehnts die Voraussetzungen dafür geschaffen sind, daß sich ADA als universelle Sprache weltweit durchsetzt. Weiter wird spekuliert, daß ADA etwa ab Mitte der 90er Jahre der Sprache FORTRAN hinsichtlich der Häufigkeit des Gebrauchs den Rang abgelaufen haben wird.

Ob solche Spekulationen realistisch sind oder eher einem Wunschdenken entsprechen, das kann man heute wohl kaum richtig beurteilen.

Quellen: (1.) SOFTWARE-PRACTICE AND EXPERIENCE, Vol. 10, 851-887 (1980)  
 (2.) DATAMATION, July 1979, 146-150  
 (3.) Computerwoche, 12. Dezember 1980, Seite 10  
 (4.) Computerwoche, 30. Januar 1981, Seite 1

Wer sich genauer informieren will, dem empfehlen wir (1.).

Wer an Details interessiert ist, der kann sich ein Reference Manual besorgen: "Reference Manual for the ADA Programming Language, U.S. DoD, Juli 1980", - zu erhalten über Dr. Clausen, IABG, Postabholfach, 8012 Ottobrunn.

---

## **Letzte Meldung:**

---

**G**roßen Ärger und Entlassungen gab's in einem Laboratorium für die Entwicklung moderner Waffen im US-Staat Neumexiko. Eine Kontrollkommission untersuchte den als „Geheimnisträger“ ersten Ranges geltenden Labor-Computer und stellte fest: Das Personal hatte das Elektronenhirn mit Witzen und Limericks gefüttert, 247 elektronische Gesellschaftsspiele waren eingespeist, sogar für illegale Buchmachergeschäfte hatte man ihn mißbraucht.

# ADA:

## Ein Denkmal für die Lady

Es geschah zu der Zeit, als zwischen Fürth und Nürnberg die erste deutsche Dampflokomotive auf die Schienen gesetzt wurde und in England die jugendliche Königin Victoria den Thron bestieg - in jener guten alten Zeit, als man an den europäischen Stammtischen noch nicht mit den Benzinpreisen haderte, sondern allenfalls auf den Metternich sauer war (nicht auf den Sekt, auf den Fürsten) - damals also sprach der britische Mathematiker Charles Babbage (1792-1871) zu seiner freiberuflichen Mitarbeiterin: "Ada", sprach er (sinngemäß), "dafür, daß dein Vater ein Lord war und ein Poet dazu, beweist du in der Tat ein enormes Verständnis für mein neues Computersystem."

Das heißt, "Computersystem" wird Babbage nicht gesagt haben, eher "analytical engine", doch man sieht heute den solchermaßen betitelten eisenwichtigen Kalkulator als den Vorläufer moderner DV-Anlagen an.

Wer jedoch weiß schon etwas von der Assistentin des berühmten Cambridge-Professors? Welches Handbuch erwähnt die frühenglische Emanze Ada Auguste Gräfin Lovelace (1815-1852), die Tochter des Dichters Lord Byron (1788 bis 1824)?

Als ihr Vater starb, war Ada acht Jahre alt. Die Eltern hatten getrennt gelebt; das Poetenkind wuchs unter dem autoritären Regiment der Mutter auf, die für neue Erziehungsmethoden schwärmte und diesem Hobby am lebenden Objekt (Ada) frönte. Das geduldige Mädchen ließ es geschehen, lernte Mathematik, Geometrie, Musik, Geographie und Französisch - bis sie dann eines Tages heiratete (1835): einen standesgemäßen Lord, den die Königin während einer summarischen Beförderungszeremonie (Anlaß: die Krönung) zum Earl of Lovelace ernannte.

In den höheren Gesellschaftskreisen Großbritanniens muß in jenen Jahren die angewandte Naturwissenschaft "in" gewesen sein. Jedenfalls war Charles Babbage im Gespräch. Der Professor hatte eine sogenannte "difference engine" gebaut und entwickelte anschließend ein innovatives Konzept für die "analytical engine". Jedermann war beeindruckt, zumal niemand recht wußte, wofür das eine oder andere gut sein sollte, denn Babbage hatte nie eine zusammenhängende Dokumentation über sein System vorgelegt. Schreiben lag ihm nicht. Das besorgten andere für ihn. Einer seiner Bewunderer war der italienische Militär-Ingenieur L.F. Menabrea. Dieser hatte Vorträge des englischen Erfinders gehört und einen Bericht über die erstaunliche Rechenmaschine ausgearbeitet (1842). Leider war Menabreas Manual nur in Französisch verfügbar - und das ist nun der Moment, wo Lady Lovelace in die Computergeschichte eintritt, denn sie übersetzte die Beschreibung.

Als Babbage die englische Version las, erkannte er sofort das Talent der Übersetzerin. Er empfahl der Lady, selbst etwas über den Automaten zu verfassen. Das tat Ada. Ihr mathematischer Durchblick ging dabei so weit, daß sie Ungereimtheiten in Babbages Ablaufschema zur Berechnung Bernoullischer Zahlen entdeckte - es war dies die erste verbürgte Fehlerfindung in einem Computerprogramm.

Charles Babbage lobte die kluge junge Dame, die offenbar besser als die Kollegen die Bedeutung der "analytical engine" begriff. Das für damalige Vorstellungen recht ungleiche Wissenschaftler-Paar arbeitet von nun an eng zusammen, wenn es darum ging, die Grundlagen der Systemprogrammierung transparent zu machen. Auf einigen Gebieten der Umsetzung von Formeln in



Funktionsschritte - heute würde man "Codierung" sagen -, stellte Ada Gräfin Lovelace ihren Lehrmeister in den Schatten seiner eigenen Maschine. Die Lady war zweifellos eine Sensation in der britischen Society und geriet zum Thema unzähliger vornehmer Tee-Stunden. Auch jetzt noch geht von den Notizen der ersten Daten-Dame eine seltsame Faszination aus, denn die Eleganz ihrer Programmierung würde mancher EDV-Anlage der achtziger Jahre (des Zwanzigsten Jahrhunderts) zur Glorie gereichen.

Ihr Know-how artikuliert Lady Lovelace geschickt und anschaulich - bisweilen untermischt mit prophetischen Zwischentönen: "Die Analysemaschine", vermerkte sie einmal, "erhebt keinen Anspruch darauf, etwas Originäres zu schaffen; sie führt das aus, was wir, aus unserem Wissen heraus, ihr vorschreiben."

Lord Byrons Wunschkind reflektierte auch über das Problem der bedingten Verzweigung und erläuterte die Prozedur. Sie erfand das Zählregister für iterative Abläufe, konzipierte ein binär-arithmetisches Rechenverfahren und erträumte sich programmiertechnische Kniffe, die vor dem Siegeszug der modernen EDV eigentlich kaum denkbar gewesen sein konnten.

Um das Genie der Engländerin gebührend zu beurteilen, muß man sich vergegenwärtigen, daß das Ursprungsmodell des 1944 konstruierten Automatic Sequence Controlled Calculator in seinem Befehlssatz noch keine Sprung-Instruktion enthielt; die wurde erst später hinzugefügt, obwohl die Methode bereits seit hundert Jahren in Lady Lovelace' nachgelassenen Schriften stand.

Würde die Automaten-Lady aus der Biedermeier-Epoche heute leben - ein Nobelpreis mit Kranz und Schleife wäre ihr sicher, wenngleich die Dame in ihrer Karriere auch herbe Rückschläge erfahren hatte. Angesteckt von der Wettleidenschaft ihrer Inselvolksgeossen und mit viel Sinn für das praktisch Nutzvolle einer abstrakten Wissenschaft begabt, ertüftelte Ada ein "todsicheres" System zur Vorherbestimmung der Sieger beim Pferderennen, womit sie dann prompt in ein finanzielles Fiasko schlitterte. Chronisten vermelden, daß sie durch diese Pleite nicht nur das Familienvermögen verlor, sondern auch ihren Ehemann vergraulte und, illiquide geworden, kurz vor ihrem Tod ihren gesamten Schmuck versetzen mußte.

Indes: Ada Augusta Countess of Lovelace kam posthum zu Ehren, als ihr die computerbefaßte Fachwelt 1979 ein symbolisches Denkmal setzte: Die "Green Language", so der bis dahin gängige Arbeitstitel, wurde in "Ada" umbenannt. Es handelt sich um eine "high order language" für Real-Time-Programmierung, die das amerikanische Verteidigungsministerium in Auftrag gab und die eine von Jean Ichbiah geleitete Forschungsgruppe der Cii Honeywell Bull anwendungsreif entwickelte. Der Präsident der Französischen Republik ehrte den Team-Leiter, indem er ihm das Kreuz der Ehrenlegion verlieh.

Wenn also das US-Militär mit seiner umfänglichen EDV echtzeitlich "Ada" redet - eine französische Sprache mit englischem Wortschatz -, dann schwingt bei jeder Instruktion ein kleiner Anerkennungsimpuls mit, ein Erinnerungs-Bit an die wenig bekannte Tatsache, daß der erste Programmierer der Welt eine Programmiererin war.

\* \* \*

(Entnommen aus "Jenseits von Bit und Byte" mit freundlicher Genehmigung des Autors Heinzgünther Klaus, Pressereferent der Honeywell Bull AG.)

"Permanent-File"-Unterprogramme für FORTRAN (CYBER)Permanent File Subroutines

NOS provides a set of FTN callable subroutines that allow the user to issue permanent file commands from FTN4, FTN5, or SYMPL.

## NOS PF ROUTINES

The following set of CALL's are provided by NOS:

```
CALL PF("APPEND",lfn,pfn,key(1),val(1),...,key(n),val(n))
CALL PF("ATTACH",lfn,pfn,key(1),val(1),...,key(n),val(n))
CALL PF("CHANGE",nfn,ofn,key(1),val(1),
CALL PF("DEFINE",lfn,pfn,key(1),val(1),...,key(n),val(n))
CALL PF("GET",lfn,pfn,key(1),val(1),...,key(n),val(n))
CALL PF("PERMIT",pfn,key(1),val(1),...,key(n),val(n))
CALL PF("PURGE",pfn,key(1),val(1),...,key(n),val(n))
CALL PF("REPLACE",lfn,pfn,key(1),val(1),...,key(n),val(n))
CALL PF("SAVE",lfn,pfn,key(1),val(1),...,key(n),val(n))
```

## PARAMETER DEFINITIONS

The parameters to these routines consist of three positional mandatory parameters, the function to be performed, lfn, and pfn followed by a set of optional order independent paired keyword and value parameters. The lfn may be of type integer or type character. If lfn is an integer, the lfn used will be TAPEnnn where nnn is the integer value converted to characters (range 0 to 999). The pfn is of type character. The first parameter of each pair is a character constant, character variable or character expression, specifying the keyword; and the second parameter is the value for that keyword. All parameters, keywords and allowed values are specified by the following table:

<u>Keyword</u>	<u>Value</u>
"BR"	"Y","N","MD"
"CT"	"P","PRIVATE","S","SPRIV","PU","PUBLIC"
"M"	"WRITE","W","MODIFY","M","APPEND","A", "READ","R","READMD","RM","READAP","RA", "EXECUTE","E","NULL","N"
"NA"	Any single word value (ignored by PF).
"NONE"	Any single word value (ignored by PF).
"PN"	Character string of 1-7 characters.
"PR"	"M","N"
"PW"	Character string of 1-7 characters.
"R"	"DE","DI","DJ","DK","DL","DM","DP","DQ", "MD"
"RC"	Integer variable.
"RRC"	Real variable.



"S"	Character string of 1-7 digits.
"SR"	"CE","MR"
"SS"	"NULL","BASIC","FORTRAN","FTNTS","EXECUTE","BATCH"
"UCW"	Integer variable.
"UN"	Character string of 1-7 characters.
lfn	Valid file name (1-7 chars) or FORTRAN unit # (integer).
nfn	Valid new permanent file name (1-7 characters).
ofn	Valid old permanent file name (1-7 characters).
pfn	Valid permanent file name (1-7 characters).

See the NOS Reference Manual for details on the meaning and allowed values for parameters.

"NONE" is used to nullify a particular keyword=value pair option if it is not to be used on the current call. The PF routine will ignore the parameter pair and proceed to the next pair.

#### PARAMETER SEMANTICS

Parameters permitted on the various permanent file functions are the same as those allowed on the corresponding NOS permanent file control card call as specified in the NOS Reference Manual. If a parameter is omitted, the default will be the same as for a control card call. Parameters have the same meaning as on control card calls except for "NA", "NONE", "RC", "RRC", and "SR".

"SR" allows the following special requests: "CE" clears the file error code on a CHANGE, and "MR" causes master device residency on a DEFINE.

String values representing numbers (the S parameter) may have leading blanks before the first digit of the numeric value.

If pfn is all blanks or zero, then pfn=lfn is assumed. Inversely, if lfn is all blanks or zero, then lfn=pfn is assumed.

UCW allows specification or return of the user control word. The contents of the user control word are returned on a GET or ATTACH request and are set on a DEFINE and SAVE. UCW is assumed to be a 59-bit integer value.

#### ERROR PROCESSING

The combination of the RC (return code) and NA (no abort) selects one of three response modes from the PF subroutines. RRC has the same meaning as RC except that the value returned is real.

- 1) Neither NA nor RC (or RRC) are specified = If the function is successful, the program proceeds as usual. If the function fails, then the standard PF error message is written to the user dayfile (or terminal) and the program aborts.
- 2) RC (or RRC) specified (no NA) = If the function fails, then the appropriate return code is returned to the RC val (n) parameter. A positive non-zero value is an error code as specified in the NOS Reference Manual. A zero value indicates successful completion. A -1 value indicates an error in the arguments of the call. If the file is unavailable on an ATTACH (some other job is updating the file, etc.), then the job will be swapped out and swapped back in when the file is available.
- 3) RC (or RRC) and NA specified = Same as case 2 except in the unavailable case. In that case, control is returned immediately with the appropriate return code in the RC val(n) parameter. This gives the user maximum control over PF functions.

#### BUFFER FLUSHING

If a program creates a local file and then issues the PF functions SAVE, REPLACE, or APPEND, the buffer for the local file lfn must first be flushed. This may be forced via a REWIND,ENDFILE, or CLOSE.

#### SUBROUTINE RESIDENCE

The PF subroutine resides in system library SYSLIB.



Regionales Rechenzentrum Erlangen  
Vorlesungsankündigung  
SS 1981

10598 Einführung in die Benutzung der Rechenanlagen TR440 und Cyber  
Woitok/Schönhut  
Vorlesung 3-stündig  
Anmeldung: In der Aufsicht des RRZE  
Zeit: 21.04. - 27.04. 1981  
Raum: 2.037, RZ-Gebäude  
Beginn: 21.04. 1981 9.15 Uhr  
Teilnehmer: Hörer aller Fakultäten  
Schein: nein

10599 Einführung in die Benutzung von SPSS - Statistik Programmsystem für die Sozialwissenschaften  
Cramer  
Vorlesung mit Übungen 3-stündig  
Anmeldung: In der Aufsicht des RRZE  
Zeit: 27.04. - 30.04. 1981  
Raum: 2.037, RZ-Gebäude  
Beginn: 27.04. 1981, 9.15 Uhr  
Teilnehmer: Hörer aller Fakultäten  
Schein: nein

Hinweis: Diese Veranstaltungen wurden schon im Vorlesungsverzeichnis WS 1980/81 angekündigt.

10484 Mikrorechner-Kolloquium  
Wolf/Holleczech  
Kolloquium 2-stündig  
Zeit: Dienstag 16.15 -17.45 Uhr  
Raum: H4, RZ-Gebäude

10473 Rechnergestützte Einführung in die Programmiersprache ALGOL  
Wolf  
Sonstige Lehrveranstaltung 4-stündig  
Anmeldung: In der Aufsicht des RRZE  
Dauer: Mai bis Juli 1981  
Einführung: Wird durch besonderen Aushang bekanntgegeben  
Raum: 2.037 RZ-Gebäude  
Teilnehmer: Hörer aller Fakultäten  
Schein: ja, durch Abgabe gelöster Übungsaufgaben und Abschlußklausur; zählt nicht als Ersatz für Problemorientierte Programmierung für Informatiker

10483 Betrieb von Großrechenanlagen

Müller, O.

Vorlesung mit Übungen 4-stündig

Anmeldung: In der Aufsicht des RRZE

Zeit: Vorlesung: Dienstag 14-16 Uhr  
Mittwoch 16-18 Uhr

Übungen nach Vereinbarung

Raum: H4, RZ-Gebäude

Beginn: 05.05. 1981 14.15 Uhr

Teilnehmer: Hörer aller Fakultäten

Schein: ja, durch praktische Arbeiten und Abschlußklausur

10482 Einführung in die Programmierung (FORTRAN 77)

Henke

Vorlesung mit Übungen 4-stündig

Anmeldung: In der Aufsicht des RRZE

Zeit: Vorlesung: Mittwoch 14-16 Uhr  
Übungen nach Vereinbarung

Raum: H8, Hörsaal-Gebäude

Beginn: 06.05. 1981 14.15 Uhr

Teilnehmer: Hörer aller Fakultäten

Schein: ja, durch Abgabe gelöster Übungsaufgaben

10481 Einführung in die Programmierung (FORTRAN 77)

Abel

Vorlesung mit Übungen 4-stündig

Anmeldung: In der Aufsicht des RRZE

Zeit: 03.08. - 21.08. 1981

Raum: H4, RZ-Gebäude

Beginn: 03.08. 1981 9.15 Uhr

Teilnehmer: Chemieingenieure werden bevorzugt aufgenommen

Schein: ja, durch Abgabe gelöster Übungsaufgaben

10472 Rechnergestützte Einführung in die Programmiersprache ALGOL

Wolf

Sonstige Lehrveranstaltung 4-stündig

Anmeldung: In der Aufsicht des RRZE

Dauer: August 1981 bis Oktober 1981

Einführung: Wird durch besonderen Aushang bekanntgegeben

Raum: 2.037 RZ-Gebäude

Teilnehmer: Hörer aller Fakultäten

Schein: ja, durch Abgabe gelöster Übungsaufgaben und Abschlußklausur; zählt nicht als Ersatz für Problemorientierte Programmierung für Informatiker



- 10474 Einführung in die Benutzung der Rechenanlagen TR440 und Cyber  
 Woitok, Cramer  
 Vorlesung 3-stündig  
 Anmeldung: In der Aufsicht des RRZE  
 Zeit: 12.10. -16.10.1981  
 Raum: 2.037, RZ-Gebäude  
 Beginn: 12.10. 1981 9.15 Uhr  
 Teilnehmer: Hörer aller Fakultäten  
 Schein: nein
- 10478 Das Erlanger Grafik-System EGS - Eine Einführung  
 Schönhut  
 Vorlesung mit Übungen 2-stündig  
 Anmeldung: In der Aufsicht des RRZE  
 Zeit: 19.10. - 20.10. 1981  
 Raum: 2.038, RZ-Gebäude  
 Beginn: 19.10. 1981 9.15 Uhr  
 Teilnehmer: Hörer aller Fakultäten  
 Schein: nein
- 10479 Einführung in die Benutzung von SPSS - Statistikprogrammssystem für die Sozialwissenschaften  
 Cramer  
 Vorlesung mit Übungen 3-stündig  
 Anmeldung: In der Aufsicht des RRZE  
 Zeit: 19.10. - 23.10. 1981  
 Raum: 2.037, RZ-Gebäude  
 Beginn: 19.10. 1981 9.15 Uhr  
 Teilnehmer: Hörer aller Fakultäten  
 Schein: nein
- 10476 Programmierpraktikum  
 Abel  
 Praktikum 2-stündig  
 Anmeldung: In der Aufsicht des RRZE  
 Einführung: Wird durch besonderen Aushang bekanntgegeben  
 Zeit: August bis Oktober 1981  
 Raum: 2.037, RZ-Gebäude  
 Teilnehmer: Hörer aller Fakultäten  
 Schein: nein  
 Hinweis: Gebühren für anfallende Rechenzeitkosten müssen von den Teilnehmern im voraus entrichtet werden. Die Mindestgebühr beträgt DM 10,--.

10475 Einführung in die Programmierung (PASCAL)

Abel

Vorlesung mit Übungen 4-stündig

Anmeldung: Im Mathematischen Institut, Zimmer 7

Zeit: Vorlesung: Freitag 14-16 Uhr  
Übung nach Vereinbarung

Raum: Hörsaal des Mathematischen Institutes

Beginn: 08.05. 1981 14.15 Uhr

Teilnehmer: Mathematiker werden bevorzugt aufgenommen

Schein: ja, durch Abgabe gelöster Übungsaufgaben

10477 Programmierpraktikum

Henke

Praktikum 2-stündig

Anmeldung: In der Aufsicht des RRZE

Einführung: Freitag, 08.05. 15.15 Uhr

Zeit: Mai bis Juli 1981

Raum: 2.037, RZ-Gebäude

Teilnehmer: Hörer aller Fakultäten

Schein: nein

Hinweis: Gebühren für anfallende Rechenzeitkosten müssen von den Teilnehmern im voraus entrichtet werden. Die Mindestgebühr beträgt DM 10,--.

10480 Einführung in die Programmierung (PASCAL)

Henke

Vorlesung mit Übungen 4-stündig

Anmeldung: In der Aufsicht des RRZE

Zeit: 19.10. - 30.10. 1981

Raum: H4, RZ-Gebäude

Beginn: 19.10. 1981 9.15 Uhr

Teilnehmer: Hörer aller Fakultäten

Schein: ja, durch Abgabe gelöster Übungsaufgaben



Lehrstuhl für Psychologie  
(insbes. Wirtschafts- und Sozialpsychologie)  
im  
Sozialwissenschaftlichen Institut  
der  
Friedrich-Alexander-Universität Erlangen-Nürnberg

— Prof. Dr. J. Franke —

Lange Gasse 20  
Postfach 3931  
D-8500 Nürnberg  
Ruf (0911) 53 02-1  
Durchwahl 53 02 259

31. März 1981

Arbeitsgemeinschaft: Computerunterstützte empirische  
Auswertungsstrategien - Insbesondere: Anwendung  
des Programmpaketes SPSS - Mit praktischen  
Übungen

Im Sommersemester 1981 wird in Zusammenarbeit mit dem Regionalen Rechenzentrum Erlangen die obengenannte Arbeitsgemeinschaft veranstaltet. Ziel dieser Arbeitsgemeinschaft ist es, den Teilnehmern Kenntnisse zu vermitteln, die eine weitgehend selbständige Auswertung empirischer Untersuchungen ermöglichen.

Gegenstand der Arbeitsgemeinschaft sind folgende Bereiche:

- Datenprüfung und Fehlersuche
- Prüfung der Verteilungen von Variablen  
(z.B. Prüfung auf Normalverteilung)
- Item- und Skalenanalyse
- Voraussetzungen, Anwendung und Interpretation von  
Korrelations- und Regressionsanalyse
- Anwendung und Probleme der Faktorenanalyse
- Voraussetzungen, Anwendung und Interpretation  
varianzanalytischer Verfahren

Diese Arbeitsgemeinschaft ist insbesondere für Studenten gedacht, die bereits Grundkenntnisse des Programmpaketes SPSS besitzen.

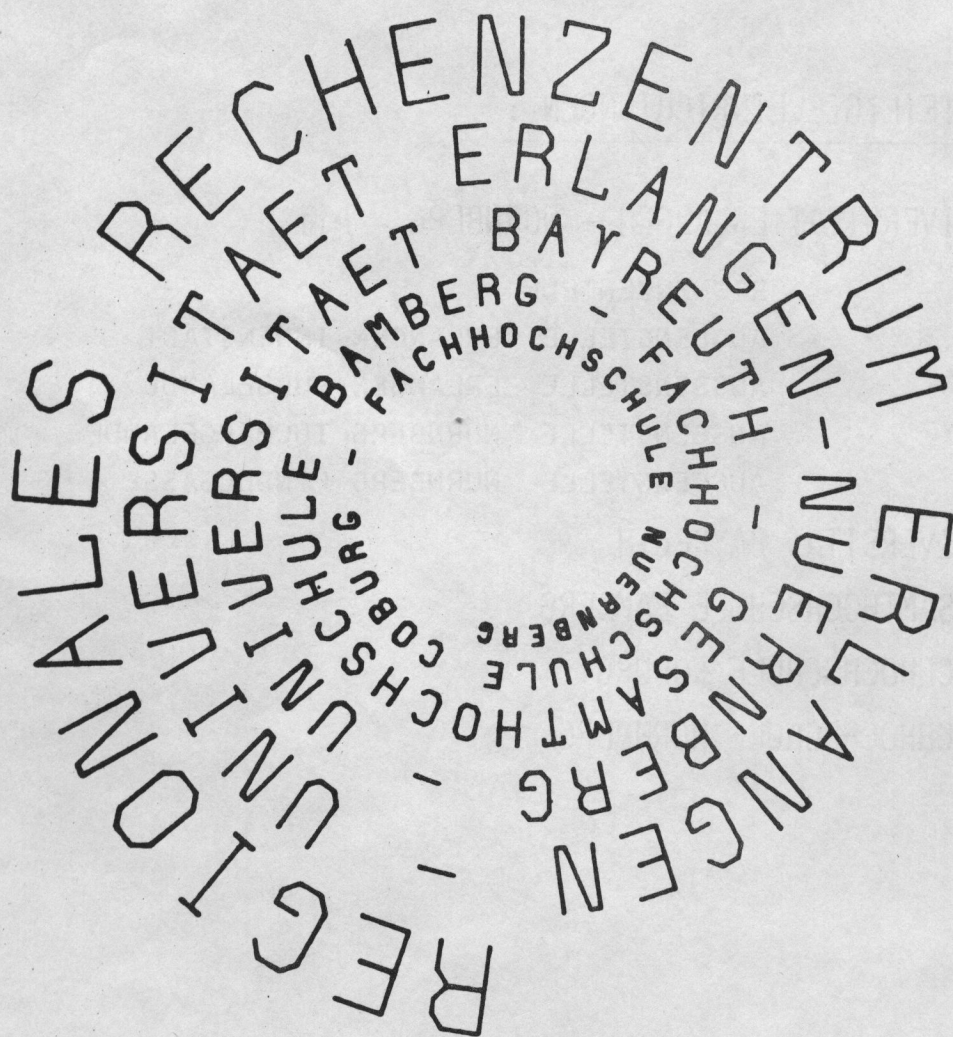
Die Veranstaltung findet donnerstags von 16.00 - 18.00 Uhr voraussichtlich im Regionalen Rechenzentrum Erlangen (Seminarraum 2.037) statt.

Sollten jedoch die Teilnehmer aus Nürnberg überwiegen, so kann in der ersten Sitzung eine Verlegung nach Nürnberg beschlossen werden.

Weitere Auskünfte erteilt Dr. F. Bauer, Zimmer 5.233, Lange Gasse 20, 8500 Nürnberg, Telefon: 0911/5302248.

# RRZE

# BENUTZER INFORMATION



BI 22 -ERLANGEN-3.DEZEMBER 1980



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

## Inhaltsverzeichnis

1	Aktuelle Informationen	2
1.1	Benutzerkolloquium	2
1.1.1	Einladung zum Benutzerkolloquium am 24.2.1981	2
1.1.2	Benutzerkolloquium am 18.11.1980	2
1.2	Geräte - Einkauf	3
1.3	Betrieb am Jahreswechsel	3
1.4	Neue Sichtgeräte am TR440	3
1.5	PET jetzt mit Drucker	4
1.6	Groß-/Kleinschreibung an der CYBER	4
1.7	Probleme mit dem Zeichen bei RJE-Stationen	4
1.8	Umleiten von Listen an Außenstellendrucker bzw. zentrale Drucker	5
1.9	Neues aus der RRZE-Küche	6
2	Neues von der Software	7
2.1	Programm- und Dokumentationsänderungen	7
2.2	PORT - Mathematische Unterprogrammbibliothek (CYBER)	8
2.3	GASP IV - Programmbibliothek zur Simulation kombiniert diskret-kontinuierlicher Systeme	9
2.4	TASH - Topographisches Aufnahme- und Auswertesystem der TU Hannover (CYBER )	9
2.5	SPSS - Nun auch an der CYBER	10
2.6	Symbolische Informationsverarbeitung mit dem Mikro- Rechner KBS-10	10
2.7	LISP an der CYBER 173	13
2.8	APL an der CYBER 173	15

Anhang:        *Software-Überblick*



## 1 Aktuelle Informationen

### 1.1 Benutzerkolloquium

#### 1.1.1 Einladung zum Benutzerkolloquium am 24.2.1981

Das nächste Benutzerkolloquium findet am Dienstag, dem 24.2.1981 um 16 hct im H4 statt. Alle Benutzer des RRZE sind dazu herzlich eingeladen. Bitte besprechen Sie Ihre Fragen möglichst vorher mit einem Benutzervertreter oder mit der Leitung des RRZE. \*

#### 1.1.2 Benutzerkolloquium am 18.11.1980

Protokoll: Dr. Koller

##### 1. Bericht des Rechenzentrumsleiters:

Einerseits hat sich das TR440-System erfreulicherweise stabilisiert, andererseits müssen sich die Benutzer darauf einstellen, daß der TR440 nur noch eine Standzeit von max. zwei bis drei Jahren hat. Nachfolgelösungen sind in der Diskussion.

##### 2. Betriebssysteme (Krausenberger):

An der CYBER wird mit der im wesentlichen unveränderten Betriebssystemversion gefahren.

- Die Speicherkapazität wurde um zwei FMD-Platten mit je 600 Mio-Byte erweitert.
- Die RJE-Stationen wurden auf RBF umgestellt.
- Des weiteren wurde ein Drucker mit Groß-/Kleinschreibung installiert. Auf diesem können auch Files, die mit dem Editor im 6 Bit/12 Bit-Code erstellt wurden, ausgedruckt werden. Das geschieht mit folgender Kommandofolge:

FCOPY,P=lfm,N=LIST.  
ROUTE,LIST,DC=PR,EC=A9.

- Abschließend wurde darauf hingewiesen, daß der E-Parameter im SUBMIT-Kommando nicht mehr funktioniert. Statt dessen wird folgende Anweisung empfohlen:

ROUTE,OUTPUT,DEF,DC=PR,UN=rjename.

(Anm. der Redaktion: Eine Liste der verfügbaren rje-namen finden Sie in dieser BI.)

##### 3. Anwendersoftware (Cramer):

Es wurde eine Reihe von neuen Programmen vorgestellt, die in der vorliegenden BI ausführlich kommentiert werden.

##### 4. Nachwahl eines Benutzervertreters:

Als Nachfolger für Herrn Woitok wurde Herr Hein vom Lehrstuhl Informatik 5 (Mustererkennung) gewählt.

## 5. Sonstiges:

- Es wird gebeten, die neuen TR-Bildschirme (Teleray) auszuprobieren.
- Das Rechenzentrumsgebäude ist während der Weihnachtsferien voraussichtlich von 9 bis 17 Uhr geöffnet. Der Rechenbetrieb wird in dieser Zeit von einem Notoperateur aufrecht erhalten.
- Es ist geplant, demnächst ein Treffen für die an der Textverarbeitung interessierten Benutzer zu veranstalten.
- Das nächste Benutzerkolloquium findet am 24.2.1981 statt.

## 1.2 Geräte - Einkauf

Das RRZE hat mit verschiedenen Herstellern und Lieferanten von EDV-Geräten Sonderkonditionen vereinbaren können, die wir auch anderen Instituten anbieten wollen. Bitte wenden Sie sich an Herrn Thomas (Tel. 7815 oder 7031 oder Zimmer 2.042 im RRZE), wenn es um Terminals, Drucker, Modems, Plotter, Lochstreifenleser usw. geht.

## 1.3 Betrieb am Jahreswechsel

Am RRZE gilt folgende Regelung:

Mi 24.12. bis So 28.12.1980: geschlossen

Mo 29.12. bis Di 30.12.1980: geöffnet von 9 bis 17 Uhr, keine Wartung

Mi 31.12.1980 bis Do 1.1.1981: geschlossen

Fr 2.1. und Mo 5.1.1981: geöffnet von 9 bis 17 Uhr, keine Wartung

Di 6.1.1981: geschlossen

ab Mittwoch 7.1.1981 normaler Betrieb.

Außerhalb der Öffnungszeiten laufen die Anlagen im "unbedienten Betrieb", d.h. Dialog- und RJE-Zugang sind (wie an Wochenenden) möglich.

## 1.4 Neue Sichtgeräte am TR440

Im Terminalraum zum TR440 (Inf.-Gebäude 01.153) sind nun, wie angekündigt, weitere Sichtgeräte Typ Teleray 1061 aufgestellt, die SIG51-ähnlich eingesetzt werden können. Der wesentliche Unterschied besteht darin, daß beim Absenden Zeilenlängen über 80 Zeichen in Einzelzeilen aufgebrochen werden. An den Geräten liegt eine Bedienungsanleitung aus. Bitte teilen Sie uns Ihre Erfahrungen auf dem Kommentarblatt oder über die Beratung mit.



### 1.5 PET jetzt mit Drucker

An dem Kleinrechner Commodore PET 2001 des RRZE ist jetzt auch ein kleiner Matrixdrucker angeschlossen. Eine Beschreibung liegt am Gerät aus. Die Benutzung des PET ist wie bisher über die CYBER-Operateure oder die Aufsicht möglich.

### 1.6 Groß-/Kleinschreibung an der CYBER

Das RRZE hat einen Drucker mit einer Kette mit großem Zeichensatz ausgerüstet, sodaß jetzt auch an der CYBER Groß-/Kleinschreibung möglich ist.

Files, die im "time-sharing ASCII mode" (6/12 Display-Code) erstellt wurden (z.B. mit XEDIT,lfn,AS), können nach Umwandlung in den 12 Bit-ASCII-Code an diesem Drucker ausgedruckt werden.

Folgende Steuerkarten sind anzugeben:

FCOPY,P=lfn,N=list,R.

ROUTE,list,DC=PR,EC=A9.

es bedeuten: lfn        Lokaler File im 6/12 Display-Code

list        Lokaler File im 12 Bit-ASCII-Code

wird der Parameter R beim FCOPY-Statement verwendet, so wird auf beiden Files vor und nach der Umwandlung ein REWIND ausgeführt.

### 1.7 Probleme mit dem Zeichen ^ bei RJE-Stationen

(PASCAL und BASIC-Programme von RJE-Stationen)

Zur Zeit werden unsere RJE-Stationen mit einem Leitungsprotokoll betrieben, bei dem das Zeichen "^" nicht übertragen wird, sondern bei der Eingabe zur CYBER als Leerzeichen, bei der Ausgabe von der CYBER als % übertragen wird.

Dieses Zeichen wird jedoch bei einigen Sprachen benötigt.

a) In PASCAL als Pointer-Symbol

Abhilfe: Anstelle ^ das Zeichen # verwenden; der PASCAL - Compiler akzeptiert beide Zeichen als Pointer-Symbol (unabhängig davon, wo der Job herkommt).

b) In Basic bei PRINT-USING

Abhilfe: Für jedes "^" die Zeichenfolge "@B" ablocken, also statt ^^ jetzt @B@B und den Parameter AS beim Basic-Aufruf angeben.

In absehbarer Zeit ist damit zu rechnen, daß auf ein anderes Leitungsprotokoll umgestellt wird, bei dem auch das Zeichen ^ übertragen werden kann.

## 1.8 Umleiten von Listen an Außenstellendrucker bzw. zentrale Drucker

### 1. Aus dem Terminal-Dialog:

Soll ein lokaler File, der im Dialog erstellt wurde (z.B. Liste eines Compilers:...,L=lfm,...), an einen Drucker geschickt werden, so ist folgendes Kommando zu benutzen:

- a) Drucker RRZE-Rechnerraum: ROUTE,lfm,DC=PR.
- b) Drucker RRZE-Expreßstation: ROUTE,lfm,DC=PR,ID=1.
- c) RJE-Drucker: ROUTE,lfm,DC=PR,UN=rjename.  
 mit rjename = BAMRJE für Bamberg  
               BTHRJE für Bayreuth  
               COBRJE für Coburg  
               ERIRJE für Erlangen, Bismarckstr.  
               ERSRJE für Erlangen, Tandemlabor  
               NFHRJE für FHS Nürnberg  
               NFIRJE für SFZ-Nürnberg, Findelgasse  
               NTURJE für WISO-Nürnberg, Tuchergelände.

Enthält die Datei noch keine Vorschubsteuerzeichen in der ersten Spalte, so ist vorher das Kommando COPYSBF zu verwenden.  
 Der Parameter REP=n bewirkt zusätzliches Drucken von n Kopien.

### 2. Drucken der Liste eines mit dem SUBMIT-Kommando erzeugten Jobs:

Soll die Liste eines mit dem SUBMIT-Kommando weggeschickten Jobs an einer bestimmten RJE-Station gedruckt werden, so sollte dieser Job-File am Anfang folgende Steuerkarten enthalten:

```
Jobkarte.
USER,username,password.
CHARGE,username,username.
ROUTE,OUTPUT,DEF,DC=PR,UN=rjename.      !
.....
```

Der Parameter rjename kann die gleichen Werte wie in 1. annehmen.  
 Der Parameter q zur Steuerung des Outputs auf dem SUBMIT-Kommando hat dann keine Bedeutung mehr, das Kommando hat die Form:  
 SUBMIT,jobfile.

Soll die Liste am zentralen Drucker gedruckt werden, so hat das ROUTE-Kommando die Form

```
ROUTE,OUTPUT,DEF,DC=PR,ID=0.    (Rechnerraum) (*)
bzw. ROUTE,OUTPUT,DEF,DC=PR,ID=1. (Expreßstation)
```

Wird der File mit dem Kommando

```
SUBMIT,jobfile,B.
```

abgeschickt und enthält der Jobfile kein ROUTE-Kommando, so wird implizit das Kommando (\*) angenommen.

### 3. Drucken der Liste eines Jobs, der am Kartenleser eingelesen wurde:

Ein am Kartenleser eingelesener Job, der kein ROUTE-Kommando enthält, wird behandelt, als ob ein bestimmtes voreingestelltes ROUTE-Kommando vorhanden wäre:



- a) beim Einlesen an einer RJE-Station:  
ROUTE,OUTPUT,DEF,DC=PR,UN=rjename.  
(mit der jeweiligen Stationskennung).
- b) beim Einlesen an der Expreßstation:  
ROUTE,OUTPUT,DEF,DC=PR,ID=1.
- c) beim Einlesen im Rechnerraum:  
ROUTE,OUTPUT,DEF,DC=PR,ID=0.

Enthält ein Job ein anderes ROUTE-Kommando, so ist dieses maßgebend.

### 1.9 Neues aus der RRZE-Küche

Die Redaktion der BI verrät Ihnen hier einen Geheimtip für die Vorweihnachtszeit und verbindet damit die besten Wünsche für ein frohes Fest und ein erfolgreiches Jahr 1981.

#### Knusperplätzchen

250g Butter, 250g Zucker, 250g Mandeln oder Haselnüsse, 1 Ei, 250g Mehl, 1 Teelöffel Zimt.

Auf einem Backbrett aus den angegebenen Zutaten einen Knetteig herstellen, 1/2 Stunde ruhen lassen, auf einem gefetteten Blech 1/2 cm dick ausrollen, bei ca. 170 C etwa 2 x 8 Minuten backen (d.h. zum gleichmäßigen Backen nach 8 Minuten umdrehen), mit Zuckerguß glasieren und noch heiß in Streifen schneiden.

## 2 Neues von der Software

### 2.1 Programm- und Dokumentationsänderungen

Anfang Dezember wird die Dokumentation der Programmbibliothek auf den neuesten Stand gebracht. Dabei ergeben sich folgende Änderungen:

#### Softwarekatalog (Band 60)

- Softwareüberblick ersetzt (siehe BI-Anhang)
- PORT-Inhaltsverzeichnis eingefügt (siehe PORT-Kurzbeschreibung in dieser BI)
- SFZ-Inhaltsverzeichnis eingefügt
- UNRZPB-Inhaltsverzeichnis ersetzt

#### Bibliothek UNRZPB/CYBER (Band 61)

- Inhaltsverzeichnis ersetzt
- Neue Programme und Prozeduren:
 

CHANGES	Vergleichen von Textfiles
COMPARE	Vergleichen von Textfiles
DECK	Generieren von *DECK - Karten für UPDATE
FMST	File Management System on Tape
PDPTAPE	Lesen von Bändern, die von der PDP11 mit dem Programm PIP beschrieben wurden
SILLY	Automatisches Verfahren zur Silbentrennung englischer Texte
STAGE2	Makroprozessor von Waite
- Änderungen:
 

EBCFR	Apostroph (') bleibt erhalten
GETDOC	Beschreibung eingefügt
GET440	Apostroph (') bleibt erhalten
SAVE440	Quote (") bleibt erhalten, Magnetbandschreib- dicke auf 1600 bpi (B60H) geändert

#### BMDP/CYBER (Band 67)

Aufrufbeschreibung für BMDP 77 eingefügt

#### Bibliothek des SFZ (Band 67)

Gesamte Dokumentation ersetzt

#### Bibliothek SPLLIB/TR440 und CYBER (Band 69)

Aufnahme der Bibliothek:

SPLLIB            Programmbibliothek für Splines und Ausgleichs-  
algorithmen

Die 90-seitige Beschreibung ist ausdrückbar

- a) am TR440 mit:
  - ×ERZEUGE,DOKUMENT.SPLLIB(gerät)
- b) an der CYBER:
  - im RRZE mit:
    - POP(GETDOC)DOC=SPLLIB
    - ROUTE,SPLLIB,DC=PR.
  - an den RJE-Stationen mit:
    - POP(GETDOC)DOC=SPLLIB
    - ROUTE,SPLLIB,DC=PR,UN=rjename.



PASCAL/CYBER (Band 73)

Dokumentation auf den Stand von PASCAL Release 3 gebracht.

GASP IV/TR440 und CYBER (Band 75)

Aufnahme der Bibliothek:

GASP IV            Programmbibliothek zur Simulation kombiniert  
diskret - kontinuierlicher Modelle.  
(siehe GASP IV - Kurzbeschreibung in dieser BI)

FAMULUS/CYBER (Band 75)

Eine 70-seitige deutsche Beschreibung mit vielen Beispielen wurde erstellt, der Ausdruck ist nur im RRZE möglich (Groß-/Kleinschrift):

POP(GETDOC)DOC=FAMULUS

ROUTE,FAMULUS,DC=PR,EC=A9.

FAMULUS wird zur Zeit auf den ASCII-Zeichensatz erweitert, sodaß zukünftig die Verarbeitung von Daten in Groß- und Kleinschrift möglich ist.

TASH/CYBER (Band 78)

Aufnahme des Programms:

TASH            Topographisches Aufnahme- und Auswertesystem  
der TU Hannover  
(siehe Kurzbeschreibung in dieser BI)

XEDIT/CYBER (Band 206)

Das deutsche Handbuch und die englische Kurzbeschreibung wurden eingefügt.

2.2 PORT - Mathematische Unterprogrammbibliothek (CYBER)

PORT ist die Programmbibliothek der Bell Telephone Laboratories, sie enthält einen ausgewogenen Satz von Fortran-Unterprogrammen für numerische Berechnungen. Das Acronym PORT steht für "Portable, Outstanding, Reliable and Tested". Diese Ziele zu erreichen war oberstes Gebot bei der Entwicklung der Bibliothek. Die numerischen Algorithmen wurden mit Sorgfalt ausgewählt und unter den Gesichtspunkten Portabilität, Robustheit, Genauigkeit und Effizienz implementiert.

Die Schwerpunkte der PORT-Bibliothek liegen auf den Sachgebieten Approximation, Differentialgleichungen, lineare Algebra, Integration und Differentiation, Wurzeln von Polynomen, spezielle Funktionen und Fast Fourier Transformationen. Daneben gibt es Hilfsroutinen zur Fehlerbehandlung, zur dynamischen Speicherverwaltung, für Vektoroperationen und zur Ein-/Ausgabe.

Arbeitsspeicher für die PORT-Routinen wird automatisch und dynamisch vom PORT-Speicherverwaltungsmechanismus bereitgestellt. Dieser Prozeß ist für den Benutzer unsichtbar, das Bereitstellen von Arbeitsfeldern für die Unterprogramme entfällt.

Die PORT-Bibliothek soll eine Ergänzung zur NAG-Bibliothek sein, sie steht als binäre Bibliothek zur Verfügung.

Dokumentation:

PORT-Inhaltsverzeichnis (Band 60/Softwarekatalog)

PORT-Aufruf und -Programmbeschreibungen (Band 79, vorerst nur in der RRZE-Beratung)

### 2.3 GASP IV - Programmbibliothek zur Simulation kombiniert diskret-kontinuierlicher Systeme

GASP IV ist ein Simulator, der zur Gruppe der niedrigen Simulationssprachen gehört. GASP IV bietet Sprachelemente, die den Modell-  
aufbau für zeitdiskrete, zeitkontinuierliche und kombiniert diskret-  
kontinuierliche Systeme unterstützen. Es besteht aus einem Fortran-  
Hauptprogramm und Fortran-Unterprogrammen. Die Fortran-Unterpro-  
gramme übernehmen hierbei die folgenden Funktionen:

- Ablaufkontrolle  
Die korrekte Folge der Zustandsübergänge wird überwacht. Hierzu  
gehört insbesondere die Verwaltung und Bearbeitung von Warte-  
schlangen und die Steuerung der Integrationsverfahren.
- Erzeugen von Zufallszahlen  
Zufallszahlen der wichtigsten Verteilungen können erzeugt werden.
- Darstellung und Auswertung statistischen Materials  
Statistisches Material wird gesammelt und in Form von Tabellen und  
Diagrammen ausgegeben.

GASP verlangt nur Fortran-Kenntnisse. Es ist gut dokumentiert und  
und leicht erlernbar.

Dokumentation:

Aufrufbeschreibung in Band 75

The GASP IV User's Manual in Band 75, nur RRZE-Beratung

The GASP IV Simulation Language in Band 75, nur RRZE-Beratung.

### 2.4 TASH - Topographisches Aufnahme- und Auswertesystem der TU Han- nover (CYBER )

Das Programmsystem TASH wurde am Lehrstuhl für Topographie und Kar-  
tographie der Universität Hannover entwickelt, es wurde dem RRZE  
testweise zur Verfügung gestellt.

TASH enthält folgende Programme:

- |        |  |
|--------|--|
| FLEMO  | Berechnung eines regelmäßigen, rasterförmigen, digita-<br>len Flächenmodells aus einem beliebigen, dreidimensio-<br>nalen Punktfeld. |
| FLEDIF | Berechnung eines Flächendifferenzmodells aus zwei digi-<br>talen Flächenmodellen.  |
| ISOHYP | Berechnung eines Isolinenbildes aus einem digitalen<br>Flächenmodell und grafische Ausgabe.  |

TASH ermöglicht die Erstellung von Flurabstandskarten auf der Grund-  
lage digitaler Flächenmodelle, die in der Geologie, Hydrologie,  
Grundwasserbilanzierung, Agrarwirtschaft usw. benötigt werden.

An der Universität Erlangen - Nürnberg wird TASH am Institut für  
Geologie zur Erstellung von Grundwasserflächenkarten eingesetzt.

Dokumentation: Band 78



## 2.5 SPSS - Nun auch an der CYBER

Ab sofort steht an der CYBER 173 das Programmsystem SPSS (Statistical Package for the Social Sciences) in der Version 8 zur Verfügung. Gegenüber SPSS 7 enthält SPSS 8 zusätzliche Prozeduren und Fehlerkorrekturen, insbesondere sind die datenabhängigen Fehler bei den nichtparametrischen Tests (NPAR TESTS) behoben.

Die CDC-Version von SPSS 8 ermöglicht mit der Prozedur PLOT die Ausgabe von Kreuzdiagrammen auf einem Grafikgerät (ähnlich wie SCATTERGRAM auf dem Drucker).

Eine vorläufige SPSS-Aufrufbeschreibung für die CYBER mit einer Liste der aufrufbaren Prozeduren wird in Band 67 (SPSS/CYBER) eingefügt, eine ausführliche Dokumentation wird zur Zeit erstellt.

Die Aufrufbeschreibung ist ausdrückbar

im RRZE mit:

POP(GETDOC)DOC=SPSS

ROUTE,SPSS,DC=PR.

an den RJE-Stationen mit:

POP(GETDOC)DOC=SPSS

ROUTE,SPSS,DC=PR,UN=rjename.

Bei Problemen, Fehlern und Fragen wenden Sie sich bitte an Herrn Cramer (Zimmer 2.030, Tel. 7816).

## 2.6 Symbolische Informationsverarbeitung mit dem Mikro-Rechner KBS-10

Auf dem Mikrorechner KBS-10 stehen zwei Programmsysteme zur symbolischen Informationsverarbeitung zur Verfügung:  
TLC-LISP und muMath/muSIMP.

### 1. TLC-LISP

TLC-LISP ist ein dialogfähiger LISP-Interpreter, der eine Teilmenge des modernsten LISP-Dialekts, LISP-Machine-LISP, implementiert. LISP-Machine-LISP ist eine Erweiterung von MACLISP für die am MIT entwickelte LISP-Maschine, die LISP-Programme aufgrund ihrer Hardware-Architektur wesentlich effizienter und über einem wesentlich größeren Adreßraum (24-Bit Adressen, virtuell bis 2Mx32) ausführt als das MACLISP-System auf einem Großrechner (DEC-10).

Die wichtigsten Charakteristika von TLC-LISP sind:

- Es bietet die volle Mächtigkeit von LISP und gestattet damit die Manipulation von Funktionen (Prozeduren) als Datenobjekte, unterstützt einen objekt-orientierten Programmierstil und bietet Hilfsmittel zur Implementierung eines Programmiersystems. Ein solches Programmiersystem wird zur Zeit im Rahmen einer Arbeitsgemeinschaft entwickelt; es soll die die Komponenten Editor, Trace&Break, Prettyprint, Index (als Dokumentationshilfe)

etc. enthalten.

- Flexible Spezifikationen von Parametern; damit können Funktionen mit beliebiger Parameteranzahl definiert und Verarbeitungsregeln für Parameter (z.B. Voreinstellungen) implementiert werden.
- Verallgemeinerte Kontrollstrukturen: Neben einem allgemeinen "strukturierten" Schleifenkonstrukt gibt es einen "nicht-strukturierten" Escape-Mechanismus.
- Verarbeitung von Zeichenketten
- Volle Arithmetik mit Integer und Real sowie Integers auf beliebiger Basis.
- Verallgemeinerte Record-Strukturen auf der Basis der "Property-Listen" von LISP.
- Ein tabellengesteuerter Scanner zur lexikalischen Analyse, der auch vom Benutzer zur Verarbeitung einer eigenen Oberflächensyntax verwendet werden kann.
- Ein allgemeines Ein-/Ausgabekonzept durch Quellen- und Senken-Spezifikationen.
- Makros, die zur Einlesezeit aktiviert werden können; insbesondere können Makros mit bestimmten Zeichen assoziiert werden, die beim Einlesen zu einer Makroexpansion führen.
- Umfassende Fehler-Kontrolle, die dem Benutzer im Fehlerfall eine Vielzahl von Interaktionen gestattet. Die vom System vorgesehene Fehlerbehandlung kann vom Benutzer auf einfache Weise umdefiniert werden.
- Großer Adreßraum: Zur Zeit arbeitet das System auf einem Adreßraum von 64 KB, jedoch wird schon in der augenblicklichen Version Speicherallokation auf einer 256 Byte-Basis vorgenommen, um später dynamische Speicherverwaltung zu ermöglichen.
- Flexible Speicherverwaltung: Mit Hilfe eines Autoload-Mechanismus ist es möglich, nur aktuell benötigte Funktionen im Arbeitsspeicher zu halten und andere Funktionen bei Bedarf von der Diskette nachzuladen.
- Hohe Ausführungsgeschwindigkeit: Durch Benchmarks wurde gezeigt, daß TLC-LISP auf dem Z80 um einen Faktor 3,5 langsamer ausgeführt wird als UTLISP auf der CYBER 173.

## 2. muMATH/muSIMP

muMATH ist ein dialogfähiges algebraisches Formelmanipulationsprogramm, das ausschließlich in der Implementierungssprache muSIMP, einem LISP-Dialekt mit PASCAL-ähnlicher Syntax, programmiert ist.



muMATH ist modular aufgebaut und damit leicht erweiterbar; es enthält die folgenden Subsysteme:

- beliebig genaue rationale Arithmetik;
- Matrizenoperationen, wobei die Matrixelemente selbst Formeln sein dürfen;
- Algebra-System, wobei der Benutzer Transformationen wie die Expansion von Potenzen von Polynomen, Vereinfachen, Kürzen etc. kontrollieren kann;
- Behandlung trigonometrischer und logarithmischer Funktionen;
- Gleichungs-Subsystem zur exakten Lösung nichtlinearer Gleichungen;
- Symbolische Differentiation und Integration;
- Trace-Paket.

Mit dem System steht eine Reihe von Dateien mit Lektionen zur Verfügung, die im Dialog in den Gebrauch des Systems einführen.

Einige Beispiele mögen die Leistungsfähigkeit von muMATH/muSIMP verdeutlichen:

Um die quadratische Gleichung  $x^2 + x = x + 4c$  für  $x$  zu lösen, wobei  $c$  ein nicht spezifizierter Parameter ist, gibt der Benutzer ein:

```
SOLVE(X^2 + X == X + 4*C, X);
```

Das System antwortet mit

```
X == 2*C^(1/2),
X == -2*C^(1/2)
```

Um die Matrix

```
{[B, 5],
 [3, 0]}
```

zu invertieren, wobei  $B$  ein nicht spezifizierter Parameter ist, muß eingegeben werden:

```
{[B, 5],
 [3, 0]} ^ -1;
```

Als Antwort erhält man

```
{[0, 1/3],
 [1/5, B/15]}
```

Sei  $A$  ein nicht spezifizierter Winkel. Wird

```
TAN(A) * COS(A) + 1 / CSC(A)
```

eingegeben, so erhält man den äquivalenten vereinfachten Ausdruck  $2 * \sin(A)$ .

Um den Ausdruck  $ax + x \sin(x^2)$  bezüglich  $x$  zu integrieren wird eingegeben:

```
INT (A*X + X*SIN(X^2), X);
```

man erhält





und COMPILR, falls der Compiler zugeladen werden soll. Mit diesem Aufruf wird gleichzeitig das Programmiersystem zugeladen, das die folgenden Subsysteme enthält:

- AUX (allgemeine Hilfsfunktionen)
- PP (Prettyprint)
- MAKEFIL (Makefile zur Generierung strukturierter Quellprogramm-Dateien)
- EDIT (LISP-Editor)
- BREAK (Testhilfen)
- DEF (Hilfsfunktionen zur komfortablen Behandlung von Funktionsdefinitionen)

Darüberhinaus werden lokale Dateien (im Makefile-Format) mit weiteren Modulen erzeugt, die bei Bedarf mit Hilfe der Funktion LOAD zugeladen werden können:

- FTRACE (Erweiterung von TRACE)
- PSTRUC (Printstructure: Dokumentationshilfe, die automatisch die Aufrufhierarchie von Prozeduren eines Programmes und Beschreibungen der einzelnen Prozeduren generiert)
- MACRO (Verarbeitung von Macros als Erweiterung von UTLISP)
- RECORD (Subsystem zur Handhabung von Records analog PASCAL)
- TRIGRAN (trigonometrische Funktionen und Zufallszahlengenerator)
- EEVAL (Erweiterung der "Top-level"-Schleife von UTLISP)

Alle Module des Programmiersystems sind weitgehend kompatibel zu den entsprechenden Modulen in MACLISP/TR440 und INTERLISP; da sich eine ausführliche Dokumentation noch in Vorbereitung befindet, sei nochmals auf diese Manuals verwiesen.

## 2. LISPF3

Neben UTLISP wurde zu Ausbildungszwecken auch das in Uppsala entwickelte und zu INTERLISP kompatible portable System LISPF3 installiert.

Bei der Standard-Initialisierung werden die (mit den in UTLISP ähnlichen) Module des Programmiersystems BASIC (=AUX), PP, MAKEFILE, EDIT, BREAK mit ADVICE und DEF zugeladen.

In Band 60 der Dokumentation befindet sich die Beschreibung von LISPF3.

## 2.8 APL an der CYBER 173

Das APL-System an der CYBER 173 kann nicht nur von Terminals aus, die den vollen APL-Zeichensatz bieten, benutzt werden, sondern auch von normalen Dialoggeräten aus und im Batch-Betrieb.

Hierzu ist der TT-Parameter anzugeben:

TT = 713  
bzw. TT = BATCH

In diesem Fall gelten die im APL-Manual, App. C angegebenen Ersatzdarstellungen für die APL-Sonderzeichen.

Um volle Kompatibilität des Systemverhaltens mit den dokumentierten Leistungen zu gewährleisten, sollten die im APL-Manual, C-2 ff ("LOGON Characters and Terminal Controls") gegebenen Hinweise beachtet werden.

Neben den im Manual beschriebenen Subsystemen zur Handhabung von Dateien steht an der CYBER 173 auch noch das in APL implementierte, dialogfähige, statistische Programmsystem STATPACK zur Verfügung. Es ist in sechs APL-Workspaces modular organisiert und bietet die in der folgenden Abbildung wiedergegebenen Leistungen.

STATPACK ist selbstdokumentierend. Um einen Überblick zu erhalten, lade man STP1 von APL1 und gebe LISTHOW sowie LIST ein.

Man beachte, daß hier anstelle von STP5 aus technischen Gründen STP9 angegeben werden muß! Zuladen von STP5 führt zu einem Abbruch des APL-Laufs (PP Abort).



ENTERED:06/11/70

TO FIND OUT WHAT IS AVAILABLE IN THE STATISTICAL PACKAGE, TYPE

## LIST

ALL FUNCTIONS WILL BE LISTED.  
THE ENTRY

SORT STP3 SORTING

E.G., INDICATES THAT A FUNCTION SORT IS STORED WITH ITS DOCUMENTATION IN WORKSPACE STP3.

ANY FUNCTION WHICH IS REQUIRED AS A SUBFUNCTION IN ANOTHER WORKSPACE HAS ALSO BEEN COPIED, WITHOUT ITS DOCUMENTATION, INTO THAT WORKSPACE.

## LIST

FUNCTION	WS	FUNCTION DESCRIPTION
ANOVA	STP1	COMPLETE FACTORIAL ANOVA
ANOVA1	STP1	CROSSED, NESTED ANOVA
ANOVA2	STP1	ONE-WAY ANOVA
ASSIGN	STP4	ASSIGNMENT PROBLEM
BINOM	STP3	BINOMIAL PROBABILITIES
CM	STP2	SIMPLE CORR. MATRIX
CORR	STP2	SIMPLE AND PARTIAL CORR.
COSTFLOW	STP4	MINIMUM COSTFLOW
CPM	STP4	CRITICAL PATH METHOD
CPM1	STP5	CPM ALGORITHM
CTAB	STP3	2-WAY CONTINGENCY TABLE
CTransPORT	STP4	CAPACITATED TRANSP. PROBLEM
DSTAT	STP1	DESCRIPTIVE STATISTICS
FR	STP1	1-WAY FREQUENCY TABLE
FREQ	STP1	NORMAL FIT
FR2	STP1	2-WAY FREQUENCY TABLE
HIST	STP1	FREQUENCY HISTOGRAM
INV	STP2	PIVOTING GAUSS-JORDAN INV
JINV	STP2	GAUSS-JORDAN INVERSE
LINPR	STP6	CONVERSATIONAL L. P.
LIST	STP1	LISTS AVAILABLE PROGRAMS
LPSOLN	STP4	L. P. SENSITIVITY ANALYSIS
MVSD	STP1	MEAN, VAR AND ST DEV
NETFLOW	STP4	FORD-FULKERSON ALGORITHM
NTILES	STP1	MEDIAN, QUANTILES ETC.
PBS	STP3	+ \ OPERATOR
PERMUTE	STP3	PERMUTE VECTOR
REG	STP2	SIMPLE AND MULTIPLE REG
RES	STP2	RESIDUALS
RND	STP3	ROUND
RSIM	STP4	LINEAR PROGRAMMING
RSIM1	STP4	LINEAR PROGRAMMING
SCORR	STP2	SIMPLE CORRELATION
SIMPLEX	STP4	LINEAR PROGRAMMING
SMOOTH	STP3	WEIGHTED MOVING AVERAGE
SORT	STP3	SORTING
SR	STP2	SIMPLE REGRESSION
SS	STP1	SUBROUTINE FOR ANOVA
STATPACK2HOW	STP1	ORGANISATION OF STATPACK2
STATRES	STP2	RESIDUAL STATISTICS
STREG	STP2	STEPWISE REGRESSION
TRANSPORT	STP4	TRANSPORTATION PROBLEM

Abb.1: Programmsystem STATPACK

RRZE	PROGRAMMBIBLIOTHEK	Software-Überblick
REGIONALES	CYBER/TR440	
RECHENZENTRUM		Bearb. : H. Cramer
ERLANGEN	RRZE-Dok: Band 60	Datum : November 80

## Software-Überblick : System- und Anwendungs-Software

### Inhaltsverzeichnis

- A Verzeichnis der Sachgebiete und Teilsachgebiete
- B Schlagwortverzeichnis
- C Verzeichnis der Programmnamen
- D Programmkatalog

### Erläuterungen zu den Spaltenbezeichnungen in den Kapiteln A - D

#### A Verzeichnis der Sachgebiete und Teilsachgebiete

SACHGEBIET                      Programme zu SACHGEBIET finden Sie auf SEITE  
SEITE                              im PROGRAMMKATALOG D

#### B Schlagwortverzeichnis

SCHLAGWORT                      zu SCHLAGWORT siehe die folgenden, eingerück-  
SEITE                              ten SACHGEBIETE bzw. PROGRAMMNAMEN auf  
SEITE im PROGRAMMKATALOG D  
(Programme sind durch P.. gekennzeichnet)

#### C Verzeichnis der Programmnamen

PROGRAMM                        Wenn Ihnen der Name eines Programms bekannt ist,  
SEITE                              finden Sie es unter PROGRAMM auf SEITE im  
PROGRAMMKATALOG D



D Programmkatalog

SACHGEBIET	Sachgebiet und Teilsachgebiet
PROGRAMM	Name des Programms oder Programmpakets
KURZBESCHREIBUNG	Kurze Beschreibung des Programms
SPR	Sprache, in der das Programm aufgerufen wird bzw. geschrieben ist F66 : FORTRAN66 (FORTRAN4) F77 : FORTRAN77 (FORTRAN5) A60 : ALGOL60 COB : COBOL BCP : BCPL CMP : COMPASS TAS : TAS K : Kommando-Sprache keine Angabe : siehe in DOKUMENTATION
T	Typ des Programms H : Hauptprogramm U : Unterprogramm K : Kommando-Prozedur keine Angabe : siehe in DOKUMENTATION
RA	Rechenanlage C : CYBER T : TR440
DOKUMENTATION	Bandnummer und Abschnitt der RRZE-Dokumentation Aufstellungsorte: Beratung, Benutzerräume und RJE-Stationen

# A 1 VERZEICHNIS DER SACHGEBIETE UND TEILSACHGEBIETE

SACHGEBIET	SEITE	SACHGEBIET	SEITE	SACHGEBIET	SEITE
------------	-------	------------	-------	------------	-------

ASSEMBLER	D 1	SPLINES	D 6		
COMPILER/INTERPRETER	D 1	STATIK/DYNAMIK	D 6		
DATENBANKSYSTEME	D 2	STATISTIK	D 6		
DATENMANIPULATION	D 2	TESTHILFEN	D 7		
EDITOREN	D 3	TOPOGRAPHIE	D 7		
EIGENWERTE UND -VEKTOREN	D 3				
ELEKTROTECHNIK	D 3				
FORMELMANIPULATION	D 3				
GRAFIK	D 4				
LISTENVERARBEITUNG	D 4				
MATRIZENOPERATIONEN	D 4				
NETZPLANTECHNIK	D 4				
OPTIMIERUNG	D 4				
PROGRAMMBIBLIOTHEKEN	D 5				
SIMULATION	D 5				
SORTIEREN	D 5				

SACHGEBIET	SEITE	SACHGEBIET	SEITE	SACHGEBIET	SEITE
------------	-------	------------	-------	------------	-------

# A 1 VERZEICHNIS DER SACHGEBIETE UND TEILSACHGEBIETE



B 1 SCHLAGWORTVERZEICHNIS MIT ZUGEHÖRIGEN SACHGEBIETEN UND PROGRAMMEN (P..)

SCHLAGWORT	SEITE	SCHLAGWORT	SEITE	SCHLAGWORT	SEITE
ALGEBRA		DATENBANKSYSTEME		DOKUMENTATION	
P..REDUCE	D 3	DATENBANKSYSTEME	D 2	P..FAMULUS	D 2
P..REDUCE	D 2			P..TELDOK	D 2
P..SAC-1	D 3	DATENMANIPULATION	D 2		
P..SCHIP	D 3	DATENMANIPULATION	D 2	DRUCKAUFBEREITUNG	D 3
				P..EDITOR	D 3
APPROXIMATION		DATENUMWANDLUNG	D 2	P..TV	D 3
P..SPLLIB	D 6	P..FORM	D 2		
ASSEMBLER		DEBUG, INTERAKTIV	D 7	DUMP	D 7
ASSEMBLER	D 1	P..CID	D 7	P..CID	D 7
				DYNAMIK	
AUSGLEICHS-ALGORITHMEN		DIALOGEDITOR	D 3	P..NONSAP	D 6
P..SPLLIB	D 6	P..EDIERE	D 3	P..SAP	D 6
		P..EDIT	D 3		
BIBLIOTHEKSWARTUNG		P..XEDIT	D 3	EDITOREN	D 3
P..SAM	D 3			EDITOREN	D 3
P..UPDATE	D 3	DIALOGSPRACHE	D 1		
		P..LOGO	D 1	EIGENVEKTOREN	D 3
CLUSTERANALYSE				P..EISPACK	D 3
P..CLUSTAN	D 6	Dienstleistungsprogramme	D 5	P..FORMAT	D 4
		P..CGK	D 5		
COMPILER/INTERPRETER		P..CGK	D 5	EIGENWERTE	D 4
COMPILER/INTERPRETER	D 1	P..STARG	D 5	P..FORMAT	D 4
		P..UNRZPB	D 5		
CROSS-ASSEMBLER				EIGENWERTE UND -VEKTOREN	D 3
P..MCS80	D 1	DISKRETE SIMULATION	D 5	EIGENWERTE UND -VEKTOREN	D 3
		P..GASP4	D 5		
DATENAUFBEREITUNG		P..GASP5	D 5	ELEKTR. NETZWERKE	D 3
P..SIR	D 2	P..GPSS	D 5	P..ANP3	D 3
		P..GPSSF	D 5	P..NAP2	D 3
				P..SPICE	D 3

SCHLAGWORT	SEITE	SCHLAGWORT	SEITE	SCHLAGWORT	SEITE
B 1		SCHLAGWORTVERZEICHNIS MIT ZUGEHÖRIGEN SACHGEBIETEN UND PROGRAMMEN (P..)			

# B 2 SCHLAGWORTVERZEICHNIS MIT ZUGEHÖRIGEN SACHGEBIETEN UND PROGRAMMEN (P..)

SCHLAGWORT	SEITE	SCHLAGWORT	SEITE	SCHLAGWORT	SEITE
ELEKTROTECHNIK ELEKTROTECHNIK	D 3	INFORMATION RETRIEVAL P..ATHENA P..DBS P..FAMULUS P..SIR P..TELDOK P..TOTAL	D 2 D 2 D 2 D 2 D 2 D 2	LINEARE OPTIMIERUNG P..MPCODE	D 4
EXTREMWERTE P..FORMAT	D 4			LINGUISTIK P..LISP P..LISPF3 P..UTLISP	D 4 D 4 D 4 D 4
FINITE ELEMENTE P..NONSAP P..SAP	D 6 D 6	INTEL8080-SIMULATION P..MCS80	D 1	LISTENVERARBEITUNG P..SIMULA	D 4 D 2
FORMELMANIPULATION FORMELMANIPULATION P..LISP P..LISPF3 P..UTLISP	D 3 D 4 D 4 D 4	ISOLINIEN P..TASH KARTOGRAPHIE P..TASH	D 7 D 7	LITERATUR-DOKUMENTATION P..FAMULUS P..TELDOK	D 2 D 2
GEOLOGIE P..TASH	D 7	KERNPHYSIK P..CERN	D 5	MAKROPROCESSOR P..STAGE2 P..STAGE2	D 2 D 2
GLEICHUNGSSYSTEME, LINEAR P..EISPACK P..FORMAT	D 3 D 4	KNUTH-ASSEMBLER P..MIX	D 1	MATHEMATIK P..CERN P..CGK P..CGK P..IMSL P..MATHLIB P..NAG P..PORT P..SSP P..STARG P..UNRZPB	D 5 D 5 D 5 D 5 D 5 D 5 D 5 D 5 D 5
GRAFIK GRAFIK P..SPLLIB	D 4 D 6	KONTINUIERL. SIMULATION P..DYNAMO P..GASP4 P..GASP5 P..SIMULA	D 5 D 5 D 5 D 2		
HILFSPROGRAMME P..CGK P..CGK P..STARG P..UNRZPB	D 5 D 5 D 5 D 5	KONTROLLEREIGNIS P..CID	D 7		

SCHLAGWORT	SEITE	SCHLAGWORT	SEITE	SCHLAGWORT	SEITE
B 2		SCHLAGWORTVERZEICHNIS MIT ZUGEHÖRIGEN SACHGEBIETEN UND PROGRAMMEN (P..)			



# B 3 SCHLAGWORTVERZEICHNIS MIT ZUGEHÖRIGEN SACHGEBIETEN UND PROGRAMMEN (P..)

SCHLAGWORT	SEITE	SCHLAGWORT	SEITE	SCHLAGWORT	SEITE
MATRIZENOPERATIONEN		PORTABLE PROGRAMME		SPLINES	
MATRIZENOPERATIONEN	D 4	P..PFORT	D 2	SPLINES	D 6
NETZPLANTECHNIK		P..STAGE2	D 2		
NETZPLANTECHNIK	D 4	P..STAGE2	D 2	STATIK/DYNAMIK	D 6
NETZWERKANALYSE		PROGRAMMBIBLIOTHEKEN		STATIK/DYNAMIK	
P..ANP3	D 3	PROGRAMMBIBLIOTHEKEN	D 5		
P..NAP2	D 3			STATISTIK	D 5
P..SPICE	D 3	PROGRAMMQUELLEN		P..CGK	D 5
		P..SAM	D 3	P..CGK	D 5
NICHTNUMERIK		P..UPDATE	D 3	P..IMSL	D 5
P..BCPL	D 1			P..NAG	D 5
P..BCPL	D 1	QUADRATISCHE OPTIMIERUNG		P..PORT	D 5
P..LISP	D 4	P..MPCODE	D 4	P..SIR	D 2
P..LISPF3	D 4			P..SSP	D 5
P..REDUCE	D 2	QUELLENEDITOR		P..STARG	D 5
P..REDUCE	D 3	P..SAM	D 3	P..UNRZPB	D 5
P..SAC-1	D 3	P..UPDATE	D 3	STATISTIK	D 6
P..SCHIP	D 3				
P..SNOBOL	D 2	RECHNERGEST. UNTERRICHT		SYNTAXPRUEFUNG	D 2
P..SNOBOL	D 4	P..PLANIT	D 2	P..PFORT	
P..UTLISP	D 4				
OPERATIONS RESEARCH		SCHOENSCHREIBEDITOR		SYSTEMPROGRAMMIERUNG	
P..MPCODE	D 4	P..EDITOR	D 3	P..BCPL	D 1
		P..TV	D 3	P..BCPL	D 1
OPTIMIERUNG		SIMULATION		P..PS440	D 2
OPTIMIERUNG	D 4	SIMULATION	D 5	P..SYMPL	D 2
PHYSIK		SORTIEREN		TABELLIERUNG	D 2
P..CERN	D 5	SORTIEREN	D 5	P..RPG	
				TESTHILFEN	D 7
				TESTHILFEN	

SCHLAGWORT	SEITE	SCHLAGWORT	SEITE	SCHLAGWORT	SEITE
B 3		SCHLAGWORTVERZEICHNIS MIT ZUGEHÖRIGEN SACHGEBIETEN UND PROGRAMMEN (P..)			

B 4 SCHLAGWORTVERZEICHNIS MIT ZUGEHÖRIGEN SACHGEBIETEN UND PROGRAMMEN (P..)

SCHLAGWORT	SEITE	SCHLAGWORT	SEITE	SCHLAGWORT	SEITE
------------	-------	------------	-------	------------	-------

TEXTAUFBEREITUNG

P..EDITOR	D 3	ZEICHENVERARBEITUNG	D 2
P..TV	D 3	P..SNOBOL	D 4
		P..SNOBOL	

TEXTEDITOR

P..EDIERE	D 3
P..EDIT	D 3
P..XEDIT	D 3

TEXTVERARBEITUNG

P..SIMULA	D 2
P..SNOBOL	D 4
P..SNOBOL	D 2

TOPOGRAPHIE  
TOPOGRAPHIE

D 7
-----

TRACING

D 7
-----

UMCODIERUNG

D 2
-----

UNTERRICHT

P..ELAN	D 1
P..LOGO	D 1
P..PLANIT	D 2

VEKTOREN

D 4
-----

SCHLAGWORT

SCHLAGWORT	SEITE	SCHLAGWORT	SEITE	SCHLAGWORT	SEITE
------------	-------	------------	-------	------------	-------

B 4 SCHLAGWORTVERZEICHNIS MIT ZUGEHÖRIGEN SACHGEBIETEN UND PROGRAMMEN (P..)



# C 1 VERZEICHNIS DER PROGRAMMNAMEN

PROGRAMM SEITE	PROGRAMM SEITE	PROGRAMM SEITE	PROGRAMM SEITE	PROGRAMM SEITE
----------------	----------------	----------------	----------------	----------------

ANP3	D	3	PS440	D	2
ATHENA	D	2	REDUCE	D	2
BCPL	D	1	REDUCE	D	3
CERN	D	5	RPG	D	2
CGK	D	5	SAC-1	D	3
CID	D	7	SAM	D	3
CLUSTAN	D	6	SAP	D	6
DBS	D	2	SCHIP	D	3
DYNAMO	D	5	SIMULA	D	2
EDIERE	D	3	SIR	D	2
EDIT	D	3	SNOBOL	D	2
EDITOR	D	3	SNOBOL	D	4
EISPACK	D	3	SPICE	D	3
ELAN	D	1	SPLLIB	D	6
FAMULUS	D	2	SSP	D	5
FORM	D	2	STAGE2	D	2
FORMAT	D	4	STARG	D	5
GASP4	D	5	SYNPL	D	2
GASP5	D	5	TASH	D	7
GPSS	D	5	TELDOK	D	2
GPSSF	D	5	TOTAL	D	2
IMSL	D	5	TV	D	3
LISP	D	4	UNRZPB	D	5
LISPF3	D	4	UPDATE	D	3
LOGO	D	1	UTLISP	D	4
MATHLIB	D	5	XEDIT	D	3
MCS80	D	1			
MIX	D	1			
MPCODE	D	4			
NAG	D	5			
NAP2	D	3			
NONSAP	D	6			
PFORT	D	2			

PROGRAMM SEITE	PROGRAMM SEITE	PROGRAMM SEITE	PROGRAMM SEITE	PROGRAMM SEITE
----------------	----------------	----------------	----------------	----------------

# C 1 VERZEICHNIS DER PROGRAMMNAMEN

D 1 PROGRAMMKATALOG NACH SACHGEBIETEN GEORDNET

SACHGEBIET PROGRAMM KURZBESCHREIBUNG SPR T RA DOKUMENTATION

ASSEMBLER

COMPASS	CYBER ASSEMBLER	K C	213	COMPASS
MCS80	INTEL 8080 SIMULATION AM TR440	F66 H T	61	UNRZPB
MIX	ASSEMBLER FUER HYPOTHETISCHE MASCHINEN VON D. KNUTH	BCP K T	76	MIX
TAS	TELEFUNKEN ASSEMBLER	K T	113	TAS

COMPILER/INTERPRETER

ALG60	TR440 ALGOL60-COMPILER	K T	111	ALG60
ALGOL	CYBER ALGOL60-COMPILER	K C	211	ALGOL
ALGOL5	CYBER ALGOL5-COMPILER (REVISED REPORT 1976)	K C	211	ALGOL5
ALGOL68	TR440 ALGOL68 COMPILER (TESTVERSION)	K T	73	ALGOL68
APL	A PROGRAMMING LANGUAGE	K C	223	APL
BASIC	CYBER BASIC-COMPILER	K C	214	BASIC
BASIC	TR440 BASIC-COMPILER	K T	114	BASIC
BCPL	PROGRAMMIERSPRACHE ZUR LOESUNG NICHTNUMERISCHER PROBLEME (TR440)	K T	117	BCPL
BCPL	PROGRAMMIERSPRACHE ZUR LOESUNG NICHTNUMERISCHER PROBLEME (CYBER: IN VORBEREITUNG )	K C		BCPL
COBOL	TR440 COBOL COMPILER	K T	112	COBOL
COBOL5	CYBER COBOL5-COMPILER	K C	212	COBOL5
ELAN	EDUCATION LANGUAGE(PROGRAMMIERSPRACHE F.D.SCHULUNTERRICHT)	K T	73	ELAN
FTN	TR440 FORTRAN4-COMPILER (FORTRAN66)	K T	110	FORTRAN
FTN	CYBER FORTRAN4-COMPILER (FORTRAN66)	K C	210	FORTRAN
FTN5	CYBER FORTRAN5-COMPILER (FORTRAN77)	K C	210	FORTRAN
LISP	LIST PROCESSOR (VERSCHIEDENE COMPILER UND INTERPRETER, SIEHE "LISTENVERARBEITUNG" )	K CT	66	LISP
LOGO	DIALOGSPRACHE FUER DEN INFORMATIKUNTERRICHT	K T	75	LOGO

SACHGEBIET

PROGRAMM KURZBESCHREIBUNG

SPR T RA DOKUMENTATION

D 1 PROGRAMMKATALOG NACH SACHGEBIETEN GEORDNET



COMPILER/INTERPRETER

PASCAL	PROGRAMMIERSPRACHE VON N.WIRTH	K	CT	73	PASCAL
PFORT	SYNTAXCHECKER FUER PORTABLES FORTRAN	K	CT	73	PFORT
PL1	CYBER PL/I-COMPILER	K	C	215	PL1
PL1	TR440 PL/I-COMPILER	K	T	115	PL1
PLANIT	PROGRAMMING LANGUAGE FOR INTERACTIVE TEACHING	K	T	120	PLANIT
PS440	TR440-SYSTEMPROGRAMMIERSPRACHE	K	T	145	PS440
REDUCE	MANIPULATION ALGEBRAISCHER AUSDRUECKE	K	T	66	REDUCE
RPG	REPORT PROGRAM GENERATOR	K	T	118	RPG
SIMULA	SIMULA COMPILER (NORWEGEN)	K	C	221	SIMULA
SNOBOL	SNOBOL4 PROGRAMMING LANGUAGE	K	C	76	SNOBOL
STAGE2	MACROPROCESSOR VON WAITE (TR440)	K	T	61	STARG
STAGE2	MACROPROCESSOR VON WAITE (CYBER)	K	C	61	UNRZPB
SYMPL	CYBER-SYSTEMPROGRAMMIERSPRACHE	K	C	222	SYMPL

DATENBANKSYSTEME

ATHENA	ABFRAGESYSTEM ZU TOTAL	K	C	240	ATHENA
DBS	TR440 DATENBANKSYSTEM	K	T	140	DBS
FAMULUS	PROGRAMMSYSTEM ZUR LITERATUR-DOKUMENTATION	F66	H C	75	FAMULUS
SIR	SCIENTIFIC INFORMATION RETRIEVAL (IN VORBEREITUNG)	F66	H C	67	SIR
TELDOK	TELEFUNKEN DOKUMENTATIONSSYSTEM	K	T	141	TELDOK
TOTAL	(DBS-ABFRAGESYSTEM) CYBER DATENBANKSYSTEM	K	C	240	TOTAL

DATENMANIPULATION

FORM	FILE ORGANIZER AND RECORD MANAGER	K	C	232	FORM
------	-----------------------------------	---	---	-----	------

D 3 PROGRAMMKATALOG NACH SACHGEBIETEN GEORDNET

SACHGEBIET PROGRAMM KURZBESCHREIBUNG SPR T RA DOKUMENTATION

EDITOREN

EDIERE	TR440 TEXTEDITOR	K	T	68	EDIERE
EDIT	CYBER TEXTEDITOR	K	C	206	EDIT
EDITOR	DRUCKAUFBEREITUNG VON TEXTEN	K	T	68	EDITOR
SAM	VERWALTEN VON PROGRAMMQUELLEN/-BIBLIOTHEKEN IM BATCH	K	T	68	SAM
TV	DRUCKAUFBEREITUNG VON TEXTEN (IN VORBEREITUNG)	K	C	68	TV
UPDATE	VERWALTEN VON PROGRAMMQUELLEN IM BATCH	K	C	233	UPDATE
XEDIT	ERWEITERTER CYBER TEXTEDITOR	K	C	206	XEDIT

EIGENWERTE UND -VEKTOREN

EISPACK	EIGENSYSYSTEM SUBROUTINE PACKAGE	F66	U	CT	65	EISPACK
---------	----------------------------------	-----	---	----	----	---------

ELEKTROTECHNIK

ANP3	ANALYTIC NETWORK PROGRAM 3	F66	H	T	72	ANP3
NAP2	NONLINEAR ANALYSIS PROGRAM FOR ELECTRONIC CIRCUITS	F66	H	T	72	NAP2
SPICE	A COMPUTER PROGRAM TO SIMULATE SEMICONDUCTOR CIRCUITS	F66	H	C	72	SPICE

FORMELMANIPULATION

REDUCE	MANIPULATION ALGEBRAISCHER AUSDRUECKE	K	T	66	REDUCE	
SAC-1	SYMBOLIC ALGEBRAIC CALCULATIONS	F66	U	CT	61	UNRZPB
SCHIP	SCHOONSCHIP: A PROGRAM FOR ALGEBRAIC MANIPULATIONS	CMP	H	C	62	CERN

SACHGEBIET

PROGRAMM KURZBESCHREIBUNG	SPR	T	RA	DOKUMENTATION
---------------------------	-----	---	----	---------------

D 3 PROGRAMMKATALOG NACH SACHGEBIETEN GEORDNET



D 4 PROGRAMMKATALOG NACH SACHGEBIETEN GEORDNET

SACHGEBIET PROGRAMM KURZBESCHREIBUNG

SPR T RA DOKUMENTATION

GRAFIK

EGS ERLANGER GRAFIK-SYSTEM  
EGS ERLANGER GRAFIK-SYSTEM

A60 U CT 69 GRAFIK  
F66 U CT 69 GRAFIK

LISTENVERARBEITUNG

LISP LISP1.5 UND MACLISP  
LISPF3 PORTABLE LISP-IMPLEMENTIERUNG,  
INTERLISP-KOMPATIBEL  
SNOBOL SNOBOL4 PROGRAMMING LANGUAGE  
UTLISP UTLISP4.0

BCP H T 66 LISP  
F66 H C 66 LISPF3  
CMP H C 76 SNOBOL  
CMP H C 66 UTLISP

MATRIZENOPERATIONEN

FORMAT FORMAT-440: UNTERPROGRAMMPAKET FUEH  
MATRIXBERECHNUNGEN

F66 U T 110.2 FORMAT

NETZPLANTECHNIK

BKN NETZPLANPROGRAMMSYSTEM

K T 144 BKN

OPTIMIERUNG

MPCODE A VERSATILE LINEAR AND QUADRATIC MATHEMATICAL  
PROGRAMMING SYSTEM

F66 H CT 61 UNRZPB

SACHGEBIET

PROGRAMM KURZBESCHREIBUNG

SPR T RA DOKUMENTATION

D 4 PROGRAMMKATALOG NACH SACHGEBIETEN GEORDNET

D 5 PROGRAMMKATALOG NACH SACHGEBIETEN GEORDNET

SACHGEBIET

PROGRAMM KURZBESCHREIBUNG

SPR T RA DOKUMENTATION

PROGRAMMBIBLIOTHEKEN

CERN

PROGRAMMBIBLIOTHEK DER EUROPAEISCHEN  
ATOMFORSCHUNGSZENTRALE GENF

F66 C 62 CERN

CGK

PROGRAMME DER COMPUTER GESELLSCHAFT KONSTANZ

F66 U T 110.1

CGK

PROGRAMME DER COMPUTER GESELLSCHAFT KONSTANZ

A60 U T 111.1

IMSL

INTERNATIONAL MATHEMATICAL&STATISTICAL  
LIBRARIES

F66 U T 71 IMSL

MATHLIB

FORTRAN COMMON LIBRARY MATHEMATICAL ROUTINES

F66 U C 210.1 MATHLIB

NAG

BIBLIOTHEK DER NUMERICAL ALGORITHMS GROUP

F66 U C 70 NAG

PORT

BIBLIOTHEK DER BELL TELEPHONE LABORATORIES

F66 U C 60 PORT

SSP

SCIENTIFIC SUBROUTINE PACKAGE (IBM)

F66 U CT 64 SSP

STARG

PROGRAMMBIBLIOTHEK DER STARG (STAENDIGE  
ARBEITSGRUPPE DER TR440-RECHENZENTREN

T 63 STARG

UNRZPB

RRZE-PROGRAMME UND -KOMMANDOPROZEDUREN

CT 61 UNRZPB

SIMULATION

DYNAMO

SPRACHE ZUR SIMULATION KONTINUIERLICHER SYSTEME

K C 75 DYNAMO

GASP4

SIMULATION KOMBINIERT DISKRET-KONTINUIERLICHER  
SYSTEME

U CT 75 GASP4

GASP5

SIMULATION KOMBINIERT DISKRET-KONTINUIERLICHER  
SYSTEME (IN VORBEREITUNG)

U CT 75 GASP5

GPSS

SPRACHE ZUR SIMULATION DISKRETER SYSTEME

K T 119 GPSS

GPSSF

GPSS-FORTRAN: SIMULATION DISKRETER SYSTEME (IN  
VORBEREITUNG)

U CT 75 GPSSF

SORTIEREN

SORT

TR440 SORTIEREN/MISCHEN

K T 130 SORT

SACHGEBIET

PROGRAMM KURZBESCHREIBUNG

SPR T RA DOKUMENTATION

D 5 PROGRAMMKATALOG NACH SACHGEBIETEN GEORDNET



D 6	PROGRAMMKATALOG NACH SACHGEBIETEN GEORDNET				
SACHGEBIET	PROGRAMM KURZBESCHREIBUNG			SPR T RA	DOKUMENTATION

SORTIEREN					
	SORT	CYBER SORT/MERGE		K C	230 SORT

SPLINES					
	SPLLIB	PROGRAMMBIBLIOTHEK FUER SPLINES UND AUSGLEICHS-ALGORITHMEN		F66 U CT	69 SPLLIB

STATIK/DYNAMIK					
	NONSAP	A STRUCTURAL ANALYSIS PROGRAM FOR STATIC AND DYNAMIC RESPONSE OF NONLINEAR SYSTEMS (IN VORBEREITUNG)		F66 H C	74 NONSAP
	SAP	A STRUCTURAL ANALYSIS PROGRAM FOR STATIC AND DYNAMIC RESPONSE OF LINEAR SYSTEMS (IN VORBEREITUNG)		F66 H C	74 SAP

STATISTIK					
	BMD	BIOMEDICAL COMPUTER PROGRAMS		F66 H CT	67 BMD
	BMDP	BIOMEDICAL COMPUTER PROGRAMS (P-SERIE)		F66 H CT	67 BMDP
	CGK	STATISTIK-UNTERPROGRAMMPAKET DER COMPUTER GESELLSCHAFT KONSTANZ (CGK)		F66 U T	110.2
	CLUSTAN	PROGRAMMSYSTEM ZUR CLUSTERANALYSE		F66 H C	67 CLUSTAN
	SFZ	PROGRAMMBIBLIOTHEK DES SOZIALWISSENSCHAFTLICHEN FORSCHUNGSZENTRUMS NUERNBERG		F66 H CT	67 SFZ
	SPSS	STATISTICAL PACKAGE FOR THE SOCIAL SCIENCES (CYBER: IN VORBEREITUNG)		F66 H CT	67 SPSS

SACHGEBIET	PROGRAMM KURZBESCHREIBUNG			SPR T RA	DOKUMENTATION
D 6	PROGRAMMKATALOG NACH SACHGEBIETEN GEORDNET				

D 7 PROGRAMMKATALOG NACH SACHGEBIETEN GEORDNET  
-----  
SACHGEBIET PROGRAMM KURZBESCHREIBUNG SPR T RA DOKUMENTATION  
-----

STATISTIK  
  
SPSS STATISTICAL PACKAGE FOR THE SOCIAL SCIENCES F66 H T 67 SPSS  
(TR440)

TESTHILFEN  
  
CID CYBER INTERACTIVE DEBUG FUER FORTRAN UND K C 205 CID  
COMPASS

TOPOGRAPHIE  
  
TASH TOPOGRAPHISCHES AUFNAHME- UND AUSWERTE-SYSTEM F66 H C 78 TASH  
DER TU-HANNOVER

-----  
SACHGEBIET PROGRAMM KURZBESCHREIBUNG SPR T RA DOKUMENTATION  
-----  
D 7 PROGRAMMKATALOG NACH SACHGEBIETEN GEORDNET  
-----