

**KERNFORSCHUNGSZENTRUM
KARLSRUHE**

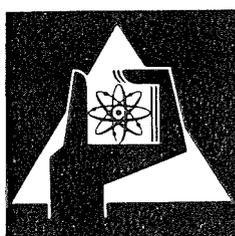
Januar 1973

KFK 1740

Datenverarbeitungszentrale

Ein Personalauskunftssystem in A P L P L U S

G. Wilhelmi



**GESELLSCHAFT
FÜR
KERNFORSCHUNG M.B.H.**

KARLSRUHE

Als Manuskript vervielfältigt

Für diesen Bericht behalten wir uns alle Rechte vor

GESELLSCHAFT FÜR KERNFORSCHUNG M.B.H.
KARLSRUHE

KERNFORSCHUNGSZENTRUM KARLSRUHE

KFK 1740

Datenverarbeitungszentrale

Ein Personalauskunftssystem

in

APL PLUS

G.Wilhelmi

Gesellschaft fuer Kernforschung m.b.H., Karlsruhe



ZUSAMMENFASSUNG :

Es wird ein Personalauskunftssystem beschrieben, welches in dem Timesharing- System *APL PLUS* erstellt wurde. Neben den Zielen der Schnelligkeit, der Benutzerfreundlichkeit und der Flexibilitaet wurde vor allem ein sicherer Datenschutz angesteht. Hierfuer werden einmal die in *APL PLUS* angebotenen Moeglichkeiten des Datenschutzes ausgenutzt. Darueber hinaus wird eine Kodierung wichtiger Personaldaten vorgenommen, welche auf dem "Chinesischen Restsatz" beruht und die in *APL* besonders effektiv verwendet werden kann.

ABSTRACT : A PERSONAL INQUIRY SYSTEM IN *APL*

A personal inquiry system in *APL PLUS* is described. One aim besides quick access of data and flexibility of the system was security of personal data. The security features of *APL PLUS* are fully used; beyond this a coding of important data is performed, which is based on the "chinese remainder theorem" . It is demonstrated that this method can be effectively used in *APL* programs.

INHALTSVERZEICHNIS :

	SEITE
EINLEITUNG	1
1. Uebersicht ueber die Programme	3
2. Beschreibung der Files	7
3. Massnahmen zum Schutz der Daten vor unbefugter Benutzung	10
ANHANG I : Programmlisten	12
ANHANG II : Beispiele	23
Literaturverzeichnis	30

EINLEITUNG :

In dieser Beschreibung wird ein System von in *APL* geschriebenen Programmen fuer schnelle Auskuenfte ueber Personaldaten vorgestellt. Die Erstellung dieser Programme war gedacht als ein exemplarisches Beispiel fuer die Anwendbarkeit des Programmiersystems *APL PLUS* fuer nichtnumerische Aufgabenstellungen. Es sollte ebenso auf die vielfaeltigen Anwendungsmoeglichkeiten von *APL* aufmerksam gemacht werden, um somit dieses hervorragende Hilfsmittel einem weiteren Benutzerkreis naeher zu bringen. Es soll an dieser Stelle darauf hingewiesen werden, dass diese gesamte Beschreibung mit einem *APL*-Texteditor erstellt wurde.

Der gesamte erfasste Personaldatenbestand der Gesellschaft umfasst die Daten von ungefaehr 4000 Mitarbeitern, wobei pro Mitarbeiter 1 Lochkarte (~ 80 bytes) Information verwendet werden.

Ziel der Erstellung dieser Programme war es, ein Personalauskunftssystem zu schaffen, das folgenden 4 Forderungen genuegt :

1. Schnelligkeit
2. Benutzerfreundlichkeit
3. Flexibilitaet
4. Datenschutz

Der genauere Inhalt dieser Schlagworte und deren Verwirklichung in dem Auskunftssystem soll naeher erlaeutert werden.

Zu 1. Schnelligkeit

Es sollte ein von einem Terminal aus in einem Timesharingsystem nutzbares Programmsystem sein. Die Zeit zwischen dem Stellen einer Frage und der Antwort sollte im Sekunden- maximal im Minuten-bereich liegen. Als Implementierungssprache wurde *APL* gewaehlt, das in seiner durch ein Filehandling-System erweiterten Form (*APL PLUS*) als ein geeignetes Instrument erschiene, die oben angefuehrten 4 Forderungen an ein solches Auskunftssystem wirksam zu unterstuetzen. Insbesondere gestattet es *APL PLUS* eine auf eine konkrete Aufgabe ausgerichtete, "massgeschneiderte" Loesungsmethode zu waehlen. Dies bringt im Vergleich zur Verwendung allgemeiner Datenbanksysteme effektivere Programme mit sich, ohne dass man sehr viel mehr an Programmieraufwand benoetigt. Das hier beschriebene System konnte in etwa 4 Wochen von einer Person entworfen, programmiert und getestet werden.

Zu 2. Benutzerfreundlichkeit

Das System sollte benutzerfreundlich sein, derart dass es von jedermann benutzt werden kann ohne Vorkenntnisse in einer Programmiersprache. Das Auskunftssystem besteht aus wenigen einfachen Kommandos ohne Parameter. Die Programme sind voll interaktiv. Die Anfragestruktur ist einfach und schnell erlernbar. Fuer den geuebten, eiligen Benutzer stehen hingegen auch nichtinteraktive Programme mit Parametern und mit kurzen Namen zur Verfuegung. Darueberhinaus wird fehlerhafte Eingabe erkannt und der Benutzer gegebenenfalls zur Korrektur aufgefordert.

Zu 3. Flexibilitaet

Das Auskunftssystem sollte flexibel sein : einmal vom Benutzerstandpunkt aus. Er sollte auf leichte Weise eine grosse Zahl auch komplexer Fragen stellen koennen. Darueberhinaus hat der *APL*-kundige Benutzer die Moeglichkeit, den vollen Umfang von *APL* fuer ausserhalb des implementierten Auskunftssystems liegende Probleme zu nutzen (zum Beispiel : komplexere Statistiken oder Sortierprobleme). Natuerlich koennen solche Erweiterungen auf Wunsch verhaeltnismaessig schnell auch vom Implementierer vorgenommen werden. Desgleichen gestaltet sich die Wartung durch die Kuerze der *APL*-Programme einfach und uebersichtlich.

Zu 4. Datenschutz

Nicht zuletzt sollte ein sicherer Datenschutz fuer vertrauliche Daten (wie z. Bsp. die Gehaltsgruppe) geboten werden. Hier koennte der hohe Grad von Komfort, den *APL PLUS* in dieser Hinsicht liefert genutzt werden. Da das *APL PLUS*-System die Daten jedoch nur gegen andere Terminalbenutzer absichert, musste darueber hinaus auf eine geeignete Kodierung der vertraulichen Daten zurueckgegriffen werden. Die in Teil 3 naeher geschilderte Kodierungsmethode ist besonders auf die Eigenschaften von *APL* ausgerichtet und in dieser Sprache effektiv anwendbar.

In wie weit die hier geschilderten Forderungen erfuellt wurden, moege der Leser den naechsten Teilen dieser Darstellung und vielleicht, sofern er die Erlaubnis hat oder es ihm gelingt die Datensicherung zu durchbrechen, nach eigenen Erfahrungen beurteilen.

1.) Uebersicht ueber die Programme

Alle Funktionen befinden sich in den Buechern 161 *NEUPERS* und 161 *ZEICHPERS* (hier ist nur die Funktion *ZEICHNE*). Es gibt 4 wesentliche Grundfunktionen fuer den Benutzer :

STARTE
FINDE
RECHNE
ZEICHNE

die in dieser Reihenfolge bei einer Anfrage logisch zu durchlaufen sind.

A.) *STARTE* muss nach dem ersten Laden des Buches 161 *NEUPERS* UNBEDINGT als erste Funktion aufgerufen werden. Hierbei wird der Benutzer nach seinem Passwort gefragt, dessen Angabe ihm erst die Benutzung der Datenfiles ermoeglicht. Gleichzeitig damit erfahrt er auch die ihm persoendlich zugaenglichen Attribute, sowie die Einschränkungen, denen seine Anfragen in der Funktion *FINDE* unterliegen. Die Funktion *STARTE* braucht nur einmal waehrend einer Sitzung (bis zum)*OFF*) aufgerufen werden.

Da in *STARTE* einige Ausgangswerte in das Buch *NEUPERS* gebracht werden , ist das Buch *NEUPERS* nach Rueckkehr aus dem Buch *ZEICHPERS* nur dann weiter benutzbar, wenn man zuerst die Funktion : *NEU* aufruft. (man darf aber auch stattdessen *STARTE* aufrufen.)

B.) Die Funktion *FINDE* berechnet auf Grund einer Anfrage einen logischen Vektor und speichert diesen als Ergebnis des Suchprozesses unter einer Nummer (Komponentennummer) in einen jedem Benutzer persoendlich gehoerenden File ab.

Der Benutzer wird zunaechst gebeten, die Komponentennummer anzugeben. Sie muss in einem jedem Benutzer zugewiesenen Rahmen liegen. Bei fehlerhafter Eingabe wird der Benutzer (dies ist allgemein der Fall) auf den Fehler aufmerksam gemacht und um Korrektur gebeten. Danach wird er aufgefordert, seine Anfrage zu formulieren.

Eine Anfrage besteht aus logischen Und- oder Oder-Verbindungen von Vergleichsoperationen mit den Attributen. Als Vergleichsoperatoren sind zugelassen :

= *GLEICH*
≠ *UNGLEICH*
> *GROESSER*
< *KLEINER*
≥ *GROESSER ODER GLEICH*
≤ *KLEINER ODER GLEICH*

Die erlaubten Symbole sind dem Benutzer nach Aufruf der Funktion *STARTE* aufgelistet worden.

Beispiele fuer Vergleichsoperationen mit Attributen :

KINDERZAHL<3
TAETIGALS=10
GEBURTSJAHR=40
ARBEITSZEIT≤2100

Mehrere Vergleichsoperationen koennen nun in einer Anfrage logisch durch Und oder Oder verknuepft werden. Hierzu werden die Symbole

^ *FUER UND*
v *FUER ODER*

verwendet.

Beispiele fuer Anfragen :

TAETIGALS≥10^*TAETIGALS*≤14 (*AKADEMIKER*)
GEBURTSJAHR≤22v*EINTRITTSJAHR*≤62 (*EINE ODER-ANFRAGE*)

Bei Anfragen duerfen ueberall Leerzeichen und Wagenruecklaufzeichen eingefuegt werden. Es koennen beliebig viele Und- und Oder-Verknuepfungen angefuehrt werden. Hierbei gibt es allerdings keine Praeferenz der Und- ueber die Oder-Verknuepfungen. Die Anfrage wird streng von links nach rechts abgearbeitet. Nach Ende der Funktion *FINDE* wird kein Output erzeugt, sondern lediglich ein logischer Vektor unter einer Nummer abgespeichert. Dort bleibt das Ergebnis solange konserviert, bis es durch eine weitere *FINDE*-Funktion oder durch eine *UND*- oder *ODER*-Funktion (siehe unten) mit derselben Nummer ueberspeichert wird.

Attribute, deren Werte nicht bekannt sind, werden mit einem Wert belegt, der groesser ist als der groesste vorkommende reale Wert. Es werden hierfuer die Werte 10 110 1110 11110 usw. je nach der Anzahl der verwendeten Stellen auf dem File Kartel (siehe unten) benutzt. Eine Ausnahme ist das Attribut *KINDERZAHL*. Hierbei stehen auf Kartel Leerzeichen, die in 0 umkodiert wurden.

Die Attribute haben normalerweise Zahlen als Werte mit Ausnahme der beiden Attribute *NAME* und *INSTITUT*, die Zeichenketten als Werte haben. Bei diesen beiden Attributen koennen Anfragen gestellt werden, die nach allen Personen fragen, deren Namen bzw. Institut mit einer gewissen Zeichenfolge beginnt. Also z. Bsp. :

NAME=SCH
NAME=A
NAME=

findet alle Personen, deren Namen mit Sch bzw. mit A anfaengt. Im 3. Beispiel mit leerer rechten Seite werden alle Personen gefunden. Es koennen hierbei maximal 6 Zeichen beruecksichtigt werden. Intern werden auch Zeichenketten als Zahlen gespeichert.

Man kann jedoch nach einem oder nach mehreren Aufrufen der Funktion *FINDE* Auskuenfte verlangen und Modifikationen vornehmen :

I. Auskuenfte
mit den Funktionen *ANZAHL*
LISTE
UNDA

II. Modifikationen
mit den Funktionen *UND*
ODER

I. Auskuenfte :

ANZAHL : gibt die Anzahl derjenigen Faelle, auf die eine bestimmte Anfrage zutrifft.

LISTE : listet eine Reihe von Attributen von allen Faellen, auf die die Anfrage zutrifft. Die Auswahl der Attribute kann der Benutzer vornehmen.

UNDA : liefert in einer Tabelle mit 2 Eingaengen (die aus Anfragenummern bestehen) die Anzahl derjenigen Faelle, die sowohl bei der ersten, als auch bei der zweiten Anfrage zutreffen. (Vorsicht : bitte grosse Tabellen vermeiden wegen grosser Rechenzeiten --- besser die Moeglichkeiten der Funktion *RECHNE* (siehe unten) ausnutzen).

II. Modifikationen :

UND

ODER : diese Funktionen liefern die Moeglichkeit beliebig viele schon beantwortete Fragen mit Und oder Oder zu verknuepfen und das Ergebnis unter einer neuen (oder einer alten) Nummer abzuspeichern.

C.) Die Funktion *RECHNE* baut auf die Funktion *FINDE* auf (ein *FINDE* muss vorausgegangen sein) und berechnet einen Vektor der aus den Zahlengroessen eines Attributs besteht, das der Benutzer spezifiziert, und bezieht sich auf eine Anfragenummer. Das Ergebnis wird ebenfalls in einem benutzereigenen File abgespeichert. (Jeder Aufruf der Funktion *RECHNE* zerstoert das Ergebnis vorhergehender Aufrufe!)

Hat man zum Beispiel in der Funktion *FINDE* unter der Nummer 2 die Anfrage *PERSONALNUMMER* ≥ 7000 (Arbeiter) gespeichert, so kann man nun in *RECHNE* von dieser Nummer 2 durch Spezifikation des Attributs *KINDERZAHL* die Anzahl der Kinder jedes Arbeiters in einem Vektor abspeichern lassen. Die Funktion *RECHNE* produziert ebenfalls keinen Output.

Jedoch gibt es hier 5 Funktionen, um im Anschluss an *RECHNE* Auswertungen des Ergebnisses vorzunehmen :

SUMME
MITTELWERT
MAXIMUM
MINIMUM
TABELLE

Diese Funktionen liefern die aus ihrem Namen hervorgehenden Dinge. Hierbei liefert die Funktion *TABELLE* eine Tabelle, in deren 1. Spalte alle ueberhaupt vorkommenden Groessen in aufsteigender Folge geordnet erscheinen. In der 2. Spalte steht die entsprechende Haeufigkeit. Es besteht ebenfalls die Moeglichkeit, die Menge der zu tabellierenden Groessen noch in der Funktion *TABELLE* selbst einzuschaeraken. Das Ergebnis der Funktion *TABELLE* wird ebenfalls intern abgespeichert und steht fuer die Funktion *ZEICHNE* zur Verfuegung.

D.) Die Funktion *ZEICHNE* steht im Buch 161 *ZEICHPERS*. Nach Laden dieses Buches und Aufruf der Funktion *ZEICHNE* erhaelt man ein Histogramm, der vorher in *TABELLE* berechneten Werte.

Bemerkung : Alle obigen Funktionen haben keine Parameter und fordern ihre Eingabe selbst an. Zu vielen Programmen gibt es eine Zweitversion, deren Namen aus dem ersten Anfangsbuchstaben der parameterlosen Funktion besteht und die statt Eingabe anzufordern Parameter haben :

Übersicht

	<i>S</i>		fuer <i>STARTF</i>
<i>NUMMER</i>	<i>F</i>	'ANFRAGE'	fuer <i>FINDE</i>
	<i>A</i>	<i>NUMMER</i>	fuer <i>ANZAHL</i>
	<i>L</i>	<i>NUMMER</i>	fuer <i>LISTF</i>
<i>ZIELNUMMER</i>	<i>U</i>	<i>NUMMERN</i>	fuer <i>UND</i>
<i>ZIELNUMMER</i>	<i>O</i>	<i>NUMMERN</i>	fuer <i>ODER</i>
<i>NUMMERN</i>	<i>UA</i>	<i>NUMMERN</i>	fuer <i>UNDA</i>
<i>NUMMER</i>	<i>R</i>	'ATTRIBUT'	fuer <i>RECHNE</i>

2.) Beschreibung der Files.

Bei der Benutzung der Personalauskunftsprogramme arbeitet der Benutzer mit 5 Datenfiles : mit 3 allgemeinen und 2 benutzerspezifischen :

A. Die allgemeinen Files

9161 KARTE1
9161 FILEPM
9161 KONT

Diese Files sind mit einem Schluessel geschuetzt. Jeder Benutzer bekommt nur die Erlaubnis aus diesen Files zu lesen. Da die Lesebefehle in nichtauflistbaren mit einem "Lock" versehenen Funktionen auftreten und in ihnen der Schluessel explizit angegeben wird, kann der Benutzer die Daten nur durch die vorgegebenen Funktionen (zum Beispiel durch *FINDE*, *LISTE* usw.) benutzen. Der Schluessel selbst bleibt ihm unbekannt.

I. 9161 KARTE1

enthaelt die Personaldaten, so wie sie ein PI/1-Programm von den urspruenglichen Karten umgewandelt und verschluesselt hat. Jede Komponente dieses Files besteht aus einer 100x90-Matrix. Jede Zeile beschreibt eine Person. Die Spalten sind fest bestimmten Attributen zugeordnet. Diese Zuordnung wird genauer in Komponente 3 des Files *KONT* beschrieben. Wichtige Personaldaten, wie zum Beispiel die Gehaltsgruppe sind verschluesselt (wie in Teil 3 beschrieben). Die Entschluesselungskonstanten befinden sich ebenfalls in dem File *KONT*. Der File 9161 *KARTE1* wird nur in der Funktion *LISTE* verwendet. (Er wird natuerlich als Zwischenglied benoetigt um den File 9161 *FILEPM* herzustellen.)

II. 9161 FILEPM

enthaelt dieselben Daten wie 9161 *KARTE1* in aufbereiteter Form. Dieser File wird (ausser bei der Funktion *LISTE*) bei den Auskunftsprogrammen benutzt. Er besteht aus einer Reihe von Komponenten, wobei jedes Attribut in genau einer Komponente des Files enthalten ist. Mehrere Attribute koennen dabei in einer Komponente verschluesselt entahlten sein. (Siehe Teil 3. Verschlusselung der Daten.) Jede Komponente enthaelt genau soviele Integer-Zahlen wie sich Personen im Auskunftssystem befinden. Die Reihenfolge der Zahlen in allen Komponenten entspricht der Reihenfolge der Personalnummern. Alle Attribute, die Zeichenketten sind, wie Name oder Institut sind in Zahlen verschluesselt. Hierbei werden maximal 6 Zeichen verwendet. Zunaechst wird jedem Zeichen der Index dieses Zeichens im Alphabet (A,B,...,Z,Blank) zugeordnet. Also dem Zeichen A die 1, B die 2,, Z die 26 und Blank die 27. Sodann wird dieser Vektor gespiegelt und im 30-Zahlensystem verschluesselt. Die Entschluesselung und die Abfrage auf irgendwelche Anfangskombinationen (wie z. Bsp. : *NAME=SCH*) geschieht dann durch Anwendung der Modulofunktion (|).

III. 9161 KONT

Dieser File kontrolliert die einzelnen Benutzer, beschreibt das Format von *KARTE1* und legt die Attributnamen fest. Er hat 5 Komponenten :

1.Komponente : Eine Zeichenmatrix, in deren Zeilen die Benutzernummern stehen, die Erlaubnis haben mit dem Auskunftssystem zu arbeiten.

2.Komponente : Eine Bitmatrix, mit sovielen Zeilen wie Benutzer des Systems existieren und soviel Spalten wie Attribute existieren. Sie dient als Maske und kann fuer jeden Benutzer jedes einzelne Attribut sperren (0) oder zugaenglich machen (1).

3.Komponente : Tabelle zur Beschreibung des Formats von *KARTE1*. Sie enthaelt 4 Spalten und soviel Zeilen wie Attribute existieren. Die 1. Spalte enthaelt die Attributnummer, die 2. die Entschluesselungskonstante dieses Attributs (siehe Teil 3) bzw. die Zahl 0, wenn es sich um kodierte Zeichenketten handelt. Die 3. Spalte enthaelt die Nummer der Spalte von *KARTE1*, in der dieses Attribut anfaengt und die 4. Spalte die Anzahl der Spalten des Attributs auf *KARTE1*.

4.Komponente : Sie enthaelt in einer 15-spaltigen Zeichenmatrix die Namen aller erlaubten Attribute. Die Reihenfolge muss mit der in Komponente 3 uebereinstimmen.

5.Komponente : Sie enthaelt in einer 15-spaltigen Zeichenmatrix eine benutzerspezifische Einschraenkung der Anfragemoeglichkeiten bzw. Blanks, wenn keine solche Einschraenkung besteht. Enthaelt zum Beispiel die 2. Zeile dieser Matrix die Zeichenkette : *^INSTITUT=DVZ*, so werden alle Anfragen des 2. Benutzers laut Zeile 2 von Komponente 1 des Files 9161 *KONT* mit dieser Einschraenkung belegt.

Bei Aufruf der Funktion *STARTE* erfahrt der Benutzer welche Attribute ihm zur Verfuegung stehen und welchen Einschraenkungen seine Anfragen unterliegen.

B. Die benutzerspezifischen Files

HILFPN
HPLOT

Diese Files hat jeder Benutzer persoendlich. Er hat die Erlaubnis damit folgende *APL PLUS*-Operationen durchzufuehren :

ERFAD
FTIE
FREPLACE
FRFNAME

Diese Files sind nicht durch einen Schluessel geschuetzt. Der Benutzer 9161 hat alle Erlaubnis, diese Files zu benutzen. Sie werden beim Erstellen eines neuen Files 9161 *KARTE1* und 9161 *FILEPM* neu initialisiert.

I. *HILFPN*

ist ein File zur Aufnahme von Anfrageergebnissen des Benutzers, die mit der Funktion *FINDE* erstellt wurden. Der File *HILFPN* hat eine Komponente mehr als die Zahl in der 1. Komponente des Files *HPLOT* angibt. Jede Komponente ist ein Bitvektor, dessen Groesse durch die Anzahl der erfassten Personen bestimmt ist. Die letzte Komponente wird als Zwischenspeicher in der Funktion *UA* verwendet.

II. *HPLOT*

enthaelt einige benutzerspezifische Kontrollgroessen und ist Speicherplatz fuer Ergebnisse der Funktionen *RECHNE* und *TABELLE*.

1. Komponente : *ANZANF* (Anzahl der verschiedenen Anfragen, die in *HILFPN* gespeichert werden koennen)

2. Komponente : *ANZPN* (Anzahl der Personen im Auskunftssystem)

3. Komponente : *ANZMA* (Anzahl der Komponenten von File 9161 *KARTE1*)

4. Komponente : Ergebnis der Funktion *RECHNE*

5. Komponente : Rechtes Argument der Funktion *R* (wird in *TABELLE* und *ZEICHNE* benutzt)

6. Komponente : Passwort des Benutzers (verschluesstelt)

7. Komponente : Stand der Daten (Datum)

8. Komponente : 0, falls in *TABELLE* von Integer in Zeichenform zurueckkodierte werden muss

9. Komponente : Ergebnis der Funktion *TABELLE*

3.) Massnahmen zum Schutz der Personaldaten vor unbefugter Benutzung

Es ist im allgemeinen ein schwieriges Problem, maschinell gespeicherte Daten vor dem Zugriff Unbefugter zu schuetzen, insbesondere wenn einer grossen Zahl von Benutzern der Rechenanlagen Zugriff auf die Datentraeger nicht verwehrt werden kann. Obwohl *APL PLUS* als geschlossenes System in sich eine grosse Zahl von Schutzmoeglichkeiten hat, die eine unbefugte Benutzung von Daten vom *APL-TERMINAL* aus ausschliesst, sind die Daten dem Zugriff vom Batchbetrieb her verhaeltnismaessig schutzlos preisgegeben.

Um hier Abhilfe zu schaffen werden schutzwuerdige Daten, wie etwa die Gehaltsgruppe, nur in einer verschluesselten Form gespeichert. Zur Verschluesselung wird hierbei eine Methode verwendet, die genuegend einfach in *APL* zu dekodieren ist, und von der wir glauben, dass sie hinreichende Sicherheit bietet. Diese Verschluesselung dient primaer dem Schutz gegen Zugriffe aus dem Batchbetrieb. In *APL* selbst wird von den Moeglichkeiten von *APL PLUS* gebraucht gemacht, um ein flexibles und effektives System zu schaffen, das es erlaubt jedem Benutzer die Verwendung einzelner Attribute (z. Bsp. *GEHALTSGRUPPE*) zu verbieten oder seine Anfrage auf gewisse Teilbereiche einzuschraenken (z. Bsp. nur *INSTITUT=DVZ*). Hierzu dienen vor allem folgende beiden Massnahmen :

1. Die Funktionen koennen nicht aufgelistet werden
2. Die Datenfiles sind durch einen Schluessel gesichert, der nur in den nichtlistbaren Funktionen vorkommt, also dem Benutzer nicht zugaenglich ist.

Die Hauptsicherheit liegt jedoch in der Verschluesselung der Daten. Dies geschieht mit Hilfe der Anwendung des Chinesischen Restsatzes (Literaturhinweis : Hardy,Wright : Zahlentheorie Muenchen 1958 S. 106-108) und soll im folgenden erlaeutert werden.

Angenommen wir haben k Attribute $a_1, a_2, a_3, \dots, a_k$, die verschluesselt werden sollen. Dabei sei bekannt, dass diese Groessen positive ganze Zahlen sind, die jeweils in dem Zahlenintervall $0 \leq a_i \leq m_i$ liegen ($i=1,2,\dots,k$), wobei die Zahlen m_1, m_2, \dots, m_k teilerfremd gewaehlt sein sollen. Bezeichnen wir mit M das Produkt $m_1 \times m_2 \times \dots \times m_k$, so besagt der Chinesische Restsatz, dass es im Bereich zwischen 0 und M eine eindeutige Zahl X gibt mit den Eigenschaften

$$\begin{aligned} a_1 &= m_1 | X \\ a_2 &= m_2 | X \\ &\vdots \\ &\vdots \\ &\vdots \\ a_k &= m_k | X \end{aligned}$$

Diese Zahl X berechnet sich nach folgender Methode : zu jedem m_i gibt es eine Zahl p_i , so dass $m_i \times p_i = M$ gilt. Da m_i und p_i teilerfremd sind laesst sich ferner ein (modulo m_i eindeutig) bestimmtes n_i finden, fuer das $1 = m_i p_i n_i$ gilt.

Berechnet man diese Groessen fuer alle $i = 1, 2, \dots, k$, so gilt

$$X = a_1 \times n_1 \times p_1 + a_2 \times n_2 \times p_2 + \dots + a_k \times n_k \times p_k$$

Dies sei an einem Beispiel erlaeutert :

Es sei $a_1 = 2$ $a_2 = 4$ $a_3 = 5$

und $0 \leq a_1 \leq 7 = m_1$ $0 \leq a_2 \leq 5 = m_2$ $0 \leq a_3 \leq 6 = m_3$

so gilt

$p_1 = 30$ $p_2 = 42$ $p_3 = 35$

und

$M = 210$

Ferner ergibt sich :

$n_1 = 4$ denn $1 = 7 | 30 \times 4$

$n_2 = 3$ denn $1 = 5 | 42 \times 3$

$n_3 = 5$ denn $1 = 6 | 35 \times 5$

Damit ergibt sich $X = 2 \times 4 \times 30 + 4 \times 3 \times 42 + 5 \times 5 \times 35 = 1619$

und es folgt

$a_1 = 2 = 7 | 1619$

$a_2 = 4 = 5 | 1619$

$a_3 = 5 = 6 | 1619$

Statt der k Zahlen a_1, a_2, \dots, a_k wird jetzt nur die eine Zahl X gespeichert und zur Entschuesselung wird diese modulo der entsprechenden Entschuesselungskonstanten genommen. Diese Operation ist in *APL* schnell und leicht durchzufuehren.

Zur weiteren Sicherung wird zu der Zahl X eine weitere Zahl, nennen wir sie *CONST* addiert, so dass bei der Entschuesselung a_i folgendermassen gewonnen wird :

$$a_i = m_i | X - CONST$$

Die Konstanten m_i befinden sich in der 2. Spalte der Matrix, die sich in der 3. Komponente des Files 9161 *KONT* befindet, waehrend sich die Konstante *CONST* nur in der verschluesselten Funktion *FRAGE* in Zahlenform befindet (*FRAGE* wird von *F* aufgerufen).

Zusaetzlich zu dem Datenschutz kann durch eine solche Verschluesselung ein Gewinn an Speicherplatz verzeichnet werden. Allerdings muss bekannt sein, dass sich die Werte der Attribute in einem verhaeltnismaessig schmalen Bereich befinden. Die Zahl X darf natuerlich nicht groesser als 2×31 werden, damit sie noch als Integer-Groesse in *APL* behandelt wird.

ANHANG I

PROGRAMMLISTEN

Bemerkung :

Die Programme sind identisch mit denen in 161 neupers und 161 zeichpers bis auf die Tatsache, dass statt der entsprechenden Konstanten hier Variable *:CONST* und *KEY* eingefuegt sind.

)LOAD NEUPERS
SAVED 14.38.58 11/07/72

)WIDTH 70
WAS 100

)GRPS

)FNS

A	ANZAHL	CMATONS	CPU	F	FINDF	FR	FRAGE	FE
L	LISTE	MAXIMUM	MINIMUM	MITTELWERT		NFU	O	ODER
PR	R	RECHNE	S	STARTE	SUMME	SYNTAX	TABELLE	TFV
U	UA	UND	UNDA	ΔFMT				

∇A[□]∇
∇ V←A N
[1] V←+/FE 6,1,N
∇

∇ANZAHL[□]∇
∇ ANZAHL;N;Z
[1] L1:'GEBEN SIE DIE NUMMER DER ANFRAGE AN :'
[2] N←□
[3] Z←N PR 0 0
[4] →(Z≠1)/L1
[5] A N
∇

∇CMATONS[□]∇
∇ NS←CMATONS CMA
[1] NS←101□¹+ '0123456789' 1CMA
∇

∇F[□]∇
∇ N F ST;I;MAX;CO;ZW
[1] SYNTAX ST,BL.,(FE 6,5,4,KEY)[ZI;]
[2] ZW←ANZPN₀0
[3] I←1
[4] MAX←(PTA)[1]
[5] CO←1,(MAX-1)+TA[;4]
[6] L4:→(L1,L2)[CO[I]+1]
[7] L1:ZW←ZW^FRAGE I
[8] →L3
[9] L2:ZW←ZW^FRAGE I
[10] L3:→(MAX≥I+I+1)/L4
[11] ZW FE 8,1,N
∇

```

      ▽FINDER[ ]▽
      ▽ FINDER;N;ST;Z
[1] L1:'UNTER WELCHER NUMMER SOLL DIE ANFRAGE ABGESPEICHERT WERDEN?'
[2] N←□
[3] Z←N PR 0 0
[4] →(Z≠1)/L1
[5] 'FORMULIEREN SIE DIE ANFRAGE'
[6] ST←□
[7] N F ST
      ▽

```

```

      ▽FR[ ]▽
      ▽ V←FR X;ZW;I
[1] →(TEV X)/L1
[2] ZW←(ZW>0,(-1+ρZW)ρZW)/ZW←(I←ΦV,X)ιΔ,X
[3] V←(,X)[I[ZW]],[1.5] ZW-0,(-1+ρZW)ρZW
[4] →0
[5] L1:V← 0 2 ρ0
      ▽

```

```

      ▽FRAGF[ ]▽
      ▽ V←FRAGF I;VZ;J;VL
[1] VZ←TA[I;2]
[2] →((TA[I;3]<0)^1≠TA[I;2])/F1
[3] →(TA[I;3]≤0)/L1
[4] VL←CMATONS(VT[I;]≠BL)/VT[I;]
[5] V←ι0
[6] L8:→(L2,L3,L4,L5,L6,L7)[VZ]
[7] L2:V←VL=TA[I;3]|(CONST×18=TA[I;1])+FE 6,2,TA[I;1],KEY
[8] →L9
[9] L3:V←VL<TA[I;3]|(CONST×18=TA[I;1])+FE 6,2,TA[I;1],KEY
[10] →L9
[11] L4:V←VL>TA[I;3]|(CONST×18=TA[I;1])+FE 6,2,TA[I;1],KEY
[12] →L9
[13] L5:V←VL≠TA[I;3]|(CONST×18=TA[I;1])+FE 6,2,TA[I;1],KEY
[14] →L9
[15] L6:V←VL≥TA[I;3]|(CONST×18=TA[I;1])+FE 6,2,TA[I;1],KEY
[16] →L9
[17] L7:V←VL≤TA[I;3]|(CONST×18=TA[I;1])+FE 6,2,TA[I;1],KEY
[18] L9:→0
[19] E1:'UNGUELTIGER VERGLEICH IN KOMPONENTE NR. : ' ; I
[20] →
[21] L1:VL←Φ(VZ+6[|TA[I;3])ρVT[I;]
[22] VL←((26+LC),BL)ιVL
[23] VL←30ιVL
[24] V←VL=(×/VZρ30)|FE 6,2,TA[I;1],KEY
      ▽

```

```

    ∇L[□]∇
    ∇ NR L LI;V;N;R;F;SCHAB;MA;MN;LL;Q;I;ZW;P1
[1]  LI←(LI≠BL)/LI
[2]  LI←(LI≠CR)/LI
[3]  V←10
[4]  L1:N←LI1', '
[5]  ZW←(N-1)+LI
[6]  LI←N+LI
[7]  R←ALΛ.=ZW,(15-ρZW)ρBL
[8]  →(~1∈R)/F1
[9]  V←V,+/R×1ρR
[10] →(0<ρLI)/L1
[11] SCHAB←((8×-1+ρV)ρ' A1,X1,')', ' A1,X1'
[12] F←'ZI2' ΔFMT SP[V;4]
[13] N←1+8×-1+1ρV
[14] SCHAB[(N,N+1)[AN,N+1]]←,F
[15] Q←10
[16] I←1
[17] L7:Q←Q,SP[V[I];3]+-1+1SP[V[I];4]
[18] →((ρV)≥I←I+1)/L7
[19] P1←(-1+1FE 6 4 3)×100
[20] MA←0
[21] N←FE 6,1,NR
[22] L3:→((1+ANZPN)=ZW←N11)/0
[23] MN←+/ZW>P1
[24] →(MN=MA)/L2
[25] LL←FE 6,3,MN,KEY
[26] L2: SCHAB ΔFMT(1,ρQ)ρLL[(100|ZW)+((0=100|ZW)×100);Q]
[27] N[ZW]←0
[28] MA←MN
[29] →L3
[30] E1:ZW;' NICHT IN ATTRIBUTLISTE'
    ∇

```

```

    ∇LISTE[□]∇
    ∇ LISTE;Z;NR;LI
[1]  L1:'GEBEN SIE DIE NUMMER AN,UNTER DER DIE ANFRAGE ABGESPEICHERT
    WURDE : '
[2]  NR←□
[3]  Z←NR PR 0 0
[4]  →(Z≠1)/L1
[5]  'GEBEN SIE DURCH KOMMAS GETRENNT DIE ATTRIBUTE AN,DIE
    GELISTET WERDEN SOLLEN : '
[6]  LI←□
[7]  NR L LI
    ∇

```

```

    ∇MAXIMUM[□]∇
    ∇ Z←MAXIMUM
[1]  Z←[/FE 6 4 4
    ∇

```

```

    ∇MINIMUM[□]∇
    ∇ Z←MINIMUM
[1]  Z←[/FE 6 4 4
    ∇

```

```

      ▽MITTELWERT[□]▽
▽ Z←MITTELWERT;A
[1] A←FF 6 4 4
[2] →(0=ρ,A)/E1
[3] Z←(+/,A)÷ρ,A
[4] →0
[5] E1:'LEERE MENGE , MITTELWERT WURDE AUF 0 GESETZT'
[6] Z←0
▽
```

```

      ▽NEU[□]▽
▽ NEU;I
[1] ANZANF←FF 6 4 1
[2] ANZMA←FF 6 4 3
[3] ANZPN←FF 6 4 2
[4] I←FF 6 4 7
[5] ZI←+/(1ρI)×I←((FF 6,5,1,KEY)Λ.=,'I4' ΔFMTI29)
[6] SP←(FF 6,5,2,KEY)[ZI;]≠FF 6,5,3,KEY
[7] AL←(FF 6,5,2,KEY)[ZI;]≠FF 6,5,5,KEY
▽
```

```

      ▽O[□]▽
▽ N O V;I;ZW1;ZW2;MAX
[1] I←1
[2] MAX←ρ,V
[3] ZW1←ANZPNρ0
[4] L1:ZW2←FF 6,1,(,V)[I]
[5] ZW1←ZW1∨ZW2
[6] →(MAX≥I←I+1)/L1
[7] ZW1 FF 8,1,N
▽
```

```

      ▽ODER[□]▽
▽ ODER;N;V;Z
[1] L1:'GEBEN SIE DIE NUMMERN DER ANFRAGEN AN,DIE
      DURCH ODER VERKNUEPFT WERDEN SOLLEN .'
[2] V←,□
[3] Z←V PR 1,ANZANF
[4] →(Z≠1)/L1
[5] L2:'GEBEN SIE DIE NUMMER AN,UNTER DER DIE ODER-VERKNUEPFUNG
      ABGESPEICHERT WERDEN SOLL : '
[6] N←□
[7] Z←N PR 0 0
[8] →(Z≠1)/L2
[9] N O V
▽
```

```

    ∇ PR[ ] ∇
    ∇ Z←A PR V;ZW
[1] ZW←'WIEDERHOLEN SIE DIE EINGABE'
[2] →(V[1]=ρρA)/L1
[3] 'DIE EINGABE MUSS ';V[1];' DIMENSION(FN) HABEN'
[4] ZW
[5] Z←0
[6] →0
[7] L1:→(V[1]=0)/L2
[8] →(1=^(ρA)≤V[2])/L2
[9] 'DIMENSIONSGROESSE ';V[2];' UEBERSCHRITTEN'
[10] ZW
[11] Z←0
[12] →0
[13] L2:→(A∈\ANZANF)/L3
[14] 'ES MUSS EINE GANZE ZAHL ZWISCHEN 1 UND ';ANZANF;' SEIN'
[15] ZW
[16] Z←0
[17] →0
[18] L3:Z←1
    ∇

```

```

    ∇ R[ ] ∇
    ∇ N R A;R;I
[1] R←ALΛ.=A,(15-ρA)ρBL
[2] →(~1∈R)/F1
[3] I←+/R×10R
[4] A FE 8 4 5
[5] (SP[I;2]) FE 8 4 8
[6] ((FE 6,1,N)/SP[I;2]|(CONST×18=SP[I;1])+FE 6,2,SP[I;1],KEY) FE
    8 4 4
[7] →0
[8] F1:A;' NICHT IN ATTRIBUTLISTE'
    ∇

```

```

    ∇ RECHNE[ ] ∇
    ∇ RECHNE;N;A;Z
[1] L1:'GEBEN SIE DIE NUMMER DER ANFRAGE AN, MIT DER SIE RECHNEN WOLL
    EN : '
[2] N←[ ]
[3] Z←N PR 0 0
[4] →(Z≠1)/L1
[5] 'GEBEN SIE EIN ATTRIBUT AN, DAS BERECHNET WERDEN SOLL:'
[6] A←[ ]
[7] N R A
    ∇

```

∇S[]∇

∇ S;I

- [1] FE 3, FE 18
- [2] 'HPLOT' FE 4 4
- [3] 'PASSWORT' : '
- [4] M←STR,BS,(ΦSTR),BS,(1ΦSTR),BS,(2ΦSTR),BS,(3ΦSTR),BS
- [5] →(1≠^/((-ρI)↑M)=LC[I←FE 6 4 6])/E1
- [6] 'HILEPN' FE 4 1
- [7] '9161 FILEPM' FE 4,2,KEY
- [8] '9161 KARTE1' FE 4,3,KEY
- [9] ANZAN←FE 6,4,1
- [10] ANZMA←FE 6 4 3
- [11] ANZPN←FE 6 4 2
- [12] LF
- [13] I←FE 6 4 7
- [14] 'STAND DER DATEN VOM ';I[2];'. ';YEAR[I[1];];' 19';I[3]
- [15] '9161 KONT' FE 4,5,KEY
- [16] ZI←+/(1ρI)×I←((FE 6,5,1,KEY)∧.=,'I4' ΔFMTI29)
- [17] SP←(FE 6,5,2,KEY)[ZI;]←FE 6,5,3,KEY
- [18] AL←(FE 6,5,2,KEY)[ZI;]←FE 6,5,5,KEY
- [19] 'SIE KÖNNEN FOLGENDE ATTRIBUT VERWENDEN :'
- [20] LF
- [21] AL
- [22] →(1=∧/BL=(FE 6,5,4,KEY)[ZI;])/L1
- [23] ES←(FE 6,5,4,KEY)[ZI;]
- [24] 'ALLE IHRE ANFRAGEN UNTERLIEGEN DER NEBENBEDINGUNG :
' ;ES
- [25] L1:→0
- [26] E1:'FALSCHES PASSWORT'
- [27] →3

∇

∇STARTE[]∇

∇ STARTE

- [1] S

∇

∇SUMME[]∇

∇ Z←SUMME

- [1] Z←+ / FE 6 4 4

∇

```
∇SYNTAX[□]∇
∇ SYNTAX ST;AN;I;IM;IN;R;V;WO;ZW;AM
[1] ST←((,ST)≠BL)/,ST
[2] ST←(ST≠CR)/ST
[3] TA← 0 4 ρ0
[4] V←4ρ0
[5] VT← 0 15 ρ0
[6] →(3>ρST)/F1
[7] →(1≠∧/ST∈LC)/F2
[8] ZW←ST
[9] L2:IN←(ZW∧'∧')[ZW∧'∨'
[10] AN←(IN-1)↑ZW
[11] ZW←(IN-1)↑ZW
[12] IM←(AM∨/AN∅.=S1)∧1
[13] →(1≠+/AM)/E3
[14] V[2]←S1∧AN[IM]
[15] WO←(IM-1)↑AN
[16] L:→((ρWO)>15)/E5
[17] R←AL∧.=WO,(15-ρWO)ρBL
[18] →(~1∈R)/E4
[19] I←+/R×∧ρR
[20] V[1 3]←SP[I; 1 2]
[21] V[3]←((V[3]=0)×-ρ(1+ρWO)↑AN)+(V[3]≠0)×V[3]
[22] VT←VT,[1] IM↑AN,(15-ρIM↑AN)ρBL
[23] TA←TA,[1] V
[24] →((ρZW)=0)/0
[25] TA[(ρTA)[1];4]←ZW[1]='∨'
[26] ZW←1↑ZW
[27] →L2
[28] F1:'UNKORREKTE ANFRAGE'
[29] →
[30] F2:'FALSCHES ZFICHEN'
[31] ST
[32] '∧'[(~ST∈LC)+1]
[33] →
[34] F3:'KEINE ODER MEHRERE VERGLEICHSOPERATOREN IN FRAGE : ','AN
[35] →
[36] F4:WO;' NICHT IN ATTRIBUTLISTE'
[37] L3:'GEBEN SIE KORREKTES ATRIBUT AN'
[38] WO←□
[39] →L
[40] F5:'ATTRIBUT ZU LANG'
[41] →L3
∇
```

```
∇TABELLE[ ]∇
∇ TABELLE;ZW;V;FL;Z;J;V1
[1] FL←0
[2] V← 0 2 ρ0
[3] ZW←FE 6 4 4
[4] 'SOLLEN ALLE GROSSEN TABELLIERT WERDEN ?'
[5] →(1=Λ/'JA'=2+□)/L7
[6] 'BITTE GEBEN SIE DIE GROSSEN AN,DIE IN DIE
TABELLE AUFGENOMMEN WERDEN : '
[7] ZW←(ZWε□)/ZW
[8] L7:3ρLF
[9] →(300<ρZW)/L2
[10] L5:→(300<ρZW)/L4
[11] V←V,[1] FR ZW
[12] →(FL=0)/L6
[13] J←AV[;1]
[14] V[;1]←(V[;1])[J]
[15] V[;2]←(V[;2])[J]
[16] L6:V FE 8 4 9
[17] (FE 6 4 5); ' ANZAHL '
[18] →(0≠FE 6 4 8)/L1
[19] ZW←Q(6ρ30)TV[;1]
[20] ZW←ZW×ZWε127
[21] ZW←ZW+27×ZW=0
[22] ZW←ZW+27×ZW=0
[23] ZW←φ((26+LC),BL)[ZW]
[24] V←'6A1,X3,I4' ΔFMT(ZW;V[;2])
[25] V
[26] →0
[27] L1:V
[28] →0
[29] L2:Z←ZW[40?ρZW]
[30] FL←1
[31] Z←FR Z
[32] →(16<(ρZ)[1])/L3
[33] L4:Z←ZW[40?ρZW]
[34] Z←FR Z
[35] J←Z[Z[;2]1[Z[;2];1]
[36] V1←ZWεJ
[37] ZW←(~V1)/ZW
[38] V←V,[1] 1 2 ρJ,ρV1/V1
[39] →L5
[40] L3:'DIE RECHENZFIT WIRD GROSS,DA EINE GROSSE MENGE
VERSCHIEDENER ZAHLFN ZU TABELLIEREN IST. MOECHTEN SIE
TROTZDEM DIE TABELLE ?'
[41] →(1=Λ/'JA'=2+□)/L4
∇
```

```
∇TEV[ ]∇
∇ Z←TEV A
[1] Z←((1=ρρA)Λ0=+/ρA)
∇
```

```

      VU[ ]V
  ▽ N U V;I;ZW1;ZW2;MAX
[1]  I←1
[2]  MAX←ρ,V
[3]  ZW1←ANZPNρ1
[4]  L1:ZW2+FF 6,1,(,V)[I]
[5]  ZW1←ZW1^ZW2
[6]  →(MAX≥I←I+1)/L1
[7]  ZW1 FF 8,1,N
  ▽

```

```

      VUA[ ]V
  ▽ V1 UA V2;I;J;MI;MJ
[1]  I←1
[2]  MI←ρV1
[3]  MJ←ρV2
[4]  MA←ι0
[5]  L2:J←1
[6]  L1:(ANZANF+1) U V1[I],V2[J]
[7]  MA←MA,A ANZANF+1
[8]  →(MJ≥J←J+1)/L1
[9]  →(MI≥I←I+1)/L2
[10] MA←(MI,MJ)ρMA
[11] ' |BI4' ΔFMT(1,1+ρV2)ρ0,V2
[12] 'A1' ΔFMT(1,6×1+ρV2)ρ'-'
[13] ' |BI4' ΔFMT(V1;MA)
  ▽

```

```

      VUND[ ]V
  ▽ UND;N;V;Z
[1]  L1:'GEBEN SIE DIE NUMMERN DER ANFRAGEN AN,DIE
      DURCH UND VERKNUEPFT WERDEN SOLLEN.'
[2]  V←,□
[3]  Z←V PR 1,ANZANF
[4]  →(Z≠1)/L1
[5]  L2:'GEBEN SIE DIE NUMMER AN,UNTER DER DIE UND-VERKNUEPFUNG
      ABGESPEICHERT WERDE SOLL : '
[6]  N←□
[7]  Z←N PR 0 0
[8]  →(Z≠1)/L2
[9]  N U V
  ▽

```

```

      VUNDA[ ]V
  ▽ UNDA;V1;V2;Z
[1]  L1:'GEBEN SIE DIE ERSTE GRUPPE VON NUMMERN AN,MIT
      DENEN EINE UND-VERKNUEPFUNG GEMACHT WERDEN SOLL : '
[2]  V1←,□
[3]  Z←V1 PR 1,ANZANF
[4]  →(Z≠1)/L1
[5]  L2:'GEBEN SIE DIE ZWFITE GRUPPE VON NUMMERN AN,MIT
      DENEN EINE UND-VERKNUEPFUNG GEMACHT WERDEN SOLL : '
[6]  V2←,□
[7]  Z←V2 PR 1,ANZANF
[8]  →(Z≠1)/L2
[9]  V1 UA V2
  ▽

```

)LOAD ZFICHPERS
SAVED 15.19.36 11/07/72

```
▽ZFEICHNE[□]▽
▽ ZFEICHNE;ZW;P;V
[1] ZW←FE 6 4 9
[2] P←(L/ZW[;1])-(L/ZW[;1])-1
[3] V←,ZW[;2]
[4] DEFAULT
[5] ORDINATE TEXT 'ANZAHL'
[6] ABSCISSA TEXT FE 6 4 5
[7] HISTOGRAM
[8] ORDINATE LOW LIMIT 0
[9] ORDINATE HIGH LIMIT[/V
[10] SET ORDINATE TOTAL(7[1+L((1+[V]))÷5) SPACED 5 NUMBERED 1
[11] ORDINATE FORMAT 'I4'
[12] ABSCISSA FORMAT 'I4'
[13] PLOTCHARS '□'
[14] →(O=FE 6 4 8)/L1
[15] ABSCISSA LOW LIMIT -1+L/ZW[;1]
[16] ABSCISSA HIGH LIMIT[/ZW[;1]
[17] SET ABSCISSA TOTAL(50[1+2×P) SPACED 2 NUMBERED 4
[18] COLUMN 1
[19] PLOT ZW
[20] →L2
[21] L1:ABSCISSA LOW LIMIT 0
[22] ABSCISSA HIGH LIMIT+[/V
[23] SET ABSCISSA TOTAL(50[1+2×+[/V) SPACED 2 NUMBERED 4
[24] PLOT V
[25] V←Q(6030)τZW[;1]
[26] V←V×Vε127
[27] V←V+27×V=0
[28] V←((26+LC),BL)[V]
[29] V←φV
[30] V←'6A1,X3,I4' ΔFMT(V;ZW[;2])
[31] 3ρLF
[32] 'ZUGEHÖRIGE TABELLE : '
[33] V
[34] →0
[35] L2:3ρLF
[36] 'ZUGEHÖRIGE TABELLE : '
[37] 3ρLF
[38] FE 6 4 9
```

▽

▽Z[□]▽

▽ Z

[1] ZFEICHNE

▽

ANHANG II

BEISPIELE

Bemerkung :

Es handelt sich bei den fuer die Beispiele benutzten Daten um alte und unvollstaendige Daten, sodass den Ergebnissen kein aktueller Wert zuzumessen ist.

)LOAD 161 NFUPERS
SAVED 13.16.02 11/07/72

STARTE
PASSWORT :
#####

STAND DER DATEN VOM 19. OCT 1972
SIE KOENNEN FOLGENDE ATTRIBUT VERWENDEN :

PERSONALNUMMER
KOSTENSTELLE
GEBURTSTAG
GEBURTSMONAT
GEBURTSJAHR
EINTRITTSTAG
EINTRITTSMONAT
EINTRITTSJAHR
BERFICH
TAETIGALS
WOHNORT
GESCHLECHT
FAMILIENSTAND
KINDERZAHL
ARBEITSZEIT
STAATSANGEH
AUSTRITT
GEHALTSGRUPPE
INSTITUT
KUNDIGUNGSGRUND
EHFGATTEBEFIGEK
ZULAGEN
BESCHVERH
STELLENPLAN
NAME
AUSTRITTSTAG
AUSTRITTSMONAT
AUSTRITTSJAHR

1. BEISPIEL : KINDERSTATISTIK BEZUEGLICH EINER KOSTENSTELLE
UND EINIGE NEBENFRAGEN

FINDE
UNTER WELCHER NUMMER SOLL DIE ANFRAGE ABGESPEICHERT WERDEN?
□:

5
FORMULIEREN SIE DIE ANFRAGE
KOSTENSELLE=541
KOSTENSELLE NICHT IN ATTRIBUTLISTE
GEBEN SIE KORREKTES ATRIBUT AN
KOSTENSTELLE

ANZAHL
GEBEN SIE DIE NUMMER DER ANFRAGE AN :
□:

5
57

RECHNE
GEBEN SIE DIE NUMMER DER ANFRAGE AN, MIT DER SIE RECHNEN WOLLEN :
□:

5
GEBEN SIE EIN ATTRIBUT AN, DAS BERECHNET WERDEN SOLL:
KINDERZAHL

MAXIMUM
3

MINIMUM
0

SUMME
68

MITTELWERT
1.192982456

TABELLE
SOLLEN ALLE GROESSEN TABELLIERT WERDEN ?
JA

KINDERZAHL ANZAHL
0 17
1 18
2 16
3 6

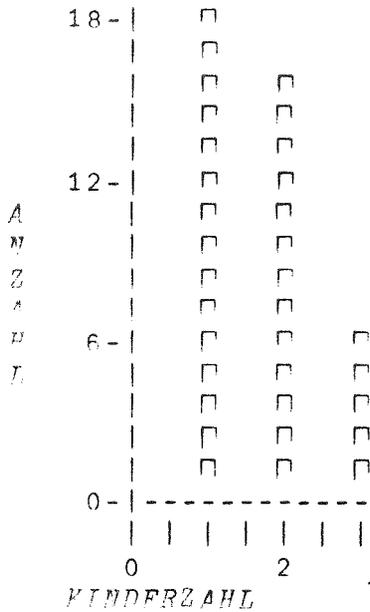
TABELLE
SOLLEN ALLE GROESSEN TABELLIERT WERDEN ?
NEIN
BITTE GEBEN SIE DIE GROESSEN AN, DIE IN DIE
TABELLE AUFGENOMMEN WERDEN :
□:

1 2 3

KINDERZAHL ANZAHL
1 18
2 16
3 6

)LOAD 161 ZFICHPERS
SAVED 16.26.14 11/03/72

ZFICHNF



ZUGEHÖRIGE TABELLE :

1	18
2	16
3	6

)LOAD 161 NFUPERS
SAVED 13.16.02 11/07/72

NFU

3 F 'GESCHLECHT=1'
4 F 'GESCHLECHT=2'

A 3

2857

A 4

822

UND

GEBEN SIE DIE NUMMERN DER ANFRAGEN AN, DIE DURCH UND VERKNÜPFET WERDEN SOLLEN.

□:

3 5

GEBEN SIE DIE NUMMER AN, UNTER DER DIE UND-VERKNÜPFUNG ABGESPEICHERT WERDEN SOLL :

□:

1

48 A 1
2 U 4 5
A 2

9 UNDA

GEBEN SIE DIE ERSTE GRUPPE VON NUMMERN AN, MIT DENEN EINE UND-VERKNUEPFUNG GEMACHT WERDEN SOLL :

□:

5

GEBEN SIE DIE ZWEITE GRUPPE VON NUMMERN AN, MIT DENEN EINE UND-VERKNUEPFUNG GEMACHT WERDEN SOLL :

□:

3 4

		3		4	

	5		48		9

3 R 'KINDERZAHL'
A 3

2857

TABELLE

SOLLEN ALLE GROESSEN TABELLIERT WERDEN ?

NEIN

BITTE GEBEN SIE DIE GROESSEN AN, DIE IN DIE TABELLE AUFGENOMMEN WERDEN :

□:

0 1 2 3 4 5 6 7 8 9 10

KINDERZAHL ANZAHL

0	947
1	673
2	798
3	312
4	89
5	24
6	11
7	1
8	2

MITTELWERT

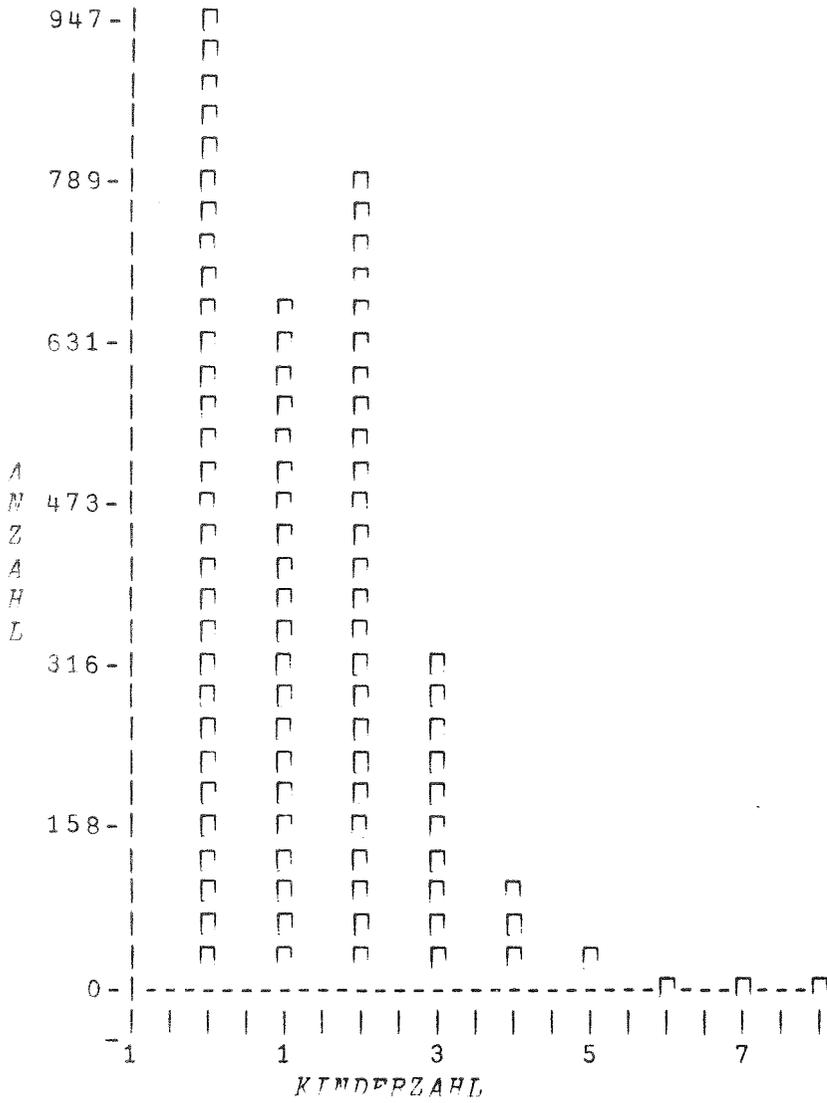
1.319565978

SUMME

3770

)LOAD 161 ZEICHPERS
SAVED 16.26.14 11/03/72

ZEICHNF



ZUGEHÖRIGE TABELLE :

0	947
1	673
2	798
3	312
4	89
5	24
6	11
7	1
8	2

• 2. BEISPIEL : SUCHE NACH EINEM BESTIMMTEN MITARBEITER

FINDE

UNTER WELCHER NUMMER SOLL DIE ANFRAGE ABGESPEICHERT WERDEN?

□:

7

FORMULIEREN SIE DIE ANFRAGE

NAME=SCHMID^WOHNORT=500

ANZAHL

GEBEN SIE DIE NUMMER DER ANFRAGE AN :

□:

7

5

LISTE

GEBEN SIE DIE NUMMER AN, UNTER DER DIE ANFRAGE ABGESPEICHERT WURDE :

□:

7

GEBEN SIE DURCH KOMMAS GETRENNT DIE ATTRIBUTE AN, DIE
GELISTET WERDEN SOLLEN :

PERSONALNUMMER, NAME, KOSTENSTELLE

3077 SCHMIDT WINFRIED D 123

5257 SCHMIDT ERICH 501

5776 SCHMIDT HEINZ 002

6439 SCHMID JUERGEN 370

8403 SCHMID ERNA 903

• WER HAT AM 29. FEBRUAR GEBURTSTAG ?

FINDE

UNTER WELCHER NUMMER SOLL DIE ANFRAGE ABGESPEICHERT WERDEN?

□:

10

FORMULIEREN SIE DIE ANFRAGE

GEBURTSTAG=29^GEBURTSMONAT=2

ANZAHL

GEBEN SIE DIE NUMMER DER ANFRAGE AN :

□:

10

1

10 L 'PERSONALNUMMER, INSTITUT, GEBURTSJAHR'

9178 AVW 56

LITERATURVERZEICHNIS :

- [1] *APL User's Manual* , IBM-Form GH20-0906 White Plains, 1970.
- [2] *APL PLUS File Subsystem Instruction Manual*. Toronto 1970.
- [3] *APL PLUS Plot Facility Instruction Manual*. Toronto 1970.
- [4] Hardy, G.H. und E.M.Wright : *Zahlentheorie*, Muenchen 1958.