

# informum

---

---

INfOrMationsforum des Rechenzentrums der Universität Münster

Jahrgang 6, Nr. 4

November 1982

---

| Inhalt                                                                   | Seite |
|--------------------------------------------------------------------------|-------|
| Anträge zur Benutzung der Rechenanlage                                   | 2     |
| Einführung in die Benutzung der Rechenanlage                             | 3     |
| Lehrveranstaltungen im WS 82/83                                          | 3     |
| Vortragsreihe über das Betriebssystem OS/VS2-MVS                         | 5     |
| Ergänzungen des Benutzerhandbuchs                                        | 6     |
| Noch einmal: Magnetplattenplatz                                          | 7     |
| NAG und IMSL contra SSP/FORTRAN                                          | 8     |
| BMDP-Programmsystem, Version 1982                                        | 13    |
| NAG-Bibliothek                                                           | 14    |
| Statistical Analysis System SAS                                          | 14    |
| ESM - Ein neues Programm zur Verwaltung privater<br>Magnetplattendateien | 15    |
| Betriebliche Mitteilungen                                                | 16    |
| Personalia                                                               | 16    |
| Leserforum                                                               | 16    |
| Vorschlag für eine einheitliche Schreibweise von<br>PL/I-Programmen      | 19    |
| FLCT-PREVIEW 2                                                           | 23    |
| Spezielle Wünsche bei der Druckausgabe                                   | 25    |
| Software-Flohmarkt                                                       | 30    |
| Verteilung von Rechenaufträgen nach Jobklassen                           | 32    |

Anträge zur Benutzung der Rechenanlage

W. Bosse

Wie in jedem Jahr werden alle vergebenen Verrechnungsnummern zum Jahresende ungültig. Sofern also ein in diesem Jahr durchgeführtes Rechenvorhaben auch im Jahre 1983 fortgesetzt werden soll, muß rechtzeitig ein neuer Antrag gestellt werden. Damit ein möglichst reibungsloser Ablauf der Bearbeitung erzielt werden kann, werden bis Mitte November neue Antragsformulare an die Institute versandt und auch im Dispatch des Rechenzentrums verfügbar sein. Um weiteren Forderungen des Landesrechnungshofes nachzukommen, ist das Antragsformular gegenüber früher noch ergänzt worden. Wir bitten deshalb, für 1983 ausschließlich die neuen Formulare zu verwenden.

Vom 15.11.1982 an werden Anträge auf Rechenerlaubnis für 1983 vom Rechenzentrum bearbeitet. Um eine Fortsetzung der Arbeiten über das Jahresende hinaus ohne Unterbrechung der Zugangsberechtigung zu gewährleisten, wird empfohlen, die Anträge spätestens bis zum 15.12.1982 im Rechenzentrum einzureichen.

In diesem Zusammenhang soll auch an die Abgabe eines Jahresberichts für jede 1982 zugeteilte Verrechnungsnummer (=Benutzerkennung + Projektkennung) erinnert werden. Gemäß der Benutzungsordnung ist jeder Benutzer verpflichtet, einen derartigen Bericht über die Art und den Umfang der Inanspruchnahme von Betriebsmitteln des Rechenzentrums zu erstellen. Anträge auf Verlängerung einer Rechenerlaubnis können nur dann bearbeitet werden, wenn auch der Jahresbericht für 1982 vorliegt.

Außerdem müssen alle im Jahre 1983 weiterhin benötigten Magnetbänder, Stellplätze und archivierte Banddateien rechtzeitig, d.h. vor Jahresende, angemeldet werden.

ImpressumRedaktion inforum

|             |                |
|-------------|----------------|
| W. Bosse    | (Tel. 83-2461) |
| H. Pudlatz  | (Tel. 83-2472) |
| R. Schmitt  | (Tel. 83-2475) |
| W. A. Slaby | (Tel. 83-2681) |
| E. Sturm    | (Tel. 83-2609) |

Satz: T. Engelbert, C. Icking

Druck: H. Mecke

Rechenzentrum der Universität  
Einsteinstr. 60  
4400 Münster

Auflage dieser Ausgabe: 700

Redaktionsschluß der nächsten Ausgabe: 31.12.1982

Einführung in die Benutzung der Rechenanlage

Unter diesem Titel sind verschiedene Einzelveranstaltungen zusammengefaßt, die jeweils

montags 15-17 Uhr im Hörsaal M4

stattfinden und interessierten Benutzern die Gelegenheit bieten, Kenntnisse für den praktischen Einsatz der Rechenanlage zu erwerben bzw. zu vertiefen. Eine Anmeldung ist dazu nicht erforderlich.

Übersicht über die Termine und Einzelthemen:

- |                       |                                                                                                                                   |             |
|-----------------------|-----------------------------------------------------------------------------------------------------------------------------------|-------------|
| 15.11.82              | Betrieb des Rechenzentrums<br>(maschinelle Ausstattung, betriebliche<br>Regelungen, Bedienung von Geräten)                        | K. Reichel  |
| 22.11.82/<br>29.11.82 | Aufbau eines Jobs, Verwendung von<br>katalogisierten Prozeduren<br>(Einsatz der Parameter; Programm-<br>und Benutzerbibliotheken) | K. B. Mertz |
| 6.12.82/<br>13.12.82  | Dateibeschreibung<br>(Magnetband- und temporäre<br>Magnetplattendateien)                                                          | H. Kamp     |
| 20.12.82              | Dienstprogramm RUMSERV<br>(Anlegen von Magnetplattendateien,<br>Aufruf einzelner Utilities)                                       | E. Schmitt  |

Lehrveranstaltungen im WS 82/83

Eine Übersicht mit kurzen Inhaltsangaben über das Lehrangebot des Rechenzentrums im WS 82/83 ist bereits in der Juli-Ausgabe von inform erschienen. Hier in Kurzform noch einmal die wichtigsten Angaben:

Einführende Vorlesungen

- |        |                                                                     |         |
|--------|---------------------------------------------------------------------|---------|
| 320074 | Einführung in die EDV<br>Mo 13-15, Hörsaal M6 (Beginn: 25.10.82)    | Kisker  |
| 320089 | Programmieren in FORTRAN<br>Mi 13-15, Hörsaal M4 (Beginn: 20.10.82) | Ost     |
| 320093 | Programmieren in FORTRAN<br>Di 13-15, Hörsaal M3 (Beginn: 19.10.82) | Reichel |

- 320108 Programmieren in PL/I (numerische Anwendungen)  
Di 13.30-15 plus 2 Std. Übungen n. V.,  
Hörsaal M6 (Beginn: 19.10.82) Mertz
- 320112 Programmieren in PL/I (nichtnumerische Anwendungen)  
Di 13.30-15 plus 2 Std. Übungen n. V.,  
Hörsaal M4 (Beginn: 19.10.82) Benduhn-Mertz
- 320127 Programmieren in PASCAL  
Mi 9-11, Hörsaal M5 (Beginn: 27.10.82) Kaspar
- Weiterführende Vorlesungen
- 320131 Programmieren in PL/I für Fortgeschrittene  
Mi 15-17, Hörsaal M4 (Beginn: 27.10.82) Sturm
- 320150 Höhere Programmiersprachen: SNOBOL4  
Di 15.30-17, Seminarraum der Bibliothek  
(Beginn: 26.10.1982) Bosse
- 320165 Datenanalyse und statistische Auswertung mit Hilfe eines statistischen Analysesystems  
Mi 9-11, Hörsaal M6 (Beginn: 20.10.82) Zörkendörfer
- 320170 Programmsysteme zur Textverarbeitung: Oxford Concordance Package  
Mi 13.30-15, Seminarraum des Rechenzentrums (Beginn: 20.10.82) Slaby
- 320184 Rechnernetze  
Mo 13-15, Hörsaal M5 (Beginn: 25.10.82) Held/Schulze
- 320199 Vortragsreihe über das Betriebssystem OS/VS2-MVS  
Mi 15-17, Hörsaal M6 (Beginn: 27.10.82) Mertz/Neukäter/  
Meyer-Rinsche/  
Ost/Spellmann/  
Schmitt
- 071436 Seminar über Computermusik  
Mo 14-15.30, Seminarraum des Musikwissenschaftlichen Seminars  
(Beginn: 25.10.82) Brockhoff/  
Hornung/Slaby

- 320203 Kolloquium über Themen der Informatik      Held/die  
Fr 15-17 bzw. 17-19, Hörsaal M6      wissenschaft-  
Vortragstermine werden durch      lichen Mitar-  
besonderen Aushang im Rechenzentrum      beiter des  
bekanntgegeben      Rechenzentrums
- 320218 Anleitung zum Einsatz der EDV bei      die wissen-  
wissenschaftlichen Arbeiten      schaftlichen  
Mitarbeiter des  
Rechenzentrums

Die bereits angekündigte Vorlesung "Methoden der Anwendungs-  
programmierung" (Vorl. Nr. 320146) muß leider entfallen.

#### Vortragsreihe über das Betriebssystem OS/VS2-MVS

Unter diesem Titel werden in Einzelvorträgen die Komponenten  
der auf dem Rechnerkomplex IBM 3032 eingesetzten Systemsoftware  
dargestellt. Es ist geplant, mit folgenden Referaten zu begin-  
nen:

- 27.10.82      Multiple Virtual Storage (MVS) -      H. Meyer-Einsche  
Einführung und Überblick
- 3.11.82      MVS/System Product -      H. Meyer-Einsche  
Funktionelle Erweiterungen des MVS
- 10.11.82      Job Entry Subsystem 2 (JES2):      B. Neukäter  
Elementare Funktionen
- 24.11.82      JES2: Spezielle Funktionen      K. B. Mertz
- 1.12.82      Die Zugriffsmethoden des MVS      R. Schmitt
- 8.12.82      Dateiverwaltung: Magnetplatten-      H. Meyer-Einsche  
organisation, Platzreservierung,  
Katalogstrukturen
- 15.12.82      Resource Access Control Facility      S. Ost  
(RACF) - Ein System zur Zugangs-  
kontrolle und zum Datenschutz
- 22.12.82      System Modification Program (SMP) -      R. Schmitt  
Ein Konzept zur automatischen  
Systempflege

Die Themen für die zweite Semesterhälfte werden im Dezember  
bekanntgegeben. Die Veranstaltungen finden jeweils mittwochs von  
15 bis 17 Uhr im Hörsaal M6 statt.

Ergänzungen des Benutzerhandbuchs

H. Pudlatz

Das Benutzerhandbuch des Rechenzentrums in der Version 4.0 vom Februar 1980 ist praktisch seit seinem Erscheinen (Juni 1980) in einzelnen Details veraltet. Dies ist dem Umstand zuzuschreiben, daß das Betriebssystem und das Softwareangebot an einem Hochschulrechenzentrum in wesentlich stärkerem Umfang fluktuiert als in einem Rechenzentrum der Wirtschaft, bei dem eine geringe Zahl unverändert wiederkehrender Aufgaben, häufig genug durch nur eine Programmiersprache unterstützt, zu bearbeiten ist.

So sind zum Beispiel inzwischen neue Betriebssystemversionen eingesetzt (dokumentiert im Update 4.1 des Benutzerhandbuchs vom Juli 1981), neue Compiler installiert (PASCAL VS, VS FORTRAN), andere zurückgezogen worden (FORTRAN IV G, FORTRAN IV E Extended), um nur die wichtigsten zu nennen. Dasselbe gilt für Programmsysteme und -bibliotheken (SPSS, BMDP, NAG, IMSL, SAS, IMS u.a.). Darüber hinaus wurden inzwischen die ausschließlich auf den Batch-Betrieb zugeschnittenen Benutzerbibliotheken SYSTEM.SOURCE und SYSTEM.DATA zurückgezogen, da sie durch das erweiterte Speicherplatzangebot für Programmquellen und Daten im Dialogsystem VSPC sowie die Möglichkeit der Einrichtung privater Magnetplattendateien überflüssig geworden sind.

Dadurch sind zahlreiche Beispiele im Benutzerhandbuch nicht mehr aktuell, was zur Folge hätte, daß auf fast jeder zweiten Seite des über 400-seitigen Handbuchs Änderungen vorgenommen werden müßten. Daß wir mit jeder durchgeführten Änderung am System, die Konsequenzen für das Benutzerhandbuch hat, dieses nicht jedesmal überarbeiten und die geänderten Seiten allen Benutzern zugänglich machen können, wurde bereits in der Ausgabe des inforum, Jg.5, Nr.3, erläutert. Eine generelle Überarbeitung des Benutzerhandbuchs kann erst bei einer Neuauflage erfolgen.

Dagegen ist die bereits in der o.g. inforum-Ausgabe in Aussicht gestellte Bearbeitung des RUMSERV-Kapitels inzwischen abgeschlossen und druckfertig. Für den Benutzer soll dadurch das bisherige lästige Nachschlagen in älteren inforum-Ausgaben (Jg.4, Nr.4 und Jg.5, Nr.2) ein Ende haben. Die neue RUMSERV-Dokumentation hat ungefähr den doppelten Umfang der alten und wird daher eine geteilte Seitennumerierung im Benutzerhandbuch erforderlich machen. Andererseits wurden durch neue RUMSERV-Funktionen inzwischen einige Prozeduren für separat aufzurufende Dienstprogramme (beschrieben im hinteren Teil des B-Kapitels des Benutzerhandbuchs) überflüssig. Genaue Angaben über den Wegfall älterer Prozeduren mit den entsprechenden Seitenangaben werden der RUMSERV-Dokumentation beigeheftet.

Zusammen mit der neuen RUMSERV-Dokumentation soll das neue Kapitel E des Benutzerhandbuchs (20 Seiten) über den LINKAGE EDITOR und LOADER zur Verteilung gelangen.

Da die Vervielfältigung und Sortierung beider Teile unmittel-

bar nach Fertigstellung dieser inforum-Ausgabe in Angriff genommen werden soll, bitten wir unsere Benutzer um Verständnis dafür, daß mit dem Beginn der Verteilung im Sekretariat des Rechenzentrums (Frau Luth) nicht vor Ende November gerechnet werden kann. Ein gesonderter Anschlag wird am Schwarzen Brett des Rechenzentrums erfolgen.

### Noch einmal: Magnetplattenplatz

H. Meyer-Rinsche

R. Schmitt

Die schwerwiegenden Engpässe im Bereich des Magnetplattenplatzes für VSPC-Dateien ("SYSTEM LIBRARY FULL") werden zu zwei Maßnahmen führen:

- In der ersten Novemberwoche wird der verfügbare Gesamtplatz für die "SYSTEM LIBRARY" um ca. 25 MBytes vergrößert werden, dafür wird der Platz für Tagesdateien von 100 MBytes auf 75 MBytes eingeschränkt. Dieses Vorgehen scheint gerechtfertigt, da bisher der Platz für Tagesdateien nur in geringem Maße genutzt wurde. Durch die veraltete Technologie und den desolaten Zustand der Magnetbandeinheiten wird allerdings eine wirkungsvolle Sicherung der vergrößerten VSPC-Bibliotheken immer problematischer - eine einzige Sicherungskopie umfaßt zur Zeit schon vier Magnetbänder, und der Defekt eines Bandes führt zur Unbrauchbarkeit der gesamten Kopie.
- Die VSPC-Bibliotheken werden alle zwei Wochen in der Nacht von Montag auf Dienstag aufgeräumt, d.h. nicht benutzte und über den Platz von 300 KBytes hinaus angelegte Dateien werden ausgelagert. Damit soll sichergestellt werden, daß grundsätzlich genügend Platz für die tägliche Arbeit mit dem System zur Verfügung steht.

Aus gegebenem Anlaß erinnern wir daran, daß auch für den Platz, den einzelne Benutzer in den vom Rechenzentrum verwalteten SYSTEM-Dateien belegen, die Regeln für privaten Magnetplattenplatz gelten. Insbesondere verlieren Eintragungen (member), die über den festgesetzten Zeitraum hinaus nicht benutzt wurden, ihre Gültigkeit. Bisher wurde darauf verzichtet, in diesem Sinne ungültige Eintragungen zu löschen, der hohe Füllungsgrad dieser Dateien und die diesbezüglichen Klagen der Benutzer zwingen uns aber, auch hier gelegentlich aufzuräumen.

Schließlich sei daran erinnert, daß das Rechenzentrum ganz analog zu privaten Dateien auch für die Eintragungen in den SYSTEM-Dateien keine Gewährleistung übernehmen kann. Jeder Benutzer sollte also in der Lage sein, seine Eintragungen jederzeit wiederherstellen zu können. RUMSEEV bietet Hilfsmittel zur Auslagerung auf Magnetband und zum späteren Rückladen.

NAG und IMSL contra SSP/FORTRAN

F.-P. Spellmann

In der Ausgabe des inforum vom April 1978 war als Auszug der Benutzerinformation 73 des Regionalen Rechenzentrums für Niedersachsen, die sich wiederum auf Datamation, Juli 1975 bezog, folgender Abschnitt zu lesen, der wegen seiner für Benutzer des SSP/FORTRAN am Rechenzentrum weiterhin bestehenden Aktualität hier noch einmal wiederholt werden soll:

IBM's Scientific Subroutine Package, eine Sammlung von über 250 Routinen, wird an der Stanford University nicht gut beurteilt und soll bis zum Herbst abgeschafft werden. Obwohl das Paket kostenlos war (oder wahrscheinlich gerade deshalb), war die Herstellerunterstützung "niemals besonders hervorragend und ist jetzt überhaupt nicht mehr vorhanden". Der wahre Grund für die Abschaffung von SSP ist nach Angaben der Universität, daß viele Routinen "ungenau, veraltet und gefährlich für den Benutzer" sind. Zum Beispiel korrelieren in RANDU, einem Generator für gleichverteilte Zufallszahlen, aufeinanderfolgende Tripel. MFGR und DMFGR "erwecken den Anschein", den Rang einer Matrix zu bestimmen. "Aufgrund von Rundungsfehlern kann das mit der verwendeten Methode gar nicht geschehen, und das Handbuch gibt das sogar zu", sagt Academic Computing Services Bulletin. Eine Vielzahl weiterer Routinen wird entsprechend beurteilt.

Auch vier Jahre nach Veröffentlichung dieser negativen Beurteilung werden Programme des SSP/FORTRAN weiterhin unverändert von Benutzern des Rechenzentrums eingesetzt. Dies wird u.a. darin begründet sein, daß bisher für einige im SSP/FORTRAN enthaltenen Routinen zur Behandlung statistischer Problemstellungen kein äquivalenter Ersatz in der NAG-Bibliothek zu finden war. Aus diesem Grunde wurde auch die bei Einführung der NAG-Bibliothek im Jahre 1977 zunächst ins Auge gefasste Abschaffung des SSP/FORTRAN nicht durchgeführt, sondern nur die Empfehlung ausgesprochen, Routinen dieser Bibliothek nur "im Notfall" einzusetzen.

Nachdem nun aber mit der IMSL-Bibliothek am Rechenzentrum neben der NAG-Bibliothek eine weitere qualitativ hochstehende Programmbibliothek zur Verfügung steht, sollte dieses Problem im wesentlichen ausgeräumt sein, da die IMSL-Bibliothek erheblich umfassender als die NAG-Bibliothek auf statistische Problemstellungen eingeht (und daneben auch die NAG-Bibliothek im Laufe der Zeit in dieser Richtung erweitert wurde).

In Anbetracht der unverändert schlechten Qualität des SSP/FORTRAN ist es deshalb wohl an der Zeit, die Benutzer der veralteten SSP/FORTRAN-Routinen noch einmal zum Übergang zu Programmen der NAG- oder IMSL-Bibliothek anzuregen. Im Hinblick auf die in diesen Bibliotheken implementierten modernen Verfahren, die darüber hinaus einer ständigen Weiterentwicklung und Wartung (Unterstützung bei der Beseitigung etwaiger Fehler) unterliegen,

und die weite Verbreitung der Bibliotheken an wissenschaftlichen Rechenzentren, die dem Austausch von Programmen entgegenkommt, sollte sich die bei der Umstellung anfallende Arbeit lohnen. Das Rechenzentrum beabsichtigt, die Programme des SSP/FORTRAN u. a. zur Einsparung von Speicherplatz auf Magnetplatten zum 30.4.83 aus seinem Programmangebot herauszunehmen.

Im folgenden sollen zur Erleichterung der Programmumstellung für die relativ häufig benutzten (unter den überhaupt noch aufgerufenen) Programme des SSP/FORTRAN denkbare Alternativen aus der NAG- oder IMSL-Bibliothek aufgezeigt werden, für die sich im Einzelfall eine Umstellung relativ unproblematisch gestalten sollte. Ein gewisser prinzipieller Unterschied zwischen SSP/FORTRAN-, NAG- und IMSL-Bibliothek ist dabei im Hinblick auf die Behandlung von mehrdimensionalen Feldern, die den Programmen durch Parameter übergeben werden, zu beachten:

- Im SSP/FORTRAN können Matrizen (zur wirtschaftlichen Ausnutzung des Speichers) unter Berücksichtigung spezieller Strukturen auf drei Arten gespeichert werden ("Storage mode"; vgl. SSP/FORTRAN-Dokumentation):

1. general: übliche spaltenweise Speicherung der Matrizen,
2. symmetric: als Vektor, der die verketteten Spalten der oberen Dreiecksmatrix enthält,
3. diagonal: als Vektor der Diagonalelemente.

Zum Zugriff auf beliebige Elemente der komprimiert gespeicherten Matrizen und zum Übergang zwischen verschiedenen Speicherungsformen existieren spezielle Hilfsfunktionen. Die SSP-Programme setzen für die übergebenen Matrizen entweder bestimmte Speicherungsformen voraus (z.B. ein Programm zur Addition zweier "general" Matrizen) oder können durch einen speziellen Parameter auf die benötigte Speicherungsform eingestellt werden.

- Für die NAG-Bibliothek existiert keine generelle Festlegung verschiedener Speicherungsformen für Matrizen mit spezieller Struktur. Soweit es durch die Problemstellung nahegelegt ist, werden Matrizen in sinnvoller Weise komprimiert gespeichert, und die entsprechenden Programmdokumentationen enthalten diesbezügliche Informationen (z.B. bei Bearbeitung symmetrischer Matrizen oder Bandmatrizen).

Mit Hilfe eines zusätzlichen Parameters, über den die bei der Deklaration festgelegte Zeilenzahl (Länge einer Spalte) der übergebenen Matrix zu spezifizieren ist, können NAG-Routinen im Gegensatz zu SSP-Routinen auch gezielt auf einer Teilmatrix "A(1:m, 1:n)" einer bei der Deklaration größer dimensionierten Matrix A(M,N),  $M > m$ ,  $N > n$  operieren. Damit wird eine größere Flexibilität erreicht, da nach einmaliger, ausreichend großer Dimensionierung der Matrix bei der Deklaration später Untermatrizen in problemsprechender, durch Parameter gewählter Größe bearbeitet werden können (vgl. Artikel "Matrizen als Parameter im SSP und im NAG", inform Jg.3, Nr.2, S.36ff).

- Für die IMSL-Bibliothek sind fünf verschiedene Speicherungsformen für speziell strukturierte Matrizen vorgesehen (vgl. Abschnitt 5.6 Matrix/Vector Storage Modes der "Introduction" im Band 1 der IMSL Library Reference Manuals):

1. Full Storage Mode:  
übliche spaltenweise Speicherung der Matrizen
2. Symmetric Storage Mode:  
als Vektor, der die verketteten Zeilen der unteren Dreiecksmatrix der eigentlichen symmetrischen Matrix enthält
3. Band Storage Mode:  
als Matrix, deren Spalten (im wesentlichen) die Hauptdiagonale und die Nebendiagonalen der eigentlichen Matrix mit Bandstruktur enthalten
4. Band Symmetric Storage Mode:  
wie Band Storage Mode, jedoch unter Ausschluß der oberen Nebendiagonalen
5. Hermitian Storage Mode:  
Speicherung komplexer Matrizen analog zum Symmetric Storage Mode bei reellen Matrizen

Zum Übergang zwischen verschiedenen Speicherungsformen und für Operationen mit Matrizen in unterschiedlichen Speicherungsformen existieren spezielle Routinen in der IMSL-Bibliothek. Programme, die entsprechend der Problemstellung auf Matrizen mit spezieller Struktur operieren, können deren Übergabe in einer adäquaten Speicherungsform voraussetzen.

IMSL-Routinen besitzen wie die NAG-Routinen zur Erhöhung der Flexibilität im Umgang mit Matrizen einen gesonderten Parameter, über den die bei der Deklaration gewählte Zeilenzahl der Matrix anzugeben ist.

Die folgende Aufstellung zeigt zu einigen SSP/FORTRAN-Routinen denkbare Alternativen aus der NAG- bzw. IMSL-Bibliothek auf. Dabei sind die Namen der zugehörigen SSP/FORTRAN-Routinen in doppelter Genauigkeit, soweit diese existieren, in Klammern zusätzlich angegeben. In der NAG-Bibliothek sind generell alle Programme in einer einfach genauen und in einer doppelt genauen Version verfügbar, so daß die Nennung der Version in doppelter Genauigkeit genügt. Die angesprochenen IMSL-Programme existieren ebenfalls alle in zwei Versionen.

#### SSP/FORTRAN

#### Alternativen

APCH, APFS  
(DAPCH, DAPFS)

---> NAG: E02ADF

Die NAG-Routine führt eine gewichtete Least-Squares Polynom-Approximation durch, wie sie durch Hintereinanderschaltung der SSP-Routinen APCH und APFS auch erreicht wird.

FMFP (DFMFP) ---> NAG: F04DBF  
IMSL: ZXCGR

Bestimmung eines lokalen Minimums einer Funktion mit  $n$  Veränderlichen. Die ersten partiellen Ableitungen sind vom Benutzer anzugeben.

QSF, QTFE, QIFG ---> NAG: D01GAF  
(DQSF, DQTFE, DQIFG)

Integration tabellierter Funktionen (Tafel enthält Argumente und zugehörige Funktionswerte). Die NAG-Routine basiert auf einer Methode von Gill und Miller (1972). Die (anders als beim SSP) nicht notwendig äquidistante Tabellierung der Funktion muß mindestens vier Punkte umfassen.

SINV (DSINV) ---> NAG: F01ABF, F01ACF, F01ADF  
IMSL: LINV1F, LINV2F, LINV3F

Invertierung einer symmetrischen positiv definiten Matrix. Die verschiedenen Routinen aus NAG- und IMSL-Bibliothek unterscheiden sich jeweils bezüglich der erzielbaren Genauigkeit der Ergebnisse und des Aufwands an Rechenzeit. Die Übergabe der zu bearbeitenden  $n \times n$ -Matrix  $A$  geschieht für SSP-, NAG- und IMSL-Routinen auf unterschiedliche Weise (s.o.):

SSP: Vektor mit Verkettung der Spalten der oberen Dreiecksmatrix von  $A$

IMSL: Vektor mit Verkettung der Zeilen der unteren Dreiecksmatrix von  $A$  (also identisch mit der SSP-Speicherungsform)

NAG: Vollständige  $n \times n$ -Matrix, wobei die obere Dreiecksmatrix als Eingabe dient, und das Ergebnis in der unteren Dreiecksmatrix zurückgegeben wird.

EIGEN ---> NAG: F02ABF, F01AJF mit F02AME  
IMSL: EIGRS

Eigenwerte und Eigenvektoren einer reellen symmetrischen Matrix. Bezüglich der Übergabe der Matrizen an die Unterprogramme gelten ähnliche Aussagen wie unter SINV.

GELB (DGELB) ---> NAG: F04LDF mit F01LBF  
IMSL: IEQT1E, LEQT2E

Lösung eines linearen Gleichungssystems  $Ax=B$  mit Koeffizientenmatrix  $A$  mit Bandstruktur und gegebenenfalls mehreren rechten Seiten. Unterschiedliche Übergabe der Bandmatrix  $A$  in SSP-, NAG- und IMSL-Routinen.

GEIG (DGELG) ---> NAG: F04AAF, F04AEF, F04AHF mit  
F03AFF, F04AYF mit F01BTF  
IMSL: LEQT1F, LEQT2F

Lösung eines linearen Gleichungssystems  $Ax=B$  mit beliebiger  $m \times m$ -Koeffizientenmatrix A und  $m \times n$ -Matrix B mit n rechten Seiten.

GMPRD ---> NAG: F01CKF  
IMSL: VMULFF

Multiplikation zweier (vollständig gespeicherter) Matrizen.

GMTRA ---> NAG: F01CRF  
IMSL: VTRAN

Transponieren einer (vollständig gespeicherten) Matrix.

HARM (DHARM) ---> NAG: C06ADF  
IMSL: FFT3D

Komplexe Fourier-Transformation eines komplexen dreidimensionalen Feldes. Die Datenübergabe geschieht bei den verschiedenen Routinen auf unterschiedliche Weise, so daß beim Austausch eine Umorganisation notwendig wird:

SSP: Vektor vom Typ REAL

NAG: Je ein Vektor vom Typ REAL für Imaginär- und Realteile

IMSL: Dreidimensionale Matrix vom Typ COMPLEX

Die NAG-Routine ist dreimal hintereinander anzuwenden, wobei die drei Indizes der Matrix jeweils einzeln als variabel betrachtet werden.

MINV (DMINV) ---> NAG: F01AAF, F04AEF  
IMSL: LINV1F, LINV2F, LINV3F

Invertierung einer beliebigen (nicht singulären) reellen Matrix. Die angegebenen NAG- und IMSL-Routinen unterscheiden sich jeweils hinsichtlich erzielbarer Genauigkeit der Ergebnisse und Rechenzeit- sowie Speicherplatzbedarf.

RANDU ---> NAG: G05CAF  
IMSL: GGUBS, GGUD, ...

Erzeugung gleichverteilter Zufallszahlen. In der NAG- und vor allem der IMSL-Bibliothek existieren (zahlreiche) qualitativ hochstehende Programme, die zum Ersatz der unzuverlässigen SSP-Routine RANDU geeignet sind.

VARMX ---> IMSL: OFROTA, OFSCHN

Orthogonale Rotation einer Matrix mit Faktorladungen im Rahmen einer Faktorenanalyse. OFROTA bietet neben der in VARMX angewendeten Varimax-Methode auch wahlweise eine Quartimax- oder Equamax-Methode.

Bei Verwendung von OFSCHN kann bei Vorliegen entsprechender Kenntnisse eine hypothetische rotierte Matrix mit Faktorladungen vorgegeben werden, an der sich die anschließende orthogonale Rotation der eigentlichen (unrotierten) Matrix orientiert (z.B. zur besseren Interpretierbarkeit der Faktorladungen oder zur Verfeinerung der Ergebnisse von OFROTA).

Detaillierte Informationen zu den einzelnen Programmen können den Dokumentationen zu den genannten Programmbibliotheken entnommen werden.

### BMDP-Programmsystem, Version 1982

F.-P. Spellmann

Die Programme des Statistik-Programmsystems BMDP (BioMedical Computer Programs) stehen seit kurzem in der Version von 1982 zur Verfügung. Die neue Version stellt im Gegensatz zu der von 1981, in der 6 neue Programme zum Einsatz kamen, im wesentlichen eine "Wartungsversion" dar, die die bekannten Programme in korrigierter und gegebenenfalls weiterentwickelter Form umfaßt. Nähere Einzelheiten über die Neuerungen zu einem bestimmten Programm werden bei Ausführung dieses Programms unter Angabe von "/PRINT NEWS." ausgegeben.

Zusätzlich im Programmsystem enthalten ist nun das schon in der "alten" Literatur (Dixon: BMDP Statistical Software, 1981 Edition) dokumentierte Programm BMDP1T zur univariaten und bivariaten Spektralanalyse.

Daneben versprechen einige weitere Neuerungen ein komfortableres Arbeiten mit dem Programmsystem. So wurde die Überprüfung der vom Benutzer einzugebenden Programme auf syntaktische Fehler in der Weise verbessert, daß auch trotz zwischenzeitlich erkannter Fehler das gesamte Programm analysiert wird. Darüber hinaus wurde die Fehlerdiagnostik erweitert. Zur Vereinfachung der Erstellung umfangreicher Benutzerprogramme wurde die Möglichkeit geschaffen, Programnteile aus einer externen Datei in Benutzerprogramme zu integrieren oder mittels einer speziellen Schleifenstruktur zu erreichen, daß ein Block von Anweisungen in eine Folge ähnlicher, inhaltlich durch eine Laufvariable jeweils modifizierter Blöcke überführt (z.B. für die Transformation jeweils einer Variablen) und anschließend als Programmtext interpretiert wird.

Zur Dokumentation der BMDP-Programme steht neuerdings neben dem oben angesprochenen Buch von Dixon der BMDP User's Digest, 2nd Edition, in der Bibliothek des Rechenzentrums zur Verfügung. Es handelt sich dabei um eine kompakte Zusammenstellung der Elemente der BMDP-Kontrollsprache und der notwendigen und optionalen Parameter für die einzelnen BMDP-Programme.

NAG-Bibliothek

F.-P. Spellmann

An den Routinen C02ADF, C02AEF, D02GAF, D02GEF, L02HEF, D02RAF und F02GJF (sowie den zugehörigen Programmen in einfacher Genauigkeit) der NAG-Bibliothek (Mark 9) wurden zwischenzeitlich Korrekturen gemäß einer von der Numerical Algorithms Group herausgegebenen Fehler- und Korrekturliste durchgeführt, um die seit der Einführung der gegenwärtigen Version der NAG-Bibliothek erkannten Fehler schon vor dem Einsatz der Version Mark 10 zu eliminieren.

Zu den beseitigten Fehlern ist zu bemerken, daß mit Ausnahme des Fehlers in der Routine F02GJF (F02GJE), der unter Umständen zur Ausgabe von Eigenwerten mit falschem Vorzeichen führte, in den übrigen Fällen keine "versteckten" fehlerhaften numerischen Ergebnisse zu erwarten waren, sondern dort kam es gegebenenfalls zu offensichtlichen Fehlersituationen (Overflow, Fehlermeldung über den IFAIL-Parameter).

Statistical Analysis System SAS

S. Zörkendörfer

Das Rechenzentrum hat das "Statistical Analysis System SAS" in seiner Grundversion angemietet. Implementiert ist derzeit die Version 79.6, sie kann durch die katalogisierte Prozedur SAS aufgerufen werden. Die Folgeversion SAS82 ist angekündigt, Benutzerhandbücher hierzu sind bereits lieferbar.

Anwendungsschwerpunkte der SAS-Programmierung liegen einerseits auf dem Gebiet der Aufbereitung von Daten, Erstellung und Manipulation von Dateien und dem Ausdruck von Listen und Tabellen, zum anderen wird eine Vielfalt von Prozeduren zur deskriptiven und multivariaten Statistik angeboten, aber auch Dienstprogramme und etwa eine Matrixsprache werden bereitgestellt.

Im Rahmen einer Lehrveranstaltung des Wintersemesters gebe ich eine Einführung in die SAS-Programmierung (Mi 9 - 11, Hörsaal M6).

HSM - Ein neues Programm zur Verwaltung  
privater Magnetplattendateien

F.-P. Spellmann

Seit einiger Zeit wird ein weiteres Programmprodukt der Firma IBM, der HSM (Hierarchical Storage Manager), am Rechenzentrum eingesetzt. Zweck dieses Programmes ist es, die Verwendung des für Benutzer des Rechenzentrums bereitstehenden Magnetplattenplatzes für private Dateien (ULIBO!) zu optimieren.

Die grundlegende Aktivität des HSM in diesem Zusammenhang besteht aus einer täglichen automatischen Auslagerung (Auto-migration) solcher Benutzerdateien, die bisher aufgrund längerer "Nichtnutzung" gelöscht worden wären, in platzsparender Form auf andere Plattenbereiche. Im Rahmen der vorhandenen Möglichkeiten werden die migrierten (und weiterhin nicht benutzten) Dateien erst nach Ablauf weiterer zwei bis drei Wochen gelöscht.

Migrierte Dateien sind bei Verwendung des RUMSERV-Kommandos LISTDS anhand einer speziellen Meldung zu erkennen ("DATASET MIGRATED..."), die so lange erscheint, bis die Dateien zurückgerufen oder endgültig gelöscht sind. Sie werden bei der Berechnung des belegten Plattenplatzes nicht berücksichtigt. Die RUMSERV-Kommandos PROTECT und RENAME können auf migrierte Dateien nicht unmittelbar angewendet werden, sondern erst nach Rückruf der Dateien, der z.B. durch das RUMSERV-Kommando VERIFY veranlaßt werden kann. Ansonsten ist die Benutzung migrierter Dateien nicht beeinträchtigt, da sie bei erneutem Zugriff ohne Zutun des Benutzers vom HSM sofort in ursprünglicher Form wiedererstellt werden.

Die durch das Migrationsverfahren angestrebte Platzersparnis wird durch eine vom HSM ausgeführte Komprimierung der Daten erreicht. Zusätzlich verwaltet der HSM sehr kleine Dateien, die die ihnen vollständig zugeordnete Magnetplattenspur eventuell nur partiell nutzen, als Teile einer speziellen VSAM-Datei (Small-data-set-packing data set). Zur Optimierung der Datei-Zugriffe werden schließlich im Rahmen der beschriebenen Auto-migration Dateien mit einer großen Anzahl zugehöriger, an verschiedenen Stellen der Magnetplatte lokalisierter Bereiche (Extents) automatisch zu einer Einheit zusammengefaßt.

Neben der beschriebenen Automigration leistet der HSM zur Vorbeugung von Störfällen eine Sicherung der von ihm verwalteten Magnetplatte ULIBO! durch Anlegen von Dateikopien auf Magnetbändern. Die damit verbundenen Möglichkeiten können jedoch erst sinnvoll nach Beschaffung zuverlässiger Magnetbandeinheiten genutzt werden.

Ergänzend ist noch zu bemerken, daß sich die veröffentlichten Konventionen des Rechenzentrums über die Benutzung privater Dateien auf Magnetplatten durch den Einsatz des HSM nicht geändert haben. Insbesondere werden also unzulässige Dateien (z.B. wegen zu geringen Füllungsgrades) nach wie vor gelöscht.

Betriebliche Mitteilungen

K. Reichel

## 1. Neue Magnetbandstationen

Das Rechenzentrum will im Jahr 1983 die jetzigen, nicht mehr voll funktionstüchtigen Magnetbandstationen der Firma Telex/Memorex, die 1974 installiert wurden, aber bereits damals nicht mehr dem neuesten Stand der Technik entsprachen, durch modernere Bandstationen ersetzen. Die Neuinstallation wird z.Zt. so weit vorbereitet, daß sie unmittelbar nach Verfügbarkeit der Haushaltsmittel 1983 vorgenommen werden kann.

Während die derzeit betriebenen Bandstationen eine Schreibdichte wahlweise von 800 oder 1600 bpi zulassen, werden die neuen Stationen nur noch eine Wahl zwischen 1600 und 6250 bpi gestatten. Alle Benutzer, die noch mit 800-bpi-Erändern arbeiten, werden gebeten, sich umgehend mit dem Rechenzentrum in Verbindung zu setzen (Herr Goorkotte, Tel. 2672).

## 2. Wochenendbetrieb

Seit einiger Zeit bietet das Rechenzentrum einen Wochenendbetrieb an. Bedienungspersonal ist allerdings in dieser Zeit nicht anwesend, das Rechenzentrumsgebäude bleibt geschlossen. Deshalb können nur Jobs ohne manuelle Eingriffe bearbeitet werden, und die Jobeingabe und -ausgabe ist nur über externe Datenstationen möglich. Es wäre wünschenswert, wenn die dadurch verfügbare Rechenzeit noch besser genutzt würde. Besonders rechenintensive Jobs werden sehr viel zügiger bearbeitet, aber auch das Antwortzeitverhalten im Dialogbetrieb ist in dieser Zeit erheblich besser als im stark überlasteten Wochenbetrieb.

## 3. Hauptspeichererweiterung

Das Rechenzentrum hat am 23.10.1982 den Hauptspeicher testweise erweitert. Wir erwarten, daß wir unseren Benutzern somit mehr Rechnerleistung bieten können, da vor allem die durch den Paging-Mechanismus verursachten Plattenzugriffe stark zurückgehen dürften.

Personalia

Zum 1.9.1982 wurden Frau I. Bartodziej sowie die Herren K. Henningsen und M. Steinke als studentische Mitarbeiter am Rechenzentrum eingestellt. Herr K. Fusenig ist seit dem 1.10.1982 ebenfalls als studentischer Mitarbeiter bei uns tätig.

Herr K. Elix, der seit über zwei Jahren als wissenschaftlicher Mitarbeiter in der Abteilung Systemsoftware tätig war, nachdem er schon als studentischer Mitarbeiter im Rechenzentrum gearbeitet hatte, hat uns zum 1.10.1982 verlassen, um eine Stelle in der freien Wirtschaft anzutreten. Wir wünschen ihm alles Gute und viel Erfolg.

Leserforum

Ab wann kann man wieder dem FORTRAN-Compiler vertrauen? Die Tatsache, daß das "Computed GOTO" nicht funktioniert, erweckt den Verdacht, daß auch andere, nicht so einfach zu entdeckende Fehler vorhanden sind.

Hin und wieder kommt es mir so vor, als ob man den EZ-Rechner auch gut durch ein größeres Ensemble Würfel ersetzen kann.

R. Stademann, Institut für  
Theoretische Physik I

Es ist mittlerweile bekannt, daß der von der IBM ausgelieferte VS FORTRAN-Compiler (Release 1) einen zweifelhaften Ruf erworben hat. Beschwerden darüber sind nicht nur durch die betroffenen Rechenzentren und Benutzergruppen, sondern auch auf internationaler Ebene vom FORTRAN Project und vom SEAS Executive Board an die IBM herangetragen worden. (SEAS ist die europäische Benutzerorganisation für IBM-Großrechenanlagen, der unser Rechenzentrum auch angehört.) Auch in den USA gab es Proteste. Die Herstellerfirma ist also bestens "im Bilde".

IBM hat nun auf die vielfältigen Interventionen mit der Ankündigung von VS FORTRAN Release 2.0 reagiert, das neben der Behebung festgestellter Fehler oder Mängel auch Erleichterungen bei der Rückverfolgung von Fehlern (diesmal solche im Benutzerprogramm!) verspricht. Außerdem soll diese Version, die uns für Dezember 1982 angekündigt worden ist, dem Standard FORTRAN 77 genügen!

W. Bosse

Wie bereits mehrfach ausgeführt, kann das Rechenzentrum Fehler in IBM-Programmprodukten nicht selbst beheben. Diese Programme werden uns nur als Objektprogramme zur Verfügung gestellt; die Quellen besitzen wir allenfalls in Form von Microfiches, also nicht maschinenlesbar, und zudem nicht in der allerneuesten Version. Dagegen haben wir einen vertraglich abgesicherten Anspruch auf Behebung erkannter Fehler durch IBM, den wir bei eigenmächtigen Änderungen gefährden würden.

Grundsätzlich melden wir erkannte Fehler umgehend den entsprechenden Stellen der Firma IBM. Auf die Maßnahmen innerhalb von IBM, insbesondere die Bearbeitungsgeschwindigkeit, haben wir jedoch nur geringen

Einfluß, konkret gesagt, wir können uns allenfalls energisch beschweren. Das haben wir im vorliegenden Fall mehrfach getan.

Normalerweise ist die Bearbeitungszeit für gemeldete Fehler sehr kurz, weil bei Standardsoftware die Chance, daß wir die ersten sind, denen ein Fehler auffällt, gering ist. Meist wurde der Fehler bereits von anderen Kunden gemeldet und die notwendigen Programmverbesserungen sind bereits lieferbar. Für viele unserer Benutzer vielleicht überraschend gehört jedoch der FORTRAN-Compiler keineswegs zu den besonders intensiv genutzten Programmen, zumindest im Vergleich aller IBM-Installationen. Wissenschaftliche Datenverarbeitung ist eben mengenmäßig von nicht sehr großem Gewicht. Zudem wurde der VS-FORTRAN-Compiler, wie inzwischen auch die Firma IBM eingesteht, in sehr unreifem Zustand ausgeliefert. Folglich waren wir diesmal häufig der Kunde, der einen Fehler erstmalig entdeckte und damit auf die Fehleranalyse und -behebung durch ein im Falle FORTRAN offenbar völlig überlastetes "Change Team" warten mußte. Die Firma IBM hat angedeutet, daß firmenintern Konsequenzen aus dieser Misere gezogen würden. Uns hilft das aktuell nicht.

Nachzutragen bleibt noch, daß von der ersten Beschwerde eines Benutzers bis zur Erkennung des Fehlers durch uns auch eine gewisse Zeit vergehen kann, die abhängt davon, wie weit der Benutzer sich bereits bemüht hat, den Fehler einzugrenzen. Häufig werden wir mit einem unübersichtlichen Programm aus mehreren tausend Befehlen und der Behauptung konfrontiert, dieses sei früher immer gelaufen, liefere jetzt aber falsche Ergebnisse. Wir können solche "Fehlermeldungen" verständlicherweise nicht so an IBM weiterleiten und bitten daher dringend darum, aufgetretene Fehler so weit wie möglich einzugrenzen. Das kann derjenige am besten, der die Haken und Ösen der betroffenen Programme aus erster Hand kennt.

R. Schmitt

Zitat aus der Zuschrift eines Benutzers im VSPC-"Briefkasten" 4400:

"Mitarbeiter der Systemgruppe der IBM ..."

Spätestens unsere monatlichen Gehaltsmitteilungen erinnern uns daran, daß wir "nur" Mitarbeiter der Universität sind.

R. Schmitt

### Vorschlag für eine einheitliche Schreibweise von PL/I-Programmen

E. Sturm

Wer oft mit fremden Programmen arbeitet und vor allem auch versucht hat, diese zu verändern, hat erfahren müssen, daß jeder Programmierer andere Konventionen benutzt, um Ordnung in seine Programme zu bringen. Auch bei eigenen Programmen beobachtet man mit der Zeit Änderungen in der äußeren Form. Mit den im folgenden aufgestellten Regeln hoffe ich dazu beizutragen, daß manch einer schon früher zu einer "optimalen" Form findet. Die eingerückten Absätze sind Erklärungen zu den Regeln.

Allgemeiner Grundsatz ist: Es kommt nicht darauf an, wie schnell ein Programm "runtergetippt" ist (das geschieht nur einmal!), sondern darauf, wie leicht es danach gelesen und verändert werden kann (nach aller Erfahrung viele Male!).

REGEL 1: /\* GLIEDERUNG DES GESAMT-PROGRAMMS \*/

Das Programm ist so zu gliedern, daß eine Programmeinheit auf eine Seite paßt.

Die ausführbaren Anweisungen eines externen Programms oder eines internen Unterprogramms oder auch der Deklarationsteil sind eine solche Programmeinheit, die dann ohne Hin- und Herblättern durchdacht werden kann. Wächst eine Programmeinheit über eine Seite hinaus, so besteht immer die Möglichkeit, einen Abschnitt als internes Unterprogramm auszugliedern.

REGEL 2: /\* GLIEDERUNG EINER SEITE \*/

Eine Programmeinheit ist so in Abschnitte zu gliedern, daß vor jedem Abschnitt 2 Leerzeilen stehen, sowie ein rechtsbündiger Kommentar den Sinn des Abschnitts erklärt.

Dies erlaubt, den Inhalt einer Seite zu verstehen, indem man nur die von den Anweisungen klar getrennten Kommentare durchliest.

REGEL 3: /\* GLIEDERUNG EINES ABSCHNITTS \*/

Innerhalb eines Abschnitts wird dadurch gegliedert, daß in sinnvoller Weise jeweils 3 Spalten eingerückt wird.

DO-Gruppe:

- die ausführbaren Anweisungen und das zugehörige END 3 Spalten,

IF-Anweisung:

- THEN bzw. THEN DO 3 Spalten, ELSE ebenso,
- die ausführbare(n) Anweisung(en) (und END) 6 Spalten,

## SELECT-Gruppe:

- WHEN, WHEN DO bzw. OTHER(WISE), sowie Gruppen-END 3 Spalten,
- die ausführbare(n) Anweisung(en) (und END) 6 Spalten.

Diese Definition ermöglicht es, ohne Hin- und Herrücken ganzer Abschnitte Anweisungen einzufügen oder zu streichen. Das END ist deshalb mit eingerückt, weil die Programmstruktur ja durch Einrücken schon klar ist, eine weitere auffallende Klammerung durch DO - END ist da nur störend.

## REGEL 4: /\* GLIEDERUNG DURCH BLANKS \*/

Operatoren, das Zuweisungszeichen sowie Klammerangaben sind durch 1 Blank links und rechts abzusetzen. Ausnahme: Arrayname und Indexausdruck stehen direkt hintereinander, Format und Klammerausdruck ebenso. Vor einem Komma oder Semikolon steht allerdings kein Blank.

Variablennamen sind besser lesbar, wenn sie nicht in einem Wust von Symbolen stehen.

## REGEL 5: /\* FORTSETZUNG \*/

Wird eine Anweisung fortgesetzt, so sind weitere 6 Spalten einzurücken.

## REGEL 6: /\* LABEL \*/

Label-Namen stehen ggfs. mit Kommentar auf einer Extrazeile.

Gemeint sind an sich nur Programmnamen und Marken vor DO-Gruppen bei Verwendung von LEAVE-Anweisungen.

Die folgenden Regeln sind mehr inhaltlicher Natur.

## REGEL 7: /\* DEKLARATION \*/

Sämtliche Variablen sind zu deklarieren, die Variablennamen sind sinnvoll zu wählen.

Der Deklarationsteil steht am besten nach allen ausführbaren Anweisungen auf einer neuen Seite, da man ihn nur zur Kontrolle gebraucht, die ausführbaren Anweisungen jedoch oftmals überdenkt. Bewährt hat sich eine Aufteilung nach Parameter, EXTERNAL, BASED, CONTROLLED, STATIC, AUTOMATIC, sowie Programmsteuerungsdaten (ENTRY, FILE usw.).

Kommt eine Variable nicht sehr häufig vor, dann ist ein Name zu suchen, der die genaue Verwendung beschreibt (bis zu 31

Zeichen sind möglich!). Wenn ein Name dagegen sehr oft im Programm auftaucht, reicht es, eine Abkürzung zu verwenden. Dies kommt den Fähigkeiten des menschlichen Gedächtnisses entgegen.

Für externe Unterprogrammnamen, die bekanntlich nur 7 Zeichen lang sein dürfen, läßt sich das GENERIC-Attribut (mit nur einer WHEN-Angabe) verwenden, damit ein lesbarer Name vereinbart werden kann.

#### REGEL 8:

/\* SPRÜNGE \*/

GOTO-Anweisungen dienen nur zum abnormalen Verlassen einer ON-Unit. LEAVE ist pro Schleife nur einmal zu verwenden.

Unübersichtlich sind nicht Sprünge an sich, sondern die Tatsache, daß man bei einem Label nie weiß, von wo man wohl an diese Stelle gesprungen ist und welchen Wert dann irgendwelche Variablen haben. Die Benutzung der LEAVE-Anweisung bleibt übersichtlich, wenn man genau weiß, daß man nur von einer LEAVE-Anweisung hinter eine DO-Gruppe gesprungen sein kann.

#### REGEL 9:

/\* AUSDRÜCKE STATT VARIABLEN \*/

Man versuche, mit möglichst wenig Variablen auszukommen. BUILTIN-Funktionen oder Ausdrücke sind besser als neue Variablen.

Mit der Anzahl der Variablen wächst die Möglichkeit, aus Versehen eine Variable für mehr als einen Zweck zu benutzen. Bei einem optimierenden Compiler werden Ausdrücke sowieso nur einmal ausgewertet.

#### REGEL 10:

/\* PARAMETER \*/

Interne Unterprogramme sollen nach außen entweder über Parameter oder über globale Variablen kommunizieren, aber nicht über beides.

Entweder man benutzt interne Unterprogramme, um nach Regel 1 ein Programm übersichtlich zu halten, dann sollten keine Parameter vorkommen. Oder das interne Unterprogramm ist in sich abgeschlossen, verarbeitet die übergebenen Parameter zu neuen Werten, dann machen Seiteneffekte über globale Variablen das Programm nur unverständlich.

Als Beispiel sei hier ein Programm vorgestellt, das den aus inform Jg. 6, Nr. 3 bekannten DIGITAB-Output zum Erzeugen eines Bildes verwendet. Wegen der Kürze des Programms sind die Regeln, die sich auf die Seitenaufteilung beziehen, hier nicht sichtbar.

```
BILD:          /* ERZEUGUNG EINES PLOTS AUS DIGITAB-DATEN */
```

```
PROCEDURE OPTIONS (MAIN);
```

```
/* INITIALISIERUNGEN */
```

```
ON ENDFILE (SYSIN) DATEN_DA = '0'B;
```

```
CALL PLOTS;
```

```
/* EINLESEN DER DIGITAB-DATEN UND AUFRUF DER */
```

```
/* ENTSPRECHENDEN PLOT-ROUTINEN */
```

```
GET EDIT (TYP) (A(1));
```

```
DO WHILE (DATEN_DA);
```

```
IF TYP = 'T'
```

```
THEN DO;
```

```
/* TYP TEXT */
```

```
GET LIST (TEXT);
```

```
CALL PCHARS (LENGTH (TEXT), TEXT);
```

```
END;
```

```
ELSE DO;
```

```
GET LIST (X, Y);
```

```
SELECT (TYP);
```

```
WHEN ('M')
```

```
/* TYP MOVE */
```

```
CALL PLOT (TRANS (X), TRANS (Y), 3);
```

```
WHEN ('D')
```

```
/* TYP DRAW */
```

```
CALL PLOT (TRANS (X), TRANS (Y), 2);
```

```
OTHER DO;
```

```
/* DATEN NICHT FEHLERFREI */
```

```
PUT LIST ('*** UNGUEITIGER TYP:', TYP);
```

```
RETURN;
```

```
END;
```

```
END;
```

```
END;
```

```
GET EDIT (TYP) (A(1));
```

```
END;
```

```
CALL PLOT;
```

```
/* BEENDIGUNG DES PLOTS */
```

```
RETURN;
```

```

TRANS:      /* ABBILDUNG DES DIGITAB-BEREICHES AUF 20 X 20 CM */
PROCEDURE (X) RETURNS (FLOAT);

DCL X BIN FIXED;

RETURN (X * 20E0 / 1023);

END TRANS;

DCL (X, Y) BIN FIXED;
DCL TYP CHAR;
DCL TEXT CHAR (100) VAR;
DCL DATEN_DA BIT INIT ('1'B);
DCL (SYSIN, SYSPRINT PRINT) FILE;
DCL LENGTH BUILTIN;
%INCLUDE PLOT; /* ALLE FUER PLOT BENÖTIGTEN DEKLARATIONEN */

END BILD;

PLOT-PREVIEW 2

```

E. Sturm

Gleichzeitig mit dem Umzug der graphischen Terminals in den neuen Raum im Flur der 1. Etage des Mathematik-Hochhauses wird auch eine weitere Version des Plot-Preview-Programms zur Verfügung stehen. Zur Bildausgabe wird hierbei nicht das Gerät TEKTRONIX 4015, sondern das an dem IBM-Terminal 3277 angeschlossene TEKTRONIX 618 benutzt. Der Dialog mit dem Benutzer (welches Bild wie gezeichnet werden soll) läuft nicht über den graphischen Bildschirm, sondern über den alphanumerischen Bildschirm IBM 3277. Ein besonderer Vorteil ist die deutlich höhere Übertragungsgeschwindigkeit, die es erlaubt, selbst größere Bilder in wenigen Sekunden sichtbar zu machen. Eine weitere Verbesserung ist die Möglichkeit, sich einen beliebigen Ausschnitt aus dem Originalbild vergrößert anzuschauen.

Neu ist zunächst aber die Art, wie der Benutzer mit dem PLOT-PREVIEW-Programm in Verbindung tritt. PLOT-PREVIEW 2 (PPV) ist ein VTAM-Anwendungsprogramm wie VSPEC, man kann also wahlweise mit VSPEC arbeiten oder sich mit PPV ein Bild anschauen. Sollte noch das VM/370-Bild zu sehen sein, muß man sich (wie auch für VSPEC) zunächst an MVS anwählen. Dies geschieht durch Drücken der ENTER-Taste und die Eingabe

```
dial mvs ac7
```

(mit anschließend erneutem Drücken der ENTER-Taste).

Danach sollte der Cursor am Anfang der fünften Zeile stehen als Zeichen dafür, daß VTAM ein Logon erwartet. An dieser Stelle könnte man VSPC ID=... eingeben oder aber auch

#### PPV

(danach wieder die ENTER-Taste drücken). Für PPV sollte man schon vorher den graphischen Bildschirm anschalten und, wenn dieser völlig hell geworden ist, einmal löschen.

PPV meldet sich mit einem "Abfragebild" auf der IBM 3277. Man wird gebeten, den Jobnamen und die Jobnummer des Jobs einzugeben, dessen Bilder man anschauen möchte. Wahlfrei sind die Angaben, ob man um das fertige Bild einen Rahmen gezeichnet haben möchte (zur Kennzeichnung der zur Verfügung stehenden Fläche), bzw. ob man das Bild (nur bei PLOT-PREVIEW, nicht später in der Plotterzeichnung) in maximaler Größe sehen möchte. Hat man alle Angaben eingetragen bzw. mit einem x "angekreuzt", drückt man die ENTER-Taste. PPV sucht dann das erste Bild dieses Jobs und zeichnet es auf dem graphischen Bildschirm.

Danach erscheint ein "Auswahlbild" auf dem alphanumerischen Bildschirm. Man kann nun wählen, was mit dem Bild weiterhin geschehen soll. Entweder kann eine Kopie auf dem Hardcopygerät angefertigt werden, oder man kann eine Plotterzeichnung veranlassen. Zusätzlich besteht die Möglichkeit, sich einmal (!) einen Ausschnitt zeigen zu lassen.

Hat man "Ausschnitt" angekreuzt, erscheint auf dem graphischen Bildschirm ein Fadenkreuz, das man mit dem "joy-stick" verschieben kann. Befindet sich das Fadenkreuz an einer Ecke des gewünschten Ausschnitts, so drückt man irgendeine Zeichentaste (z.B. die Leertaste). PPV kennzeichnet diese Position (ohne daß man die ENTER-Taste drückt) mit einem punktierten Fadenkreuz. Die gegenüberliegende Ecke des Ausschnittes ist danach auf die gleiche Weise zu markieren. PPV legt dann diesen Ausschnitt unverzerrt so in die verfügbare Bildschirmfläche, daß die linken unteren Ecken übereinstimmen.

In jedem Fall kommt nach dem Auswahl- das Abfragebild wieder. Wurden mehrere Bilder in einem Job erzeugt, sind Jobname, Jobnummer und die alten Setzungen für Rahmen und maximale Bildgröße schon eingetragen, man braucht nur die ENTER-Taste zu drücken und das nächste Bild wird auf dem graphischen Bildschirm gezeichnet. Ändert man in diesem Zustand des PPV-Programms Jobname oder Jobnummer, so werden alle weiteren Bilder, die in demselben Jobstep erzeugt wurden, gelöscht, und es wird nach einem Bild des neuen Jobs gesucht. Hat man alle Bilder gesehen und wartet auch kein weiterer PPV-Interessent, so kann man durch Drücken der CLEAR-Taste an der IBM 3277 die Verbindung zu PPV lösen, das Terminal ist danach z.B. wieder für VSPC verfügbar.

Die Benutzung der beiden graphischen Terminals wird vorläufig in folgender Weise geregelt: Das Tektronix 4015 wird wie bisher vor 16 Uhr vorrangig für das alte PLOT-PREVIEW und danach vorrangig für interaktive Arbeiten (mit PLOT10 bzw. CPLOT) benutzt. An der IBM 3277 mit graphischem Anschluß (TEKTRONIX 618) haben interaktive graphische Arbeiten Vorrang vor dem neuen PLOT-PREVIEW. In der Praxis sollte immer auf dem Gerät PLOT-PREVIEW gemacht werden, das gerade nicht für interaktive Arbeiten benötigt wird.

An dieser Stelle soll auch noch auf den Plot-Überblick aus inforum Jg.6, Nr.1 hingewiesen werden. Jener Artikel, in dem das alte PLOT-PREVIEW, das Anschauen von Bildern in VSPC und das PLOTSERV-Programm (zum sofortigen Anfertigen eines Plots) beschrieben sind, hat auch weiterhin Gültigkeit.

#### Spezielle Wünsche bei der Druckausgabe

K.-B. Mertz

Wer von seinen Programmen zum Druck bestimmte Ausgabedateien erzeugen läßt, sollte wissen, wo und wie diese gedruckt werden. Er muß aber auch darüber informiert sein, welche Alternativen vom Rechenzentrum angeboten werden und wie er diese anfordern kann. Diese Informationen sollen im folgenden zusammengestellt werden.

Das im Rechenzentrum eingesetzte Betriebssystem MVS/SE bedient sich des Job Entry Subsystems JES2, um die Lochkarten-Leser, -Stanzer und Drucker im Rechenzentrum und an den Stapelstationen zu steuern und die anfallenden Ausgabedateien entsprechend den gewünschten Spezifikationen zu bearbeiten. Im JES2 werden die Anforderungen für die Ausgabe durch fünf verschiedene Parameter spezifiziert:

- SYSOUT - Klasse der Ausgabedatei,
- DEST - Ort der Ausgabe,
- FORMS - Art, Format und Vordruck des Papiers sowie Art des Drucktuchs,
- UCS - Drucktypen und Zeichenvorrat,
- FCB - Vorschubsteuerung.

Alle Ausgabedateien mit gleichen Parameterwerten werden in jeweils eine eigene Warteschlange eingeordnet, die erst dann abgearbeitet wird, wenn der Drucker am Ort der Ausgabe entsprechend ausgerüstet ist und dies dem JES2 durch eine entsprechende Eingabe vom Operateur bzw. Bediener der Remote-Station bekanntgemacht wurde.

### 1) SYSOUT (Ausgabeklasse)

Folgende Ausgabeklassen werden im Rechenzentrum durch die JCL (Job Control Language) unterstützt:

- A - Ausgabe auf einem Drucker,
- B - Ausgabe auf einem Lochkartenstanzer,
- E - Dummy-Ausgabe, die sofort nach Ende der Ausführung des Jobs gelöscht wird (d.h. Unterdrückung der Ausgabe),
- P - Ausgabe auf einem Plotter oder Behandlung durch PICT-PREVIEW oder PLOTSERV (nur in Verbindung mit bestimmten FORMS-Parametern),
- Q - "gehaltene" Ausgabe zur Behandlung durch VSPC-Kommandos wie LCAD OUTPUT oder ROUTE,
- C - Ausgabe auf Disketten (nur in Verbindung mit dem ISIL-Parameter).

Die Ausgabeklasse wird für die System-Ausgabedateien (Job-Log, JCL-Liste, Meldungen des JCL-Konverters) festgelegt durch den Parameter MSGCLASS in der Jobanweisung. Fehlt dieser, so ist die Art der Eingabe ins System und der Jobtyp maßgebend: Wurde der Job in Form von Lochkarten oder von einer Diskette eingelesen, wird MSGCLASS=A angenommen; bei Jobs, die aus dem VSPC durch das Kommando SUBMIT ins System kommen, wird MSGCLASS=Q angenommen. Ausnahmen sind Jobs der Typen LIST, DUPL und VSPC, bei denen unabhängig von der Art der Eingabe MSGCLASS=A, E bzw. C gesetzt wird.

Die Ausgabeklasse aller anderen Ausgabedateien wird im SYSOUT-Parameter der entsprechenden DD-Jobkontrollanweisung angegeben, wobei ein Stern (\*) bedeutet, daß die Klasse mit der MSGCLASS des Jobs übereinstimmen soll. In den katalogisierten Prozeduren sind in der Regel symbolische Parameter (&CPRINT, &LPRINT, &EPRINT) vorgesehen, um dem Benutzer Änderungen gegenüber dem Standard zu erleichtern.

Es sei darauf hingewiesen, daß das VSPC-Kommando ROUTE die Ausgabeklasse aller gehaltenen Dateien des Jobs oder der mit DS spezifizierten Datei neu festlegt und die Datei zur entsprechenden Weiterverarbeitung freigibt. Normalerweise wird als neue Ausgabeklasse A angenommen, doch kann man durch die Angabe von CLASS x eine neue Ausgabeklasse x (aus der o.g. Liste) bestimmen.

### 2) DEST (Ort der Ausgabe)

Als Ort der Ausgabe für Dateien der Ausgabeklasse A wird zunächst der Ort der Eingabe angenommen, also das Rechenzentrum oder eine Stapelstation für Jobs, die in Form von Lochkarten eingegeben wurden, bzw. das Rechenzentrum für Jobs von Disketten oder aus dem VSPC. Diese Annahme kann für einen ganzen Job oder eine einzelne Datei außer Kraft gesetzt werden, wie weiter unten beschrieben wird. Da Lochkartenstanzer, Plotter und Disketten-Ein/Ausgabegerät nur im Rechenzentrum vorhanden sind, kommt für

Dateien der Ausgabeklassen B, P und O kein anderer Ort der Ausgabe in Frage.

Daß die gesamte Ausgabe der Klasse A für einen Job an einer bestimmten Stelle gedruckt werden soll, kann man durch Einfügen einer Anweisung

```
/*ROUTE PRINT ort
```

(mit den Zeichen /\* in den ersten beiden Spalten) zwischen der Job- und der nächsten Jobkontrollanweisung vereinbaren, wobei als ort eine der folgenden Bezeichnungen benutzt werden muß:

|               |                                          |
|---------------|------------------------------------------|
| RO oder LOCAL | für normale Ausgabe im Rechenzentrum,    |
| oder RZ       |                                          |
| GEHEIM        | für spezielle Ausgabe personenbezogener  |
|               | Daten im Rechenzentrum,                  |
| Rn oder RMIn  | für die Stapelstation n (n = 1,2,...10). |
| oder REMOTEn  |                                          |

Darüber hinaus sind für die einzelnen Stapelstationen bestimmte Namen als Synonyme für Rn vereinbart, die ebenfalls benutzt werden können.

Soll eine bestimmte Datei an einem anderen Ort als die normale Druckausgabe des Jobs gedruckt werden, kann man diesen durch die Ausgabe DEST=ort auf der entsprechenden DD-Anweisung oder einer der Datei zugeordneten /\*OUTPUT-Anweisung festlegen (s.u.). Auch im VSPC-Befehl ROUTE kann DEST ort angegeben werden.

### 3) FORMS (Papier, Druckfarbtuch)

Papierart, -format und Vordruck sowie die Art des Druckfarbtuches wird durch einen FORMS-Code vereinbart, der aus 1 bis 4 Buchstaben, Ziffern oder bestimmten Sonderzeichen bestehen kann. Wird für eine Datei kein FORMS-Code oder der Code "STL." angegeben, so wird sie mit einem Standard-Textilfarbtuch auf Normalpapier mit einer Blattgröße von 375 mm Breite und 12 Zoll Höhe gedruckt.

Andere Codes sind entweder auf einer /\*OUTPUT-Anweisung (s.u.) durch die Angabe FORMS=code oder F=code zu benennen oder, falls der Datei keine /\*OUTPUT-Anweisung zugeordnet werden soll, als dritter Unterparameter im SYSOUT-Parameter der entsprechenden DD-Anweisung, der dann die Form SYSOUT=(class,,code) haben muß. Selbstverständlich müssen die benutzten Codes dem Rechenzentrum bekannt sein, falls die Dateien nicht an einer Stapelstation mit eigenen Konventionen gedruckt werden sollen. Das kann entweder durch eine längerfristige Vereinbarung mit dem Rechenzentrum erreicht werden, oder es ist der Code 2000 anzugeben und auf einem Jobbegleitzettel näher zu erläutern. Als allgemein vereinbart gilt der Code 0900 für weißes holzfreies Papier mit der oben genannten Standard-Blattgröße.

Achtung:

Der Einsatz des neuen Druckers vom Typ IBM 3203-5 erlaubt es nicht mehr, daß wie in einigen bisher gültigen Vereinbarungen der FORMS-Code zugleich Angaben über die Drucktypen und die Vorschubkontrolle enthält. Diese sind jetzt unbedingt getrennt als UCS- bzw. FCB-Code zu machen.

4) UCS (Drucktypen, Zeichenvorrat)

Die Drucker im Rechenzentrum sind mit auswechselbaren Druckketten ausgerüstet. Beim Starten eines Druckers durch JES2-Kommandos wird in seinem Zeichen-Puffer (Universal Character Set Buffer) ein genaues Abbild der zu benutzenden Druckkette übertragen. Die Angabe, welche Kette für eine Datei benutzt werden soll, wird mit dem Parameter UCS=Kette auf der entsprechenden DD- bzw. der zugeordneten /\*OUTPUT-Anweisung gemacht.

Im Rechenzentrum stehen zur Zeit zwei Typen von Druckketten mit der Bezeichnung PN und TND zur Verfügung. Die bei fehlender Angabe des UCS-Parameters standardmäßig benutzte PN-Kette trägt vierfach einen Satz von 60 verschiedenen Zeichen (26 Großbuchstaben, 10 Ziffern, 24 Sonderzeichen). Durch diese 60 Zeichen werden auch andere, eigentlich nicht druckbare Bit-Kombinationen dargestellt, indem ersatzweise das druckbare Zeichen benutzt wird, welches in den letzten 6 Bits mit dem vorliegenden Byte übereinstimmt.

Die TND-Kette trägt zweifach einen Satz von 120 verschiedenen Zeichen, darunter Groß- und Kleinbuchstaben, Umlaute, 8 sowie zwei Arten von Ziffern. (Die vorliegende Ausgabe des inform ist mit der TND-Kette gedruckt.)

Die Drucker an den Stapelstationen sind mit Typenstäben bzw. -walzen ausgerüstet, deren Zeichenvorrat der PN-Kette entspricht.

5) FCB (Vorschubsteuerung)

Das Getriebe für den Papiervorschub kann an den beiden seit 1967 bzw. 1970 installierten Schnelldruckern vom Typ IBM 1403-N1 sowie an den Stapelstationen manuell auf Zeilendichten von 6 und 8 (bzw. 10) Zeilen pro Zoll umgeschaltet werden; die Positionierung des Papiers auf Seitenanfang, Seitenende und bestimmte Zwischenpositionen wird bei diesen Geräten durch Abtasten eines mitlaufenden, an den Enden zusammengeklebten Vorschubstreifens gesteuert. Bei dem vor einigen Wochen neu installierten Drucker vom Typ IBM 3203-5 werden Zeilendichte und Positionierung elektronisch durch den "Forms Control Buffer" gesteuert, der wie der Zeichenpuffer beim Starten des Druckers vom JES2 aus einer Bibliothek geladen wird.

Die für eine Datei gewünschte Einstellung des Druckers und Auswahl des Vorschubsteuerstreifens wird in der Form FCB=code auf der DD- bzw. der zugeordneten /\*OUTPUT-Anweisung angegeben. Als code ist dabei eine mit dem Rechenzentrum vereinbarte Bezeichnung aus bis zu 4 Buchstaben oder Ziffern zulässig. Fehlt eine solche Angabe, wird FCB=STD6 angenommen, das bedeutet 6 Zeilen pro Zoll, also 72 Zeilen auf dem Standardpapier mit einer standardmäßigen Seiteneinteilung. Als weiterer allgemein vereinbarter Code gilt STD8 für 8 Zeilen pro Zoll, entsprechend 96 Zeilen auf dem Standardpapier.

#### 6) Die /\*OUTPUT-Anweisung

Wenn für eine Datei mehrere besondere Wünsche angegeben werden oder dieselben Wünsche für mehrere Dateien eines Jobs spezifiziert werden sollen, empfiehlt sich die Benutzung einer /\*OUTPUT-Anweisung. Ihre Beschreibung ist im Benutzerhandbuch zu finden, soll hier aber kurz wiederholt werden.

Den Zeichen /\*OUTPUT in den ersten 8 Spalten folgen ein oder mehrere Leerzeichen, ein OUTPUT-Code aus bis zu 4 Buchstaben oder Ziffern, wiederum ein oder mehrere Leerzeichen und schließlich die Wünsche in der vorher beschriebenen Form, durch Komma getrennt und ohne eingestreute Leerzeichen, höchstens bis Spalte 71. Dabei können als Abkürzungen benutzt werden:

|   |   |       |
|---|---|-------|
| D | - | DEST  |
| F | - | FORMS |
| T | - | UCS   |
| C | - | FCB   |

Sollte eine Zeile nicht reichen, kann unmittelbar eine /\*OUTPUT-Anweisung mit \* als Code und den restlichen Wünschen angehängt werden. Die Zuordnung einer /\*OUTPUT-Anweisung zu einer Datei geschieht dadurch, daß der OUTPUT-Code, wie oben (Abschnitt 3) beschrieben, als dritter Unterparameter im SYSOUT-Parameter der DD-Anweisung für die Datei eingesetzt wird.

#### Achtung:

Vorsicht bei Tippfehlern im OUTPUT-Code! Findet der Code im SYSOUT-Parameter keine Entsprechung auf einer /\*OUTPUT-Anweisung, so wird er als FORMS-Code interpretiert.

Selbstverständlich kann derselbe OUTPUT-Code auf mehreren DD-Anweisungen angegeben werden, umgekehrt sind mehrere /\*OUTPUT-Anweisungen mit unterschiedlichen OUTPUT-Codes in einem Job erlaubt. Angaben auf einer /\*OUTPUT-Anweisung mit dem OUTPUT-Code SYS beziehen sich immer (auch) auf die Systemmeldungen am Anfang der Jobausgabe.

Software-Flohmarkt

E. Pudlatz

An dieser Stelle sollen in unregelmäßigen Abständen Programmprodukte vorgestellt werden, die punktuell bei einzelnen Anwendern im Einsatz sind und von denen wir glauben, daß sie von allgemeinem Interesse sein könnten. Voraussetzung für die Aufnahme derartiger Softwareangebote in diese Rubrik ist die Lauffähigkeit auf der am Rechenzentrum eingesetzten Hardware, die Zuverlässigkeit und eine ausreichende Dokumentation solcher Produkte. Dazu gehört auch die leichte Verfügbarmachung für interessierte Benutzer z.B. über das "sharing" in VSPC.

Durch Veröffentlichung im inform übernimmt das Rechenzentrum weder eine Verantwortung für die genannten Programme, noch eine Verpflichtung, diese allgemein verfügbar zu halten. Ferner sind damit auch keine Aussagen über die Qualität dieser Software verbunden.

Wir beginnen mit drei Programmprodukten, auf deren Vorhandensein Herr Ch. Tarnai vom Institut für Erziehungswissenschaften aufmerksam machte. Interessenten werden gebeten, sich direkt mit ihm in Verbindung zu setzen (Tel. 2504).

PML: A computer program for conditional estimation and testing the Rasch model for dichotomous items.

von Jan-Eric Gustaffson, Universität Göteborg, Januar 1981, Version 3.1 (Programmbeschreibung)

Das Programm erlaubt die Schätzung der Aufgaben- und Personenparameter des dichotomen linearen logistischen Testmodells von Rasch nach der bedingten Maximum-Likelihood-Methode. Neben einer Reihe von Modelltests (z.B. Andersen: Conditional Likelihood Ratio Test) werden Deskriptivstatistiken (z.B. Reliabilität nach Kuder-Richardson) berechnet.

Literatur:

- J.-E. Gustaffson: Testing and obtaining fit of data to the Rasch model  
British Journal of Mathematical and Statistical Psychology 1980  
- : A solution of the conditional estimation problem for long tests in the Rasch model for dichotomous items  
Educational and Psychological Measurement 1980

Implementierungszustand:

Am Institut für Erziehungswissenschaft verfügbar. Bei größerem Bedarf soll es allgemein zugänglich gemacht werden.

LVPLS: Latent variables path analysis with partial least-squares estimation.

von Jan-Bernd Lohmöller, Oktober 1981, Version 1.6. Programmbeschreibung; Forschungsbericht 81-04, Fachbereich Pädagogik, Hochschule der Bundeswehr, München, Werner-Heisenberg-Weg 39, 8014 Neubiberg.

LVPLS besteht aus vier Hauptprogrammen für Pfadanalysen mit latenten Variablen: LVPLSX geht von einer Rohdatenmatrix aus und schätzt ein Modell, das der zugrundeliegenden Methode von Wold (1975, 1981) entspricht. LVPLSC benutzt die Kovarianzmatrix und berechnet ein Modell, das der von Lohmöller (1979) erweiterten Methode entspricht, in der Modelle mit verschiedener Zahl von latenten Variablen pro Block von manifesten Variablen möglich sind. Hier wird auch das komplette Eingabedeck (einschließlich der Startwerte) für LISREL IV von Jöreskog und Sörbom erstellt (s.u.). LVPLS3 ermöglicht die Berechnung einer drei-modalen Pfadanalyse (Lohmöller und Wold 1980). LVPGEN generiert Daten und Kovarianzmatrizen für Pfadmodelle.

Literatur:

Angaben in der o.g. Programmbeschreibung.

Implementierungszustand:

Es liegt z.Zt. die FORTRAN-Quelle vor, die noch den Gegebenheiten des Rechenzentrums angepaßt werden muß.

LISREL IV: Analysis of linear structural relationships by the method of maximum likelihood

von Karl G. Jöreskog und Dag Sörbom, International Educational Services Chicago (Programmbeschreibung und Textbuch)

Das Programm berechnet und testet Modelle für lineare Strukturbeziehungen bei multivariat-normalverteilten Daten. Die Variablen im Strukturgleichungssystem können manifeste oder latente Variablen sein. Das Programm schätzt nach der Maximum-Likelihood-Methode die unbekannten Koeffizienten der Strukturgleichungen und der Residualkovarianzmatrizen. Das Programm LISREL IV ist vergleichbar mit Teilen des Programms LVPLS, das sich der Least-Squares-Methode bedient und daher voraussetzungsärmer hinsichtlich der Datenqualität ist. Beide Programme können aufeinander bezogen werden (s.o. LVPLSC).

Literatur:

Neben der in der Programmbeschreibung genannten Literatur:

D. J. Aigner und A. S. Goldberger (Eds.): Latent variables in socioeconomic models. North Holland Publ., Amsterdam 1976.

A. S. Goldberger und O. D. Duncan (Eds.): Structural equation models in the social sciences, Seminar Press, N.Y. 1973

Implementierungszustand:

Am Institut für Erziehungswissenschaften verfügbar.

Verteilung von Rechenaufträgen nach Jobklassen

H. Schlattmann

In dieser Statistik der Monate Juli bis September 1982 sind für die einzelnen Jobklassen einige Daten aufgelistet. Die mittlere Verweilzeit (in Stunden und Minuten) bezeichnet die mittlere Zeit vom Einlesen eines Rechenauftrages bis zum Ende der Ausführung. Die Verteilung der Häufigkeit der Programme nach Job-CPU-Zeit innerhalb der einzelnen Klassen wurde nach den Angaben der JOBPARM-Anweisung vorgenommen.

Die Tatsache, daß z.B. 20-Sekunden-TEST-Läufe eine niedrigere mittlere Verweilzeit haben als die entsprechenden 10-Sekunden-Läufe, sollte nicht dazu verleiten, Jobs mit einem Zeitbedarf von 20 Sekunden abzugeben, um schneller bedient zu werden. Das bezeichnete Phänomen erklärt sich teilweise dadurch, daß die erstgenannten Jobs häufig dann abgeschickt werden, wenn der Füllungsgrad der entsprechenden Warteschlange niedrig ist, also am frühen Morgen oder abends, um günstige Prioritätszuweisungen wegen "liegendebliebener" 20-Sekunden-Jobs nicht zu gefährden.

| Klasse         | Anzahl | Job-CPU-Zeit  | Verteilung | mittlere Verweilzeit |
|----------------|--------|---------------|------------|----------------------|
| TEST ohne Band | 60458  | 1 - 4 sec     | 60.34%     | 0:21                 |
|                |        | 5 - 10 sec    | 23.38%     | 1:12                 |
|                |        | 11 - 20 sec   | 16.28%     | 0:48                 |
| TEST mit Band  | 7530   | 1 - 4 sec     | 39.13%     | 0:48                 |
|                |        | 5 - 10 sec    | 29.00%     | 2:52                 |
|                |        | 11 - 20 sec   | 31.87%     | 1:53                 |
| PROD ohne Band | 7895   | 1 - 40 sec    | 29.83%     | 2:42                 |
|                |        | 41 - 100 sec  | 29.67%     | 5:12                 |
|                |        | 101 - 500 sec | 19.79%     | 9:07                 |
|                |        | >500 sec      | 20.71%     | 11:38                |
| PROD mit Band  | 4316   | 1 - 40 sec    | 41.14%     | 3:03                 |
|                |        | 41 - 100 sec  | 28.82%     | 7:52                 |
|                |        | 101 - 500 sec | 19.97%     | 9:45                 |
|                |        | >500 sec      | 10.07%     | 9:22                 |
| LONG ohne Band | 51     |               |            | 16:46                |
| LONG mit Band  | 42     |               |            | 14:40                |