

**KERNFORSCHUNGSZENTRUM
KARLSRUHE**

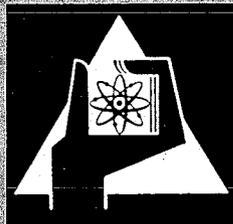
Juni 1971

KFK 1424

Datenverarbeitungszentrale

Netzplanberechnung mit dem Programmsystem TERM-2

P. Fette, P. Tack



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

Juni 1971

KFK- 1424

Datenverarbeitungszentrale

Netzplanberechnung mit dem Programmsystem TERM-2

von

P. Fette

P. Tack

Gesellschaft für Kernforschung mbH. Karlsruhe

Abstract

Das Programmsystem TERM-2 berechnet und zeichnet Netzpläne mit maximal 1000 Tätigkeiten. Es wurde Wert gelegt auf eine möglichst einfache Form der Dateneingabe. Der Bericht enthält neben dem Quellcode und einer Programmbeschreibung auch eine Einführung in die Netzplantechnik. Zur weiteren Erläuterung ist die Ein- und Ausgabe eines einfachen Netzplanbeispiels abgebildet. Die Netzplanzeichnung wird in der durch PERT eingeführten Form des Ereignisgraphen erstellt.

TERM-2 läuft in der GfK auf der Rechenanlage IBM 360/65 oder IBM 360/85 (in Zukunft auf der IBM 370/165) und dem CALCOMP Plotter-system 763/770.

The programsystem of TERM-2 computes and draws networks with a maximum size of 1000 tasks. The input to the program is kept easy. This report contains the source-code and a guide to the program and gives a short introduction to the technic of networks. The input and output of an example is given in pictures at the end of this report. Drawings of networks are done in the form introduced by PERT.

TERM-2 runs at the GfK on the IBM computer 360/65 or 360/85 (in future on IBM 370/165) and on the CALCOMP plotter system 763/770.

I n h a l t

1. Einleitung
2. Allgemeines zur Netzplantechnik mit dem Programmsystem TERM-2
 - 2.1. Aufteilung der Arbeiten
 - 2.2. Aufschlüsselung der Tätigkeiten in Zuständigkeitsbereiche
 - 2.2.1. Tätigkeiten
 - 2.2.2. Dauer der Tätigkeit
 - 2.2.3. Anfangs- und Endtätigkeit des Projektes
 - 2.3. Bestimmung der Folgetätigkeiten
 - 2.4. Pufferzeit
3. Genereller Aufbau des Programmsystems TERM-2
4. Unterschiede zu TERM-1
5. Ein- und Ausgabe
 - 5.1. Form der Eingabe
 - 5.1.1. Die Projektkarte
 - 5.1.2. Die Tätigkeitskarte
 - 5.2. Ausgabe
 - 5.2.1. Liste der Eingabekarten
 - 5.2.2. Kalenderausdruck
 - 5.2.3. Netzplan-Gesamtausdruck
 - 5.2.4. Ausdruck geordnet nach Zuständigkeiten
 - 5.2.5. Netzplanzeichnung
 - 5.2.6. Plotter-Information
 - 5.3. Eingabebeispiel
6. Fehlermeldungen
 - 6.1. Formale Fehler
 - 6.2. Loop Fehler
 - 6.3. Zeitfehler
 - 6.3.1. Kalenderüberschreitung
 - 6.3.2. Endterminüberschreitung
 - 6.4. Fehler bei der Netzplanzeichnung

1. Einleitung

Die Netzplantechnik ist für die Terminverfolgung von Großprojekten ein nützliches Hilfsmittel. Bei der Anwendung dieser Technik erweist es sich als sehr zweckmäßig, die optimale Terminberechnung, die Berechnung von Pufferzeiten und letztlich die graphische Darstellung des Netzplanes von einem Computer durchführen zu lassen. Dafür spricht die schnelle Verfügbarkeit eines vom Computer berechneten und gezeichneten Netzplanes. Weil eine leistungsfähige Rechenanlage nur wenige Minuten braucht für eine komplette Berechnung und Zeichnung eines Netzplanes, ist für die Projektleitung eine sich ändernde Terminalsituation praktisch sofort zu durchschauen. Voraussetzung dafür ist allerdings, daß die Dateneingabe möglichst einfach und schnell zu erstellen ist.

Im Zusammenhang mit den Planungsarbeiten für das Projekt "Schneller Brüter" wurde 1966 das Netzplanprogramm TERM-1 entwickelt [1]. Es wurde speziell für die damals im Kernforschungszentrum Karlsruhe installierte Rechenanlage IBM-7074 geschrieben. Die heutige Installation mit den Maschinen IBM 360/65 und IBM 360/85 in Verbindung mit dem Calcomp-Plottersystem 763/770 erlauben ein erheblich schnelleres und umfangreicheres Programm zur Erstellung von Netzplänen. So wurde aufbauend auf TERM-1 das Programmsystem TERM-2 geschrieben. Um dem Benutzer, der schon mit TERM-1 gearbeitet hat, möglichst wenig zusätzliche Arbeit zu machen, ist bei TERM-2 die Form der Eingabekarten im wesentlichen beibehalten worden. Die Ausgabe der Ergebnisse in Form von Listen ist ebenfalls die gleiche wie bei TERM-1. Zusätzlich ist es bei TERM-2 wahlweise möglich, einen Netzplan in der Form des Ereignisgraphen — nach PERT — gezeichnet zu bekommen.

2. Allgemeines zur Netzplantechnik mit dem Programmsystem TERM-2

Der Netzplan ist die Schematisierung des Arbeitsablaufes eines Projektes beginnend bei einer Anfangstätigkeit und endend mit einer Abschluß- oder Endtätigkeit. Voraussetzung für die Berechnung eines Netzplanes ist ein sinnvolles Zergliedern des Projektes in einzelne Tätigkeiten. Dieses Zergliedern in Einzeltätigkeiten kann im Prinzip nicht vom Computer vorgenommen werden und stellt einen schwierigen Teilabschnitt der Arbeit an einem Netzplan dar, der nun vom Menschen zu lösen ist. Planungstechniker und Sachbearbeiter haben dabei folgende Aufgaben zu lösen:

2.1. Aufteilung der Arbeiten am Projekt nach Zuständigkeitsbereichen (z.B. Arbeiten für Maurer, Elektrotechniker, Schlosser usw.)

2.2. Aufschlüsselung der Tätigkeiten innerhalb eines jeden Zuständigkeitsbereiches nach folgendem Prinzip:

2.2.1. Welche Tätigkeiten müssen ausgeführt werden ? Hier kommt es darauf an, eine gute aber nicht zu weitgehend detaillierte Struktur aufzustellen, damit der spätere Netzplan für Projekt- und Abteilungsleiter noch übersichtlich genug ist. Beispiel: Für den Zuständigkeitsbereich Elektriker wird es genügen, neben anderen z.B. die Tätigkeiten zu spezifizieren:

I. Kabel verlegen und

II. Beleuchtungskörper anschließen.

Es würde der Übersichtlichkeit eines Netzplanes schaden, würde man sämtliche Handgriffe des I. Kabelverlegens oder des II. Beleuchtungskörperanschließens als selbständige Tätigkeiten spezifizieren.

Wichtig ist jedoch, daß diese beiden Tätigkeiten überhaupt voneinander unterschieden werden, da z.B. zwischen I und II noch Tätigkeiten aus anderen Zuständigkeitsbereichen liegen werden; z.B. das Verputzen und Tapezieren der Wände.

Eine optimale Aufschlüsselung der Tätigkeiten richtet sich danach:

- für wen der Netzplan aufgestellt werden soll; der Projektleiter braucht u.U. einen weniger detaillierten Plan als der Bauleiter.
- die Tätigkeiten müssen gerade soweit aufgeschlüsselt werden, daß sie für die entsprechende Arbeitsgruppe deutlich genug spezifiziert sind.

Droht ein Plan dennoch unübersichtlich zu werden, ist es sinnvoll, Teilnetzpläne für definierte Unterprojekte aufzustellen.

2.2.2. Schätzen der Zeitdauer jeder Tätigkeit (bei TERM-2 ist die Angabe in Arbeitswochen erforderlich, eine Arbeitswoche = 5 Tage und demnach ist als kleinste vom Programm zu verarbeitende Einheit der Arbeitszeit 1 Tag = 0,2 Arbeitswochen).

2.2.3. Bestimmung der Anfangs- und der Endtätigkeit des Projektes. Für die Berechnung und graphische Darstellung eines Netzplanes mit dem Programmsystem TERM-2 muß jeweils ein Anfangs- und Endpunkt definiert sein.

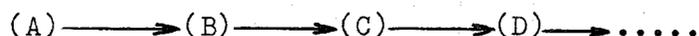
2.3. Bestimmung der Folgetätigkeiten einer Tätigkeit (siehe auch 5.1.).

Alle Tätigkeiten eines Projektes außer der Anfangstätigkeit können nur ausgeführt werden, wenn zuvor andere Tätigkeiten beendet sind. Es ergeben sich somit Ketten von hintereinander aufgereihten Tätigkeiten. Das letzte Glied jeder Kette ist die Endtätigkeit. Diese Ketten werden Wege durch den Netzplan genannt. Die Dauer jeder einzelnen Tätigkeit ist dabei eine charakteristische Eingabegröße, die für eine graphische Darstellungsform des Netzplanes ausgenutzt werden kann. Mit Länge eines Weges wird die Summe der Zeiten der einzelnen Tätigkeiten dieses Weges bezeichnet.

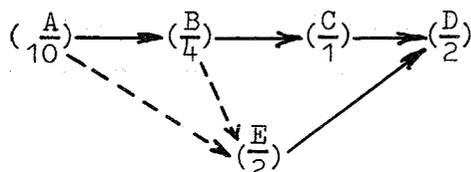
Der Begriff Folgetätigkeit soll differenziert werden in

- unbedingte Folgetätigkeit und
- bedingte Folgetätigkeit.

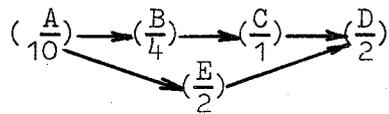
Für die Netzplanberechnung mit TERM-2 ist es nun wichtig, für jede Tätigkeit (außer der Endtätigkeit) eine oder mehrere Nachfolgetätigkeiten anzugeben. Auf diese Weise wird dem Programm mitgeteilt, wie die Wege durch den zu berechnenden Netzplan fixiert sind. Für einige Wege wird es nicht schwierig sein, zu jeder Tätigkeit die Nachfolgetätigkeit anzugeben, nämlich immer dann, wenn eine Tätigkeit verlangt, daß vor ihrer Ausführung eine andere Tätigkeit unbedingt beendet sein muß.



In dieser Skizze ist (B) eine unbedingte Folgetätigkeit von (A) und (C) eine unbedingte Folgetätigkeit von (B) usw. In einem optimal berechneten Netzplan gibt es mindestens einen Weg, der nach der Anfangstätigkeit bis zur Endtätigkeit nur aus Arbeiten besteht, die zugleich auch alle unbedingte Folgetätigkeiten sind. Dieser Weg wird der kritische Weg genannt, wenn die Länge dieses Weges gleich ist der Summe der unter 2.2.2. geschätzten Zeiten für jede auf diesem Weg liegende Tätigkeit. Diese Länge gibt damit gleichzeitig die Gesamtdauer des Projektes an. Nun gibt es aber auch Tätigkeiten, die zwar unbedingte Folgetätigkeiten haben können, die aber selbst nicht auch unbedingte Folgetätigkeit sein müssen.



In dieser Skizze ist (E) eine bedingte Folgetätigkeit; denn sie kann sowohl als Folgetätigkeit von (A) als auch von (B) genannt werden. An welcher Stelle bedingte Folgetätigkeiten als Nachfolger definiert werden, kann recht wichtig sein für die Gesamtdauer des Projektes:

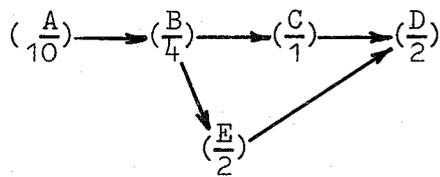


$\frac{B}{4}$ bedeutet: Tätigkeit B mit 4 Wochen Dauer.

Bei dieser Einteilung - (B) und (E) als Folgetätigkeiten von (A) - wird die Gesamtdauer des Projektes aus den 5 Arbeiten A, B, C, D und E von dem kritischen Weg A - B - C - D bestimmt mit den Zeiten $10 + 4 + 1 + 2 = 17$ Wochen.

Für die Ausführung von Tätigkeit E mit 2 Wochen ist jetzt ein Spielraum vorhanden. E braucht nicht beendet zu sein, bevor nicht B und C beendet sind, die oben zusammen $4 + 1 = 5$ Wochen benötigen. Für E läßt sich also ein Spielraum $5 - 2 = 3$ Wochen errechnen. Ein derartiger Spielraum wird totale Pufferzeit genannt. Die totale Pufferzeit ist ein Zeit-Spielraum, der verbraucht werden kann ohne daß der Endtermin gefährdet wird.

Wenn E als Folgetätigkeit von B angegeben wird, verschiebt sich in diesem Beispiel die Terminsituation:



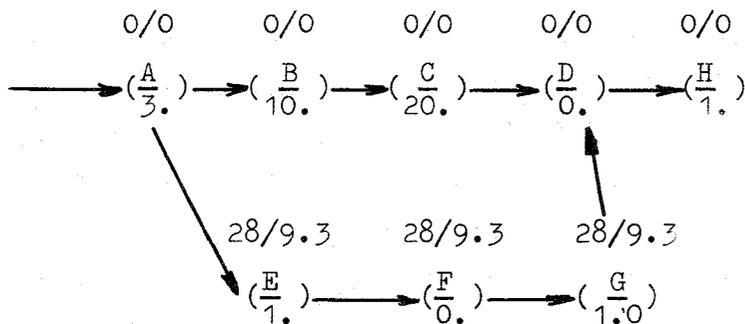
Es entsteht ein anderer kritischer Weg, nämlich A - B - E - D, die Gesamtdauer erhöht sich auf 18 Wochen. Die Tätigkeit C erhält eine totale Pufferzeit von $2 - 1 = 1$ Woche.

Aus diesen beiden Skizzen ist ersichtlich, wie die Plazierung der bedingten Folgetätigkeiten sich auf die Terminsituation auswirken kann. Der Planungstechniker wird jedoch immer günstig liegen, wenn er bedingte Folgetätigkeiten dort angibt, wo sie frühestens angegeben werden können.

In den Eingabekarten für das Programmsystem TERM-2 wird nicht unterschieden zwischen unbedingter und bedingter Folgetätigkeit (siehe 5.1.). Diese Differenzierung ist hier nur gemacht, um dem Planungstechniker Hilfsgrößen zu definieren, deren Veränderung bei wiederholter Netzplanberechnung eventuell günstigere Endtermine bewirken kann.

2.4. Die totale Pufferzeit ist oben schon definiert worden.

Mit freier Pufferzeit wird ein Zeit-Spielraum bezeichnet, der bei der Einmündung eines Weges W1 mit der totalen Pufferzeit P1 in einen zweiten Weg W2 mit der totalen Pufferzeit P2 als Differenz $P1 - P2$ der letzten Tätigkeit des Weges W1 zur Verfügung steht. Wenn man die freie Pufferzeit auf alle Tätigkeiten des zugehörigen Weges gleichmäßig verteilt, erhält man die verteilte Pufferzeit; nach dieser Definition steht den Tätigkeiten E, F, G jeweils eine verteilte Pufferzeit von 9,3 Wochen zur Verfügung. Der kritische Weg in dieser Skizze heißt A-B-C-D-H



$\frac{A}{3.}$ bedeutet: Tätigkeit A mit 3,0 Wochen Dauer

28/9.3 bedeutet: Totale Pufferzeit 28 Wochen.

Verteilte Pufferzeit 9,3 Wochen.

3. Genereller Aufbau des Programmsystems TERM-2 (siehe Abb. 1)

Ein in ØS-Assembler geschriebenes Rahmenprogramm TERM2 (Abb. 2) steuert den Programmablauf. Darin wird zunächst geprüft, ob neben der Netzplanberechnung auch eine Netzplanzeichnung gewünscht wird. Wenn ja, muß auf der ØS-360 EXEC-Karte der Parameter PARM='NETZPLAN' oder PARM='NETZPLAN=XFACH' angegeben werden (mit $0 < X < 6$, wenn die Zeichnung x-fach ausgeführt werden soll). In diesem Fall wird im Rahmenprogramm eine Zeichenkette mit dem Inhalt: TERMPc bereitgestellt, andernfalls heißt diese Zeichenkette TERMPT. Danach wird in dem Rahmenprogramm das eigentliche Programm der Netzplanberechnung TERM (Abb. 3) dynamisch geladen und die Zeichenkette TERMPc oder TERMPT als Parameter übergeben. Wird TERMPc übergeben, also wenn eine Netzplanzeichnung gewünscht wird, so wird zu Beginn des in PL/1 geschriebenen Programms TERM ein Schalter gesetzt, der nach erfolgreicher Netzplanberechnung bewirkt, daß eine Datei mit Plotterzeichnungsinformation beschrieben wird. In TERM wird zunächst eine umfangreiche Fehlersuche durchgeführt (siehe 6.). Wenn keine Fehler gefunden wurden, optimiert das Programm in mehreren Iterationen zunächst die Zahl der Arbeitstage des kritischen Weges. Dann werden Einzeltermine unter Berücksichtigung diverser Feiertage berechnet, die ein eingebauter Kalender bis zum 31.12.1986 bereithält. (Als arbeitsfrei gelten folgende Tage: Samstag, Sonntag; 1. und 6. Januar; Karfreitag, Ostermontag, Christi Himmelfahrt, Pfingstmontag; 1. Mai, 24. bis 31. Dezember.) Nach Berechnung der Pufferzeiten liegen nunmehr alle Netzplandaten vor, so daß es zur Druckausgabe kommen kann und danach zum Beschreiben einer Datei mit dem Namen GRAPH für die Plotter-Zeichnungsinformation, wenn eine Netzplanzeichnung erwünscht ist.

Nach der Rückgabe der Kontrolle an TERM2 wird ein Fehlercode abgefragt. Wenn TERM erfolgreich einen oder mehrere Netzpläne berechnet hat, wird wiederum dynamisch zu einem in FØRTRAN IV geschriebenen Plotterprogramm verzweigt mit dem Namen CALCPL. Bei fehlerhafter Eingabe oder wenn keine Zeichnung gefordert ist, wird der Aufruf von CALCPL umgangen.

Das FØRTRAN Hauptprogramm CALCPL (siehe Abb. 4) ruft zunächst eine Initialisierungssubroutine CALCIN (Abb. 5) für den Calcomp-Plotter auf. Dabei werden der Jobname und die Startzeit des Jobs in der Maschine als erste Zeichnungsinformation für den Plotter übergeben. Eine Nachricht wird über die Konsolschreibmaschine an den Operator geschrieben. Diese Nachricht ist aus betriebsinternen Gründen für den off-line Plotterbetrieb nötig. Eine 2. Subroutine ZAEHL (siehe Abb. 6), die in CALCPL aufgerufen

wird, holt die Zahl X des Parameters der EXEC-Karte: PARM='NETZPLAN=XFACH'. ($0 < x < 6$). Die danach aufgerufene Subroutine TERMPL. (Abb. 7) liest den von TERM angelegten Data-Set und bewirkt die eigentliche Netzplanzeichnung. TERMPL wird x-mal angesprungen, auf diese Weise ist max. eine 5-fache Ausfertigung eines Netzplanes möglich.

Folgende Subroutinen der Calcomp-Software werden vom Programmsystem TERM-2 aufgerufen:

1. in der Subroutine CALCIN die Calcomp-Routine PLØTS
2. in der Subroutine TERMPL die Calcomp-Routine PLØTC
- in der Subroutine TERMPL die Calcomp-Routine SYMBØL
- in der Subroutine TERMPL die Calcomp-Routine NUMBER
- in der Subroutine TERMPL die Calcomp-Routine ARØHD

Die CPU-Rechenzeit für einen Netzplan bestehend aus 163 Tätigkeiten und 302 Verknüpfungen (= 302 mal wurden Tätigkeiten als Folgetätigkeiten genannt) inklusive Bereitstellung der Plotterinformation beträgt ca. 15 sec auf der IBM 360/85 und ca. 75 sec auf der IBM 360/65. Die Ausführung dieser Netzplanzeichnung mit dem Calcomp-Plotter 763/770 dauert ca. 35 Minuten (Länge der Zeichnung 2 m, Höhe 0,7 m).

4. Unterschiede zu TERM-1

Die äußere Form der Eingabe für TERM-2 richtete sich aus nach der Eingabe für TERM-1 und ist im wesentlichen beibehalten worden. Lediglich die für TERM-1 nötige 2. Eingabekarte mit Steuerinformation konnte bei TERM-2 entfallen. Die Steuerinformationskarte enthält bei TERM-1 folgende Größen:

1. Anzahl der Zwischentermine
2. Anzahl der Fertigmeldungen
3. Entscheidung, ob Zuständigkeitsausdruck und Balkendiagramm oder nicht
4. Entscheidung, ob Ausdruck geordnet nach totaler Pufferzeit und Rang gewünscht wird
5. Entscheidung, ob ausführliche Fehlersuche durchgeführt werden soll oder nicht

Diese Optionen bedeuten, wenn sie durchgeführt werden sollen, für TERM-1 eine erhebliche Steigerung der Rechenzeit auf der IBM 7074. In TERM-2 wurden die Optionen 3), 4) und 5) grundsätzlich eingebaut, da die Rechenzeit an den Maschinen IBM 360/85 und IBM 360/65 auch für umfangreichere Programme ohnehin nur im Sekunden- bis Minutenbereich liegt. Jedoch ist zur Zeit ein Balkendiagrammausdruck nicht eingebaut. Auf die Bearbeitung von Zwischenterminen und Fertigmeldungen wurde verzichtet:

- a) weil das Programm ohnehin optimiert und diese Termine an Hand des Ausdrucks der Wege durch den Netzplan (siehe 5.2) und der Liste geordnet nach Zuständigkeitsbereichen abgelesen werden können.
- b) Weil die Eingabe dadurch weniger kompliziert ist.

5. Ein- und Ausgabe

5.1. Form der Eingabe (siehe Abb. 8)

Eine Anleitung für die Vorarbeiten zur Gewinnung der Eingabedaten ist unter 2.1, 2.2 und 2.3 gegeben worden. Um der Maschine die Sortiervorgänge zu erleichtern ist es nötig, jede Tätigkeit zu nummerieren. Dafür können Zahlen von 1 bis 9999 gewählt werden. Zusätzlich zu dieser Numerierung wird auch der Name der Tätigkeit genannt. Er darf aber nicht länger als 20 Zeichen sein. Die für TERM-1 angegebene Beschränkung auf max. 1000 Tätigkeiten und 1200 Verknüpfungen wurde auch für TERM-2 beibehalten.

Die Angabe: Zuständigkeitsbereich (=Name der Firma oder sonstige Institution, die die Tätigkeit ausführt) darf maximal 5 Zeichen (Zahlen oder/und Buchstaben) lang sein.

Als Sonderzeichen für die Namensgebung (Zuständigkeitsbereich und Tätigkeitsname) sind alle in FORTRAN und PL/1 bekannten Sonderzeichen erlaubt, bzw. alle Zeichen, die über den IBM-Schnelldrucker ausgegeben werden können. Der Calcomp-Plotter kennt jedoch nicht alle diese Sonderzeichen und würde in der Zeichnung die unbekanntenen Zeichen mit einer Leerstelle übergehen. Deshalb sollte man, wenn man auch Netzplanzeichnungen erhalten möchte, sich auf folgende Sonderzeichen beschränken: + | _ # - / . < () & \$ * ; → , % > ? :

ⓐ ' = " !

Zwei unterschiedliche Arten von Eingabekarten sind nötig für die Berechnung eines Terminplanes mit TERM-2:

1. Projektkarte
2. Tätigkeitskarte

5.1.1. Die Projektkarte enthält den Namen des Projektes, den gewählten Anfangstermin, evtl. einen Endtermin, sowie als drittes ein beliebiges Datum, z.B. den Tag an dem die Berechnung durchgeführt wird.

Folgende fest vorgeschriebene Form ist beim Ablochen dieser Karte einzuhalten:

| Lochkarten- spalte Nr. | Text | Bemerkung |
|---------------------------|-------------|--|
| 1 | * oder @ | ein in TERM-1 definiertes Trennungszeichen, das in TERM-2 der Einfachheit halber beibehalten wurde. |
| 2 - 51 | Projektname | bestehend aus maximal 50 Buchstaben, Zahlen und Sonderzeichen. |
| 52 | * oder @ | wie in Spalte 1 |
| 54 + 55 | Starttag | } z.B. für das Startdatum 6. Mai 1971 muß stehen: 6 5 71 |
| 57 + 58 | Startmonat | |
| 60 + 61 | Startjahr | |
| 63 + 64 | Endtag | } Beispiel: 19 6 72, wenn das Programm zusätzlich zur Netzplanberechnung prüfen soll, ob der auf diese Weise eingegebene Endtermin eingehalten werden kann. Für den Fall, daß eine solche Prüfung unterbleiben kann, loche man in die Spalten 64, 67 und 70 je eine 0. Die Termingestaltung bezieht sich aber in beiden Fällen auf den von TERM-2 optimal berechneten Endtermin. |
| 66 + 67 | Endmonat | |
| 69 + 70 | Endjahr | |
| 72 + 73 | Tag | } Report-Datum = Datum der Eingabe des Programms oder andere beliebige Zahlen, z.B. zur Identifikation des Rechenlaufs. |
| 75 + 76 | Monat | |
| 78 + 79 | Jahr | |

5.1.2. Die Tätigkeitskarten übergeben dem Programm die eigentlichen Daten für die Netzplanberechnung. Folgende Angaben werden hier benötigt:

- a) Nummer der Tätigkeit
- b) Zuständigkeitsbereich
- c) Name der Tätigkeit
- d) Dauer der Tätigkeit in Wochen
- e) Anzahl der unbedingten und bedingten Folgetätigkeiten für diese Tätigkeit
- f) Nummern dieser Folgetätigkeiten
- g) Endmarke

zu a) Jede Tätigkeit wird numeriert. Zulässig sind die Zahlen von 1 bis 9999.

zu b) Zuständigkeitsbereich darf mit maximal 5 Zeichen (Buchstaben, Zahlen und Sonderzeichen) angegeben werden. Da die Zuständigkeit als Sortiergröße verwendet wird, müssen die Zuständigkeitsbezeichnungen spaltengebunden völlig identisch sein für alle die Tätigkeiten, die unter denselben Zuständigkeitsbereich fallen. Jede Abweichung wird als eigener Zuständigkeitsbereich verstanden.

zu c) (siehe auch oben) Der Tätigkeitsname wird durch max. 20 Zeichen angegeben.

zu d) Wie schon erwähnt, bezieht sich TERM-2 auf die 5-Tage Woche. Ein Arbeitstag entspricht also 0,2 Arbeitswochen. Die Tätigkeitsdauer ist stets als Gleitkommazahl im F-Format anzugeben. 13 Wochen Dauer muß also als 13.0 oder 13. angegeben werden.

zu e) Bei z.B. 15 vorhandenen Folgetätigkeiten muß die Zahl 15 angegeben werden.

zu f) In beliebiger Reihenfolge werden hier die Tätigkeitsnummern (siehe oben) der Nachfolgetätigkeiten angegeben. (Zur Definition der Folgetätigkeiten siehe 2.3) Max. 50 Folgetätigkeiten können hier angegeben werden.

zu g) Nach der Nummer für die letzte Folgetätigkeit muß eine Endmarke gesetzt werden.

Endmarke = 1 bedeutet: Es folgt eine weitere Tätigkeitskarte,
Endmarke = 0 bedeutet: Dieses war die letzte Eingabekarte und damit zugleich die Tätigkeitskarte für die letzte - für die End - Tätigkeit.

Beispiel für eine Tätigkeitskarte:

Folgende äußere Form muß für das Ablocken von Tätigkeitskarten eingehalten werden:

| Lochkarten Spalte Nr. | Text | Bemerkungen |
|--------------------------|--|---|
| 1 - 4 | Tätigkeitsnr. | laufende Nr. der Tätigkeiten |
| 5 | Leerspalte | |
| 6 | * oder @ | Trennungszeichen s. 5.1.1. |
| 7 - 11 | Zuständigkeitsbereich(max. 5 Zeichen) | |
| 12 - 31 | Name der Tätigkeit (max. 20 Zeichen) | |
| 32 | * oder @ | Trennungszeichen wie bei Spalte 6 |
| 33 - 39 | Zeitdauer | als Gleitkommazahl z.B. 13.6 oder 2. |
| 40 - 41 | Anzahl der Folgetätigkeiten | |
| 42 - 46 | Tätigkeits- Nr. der Folge- tätigkeit | jeweils 5 Lochspaltenplätze für die max. 4-stellige Nummer einer Folge- tätigkeit. Nach der letzten Nr. folgt eine Leerspalte und danach die entspre- chende Endmarke 0 oder 1. |
| 47 - 51 | | |
| 52 - 56 | | |
| . | | |
| . | | |

Bei mehr als 7 Folgetätigkeiten muß eine Fortsetzungskarte beschrieben werden. Dabei bleibt die 1. Spalte leer, um den Rhythmus von 5 Lochspaltenplätzen für die Nr. der Folgetätigkeiten zu wahren. Max. 50 Folgetätigkeiten ergeben 3 Fortsetzungskarten. Ein Beispiel für die Eingabe ist in Abb. 8 zu finden. Mehrere Netzplanmodelle dürfen hintereinander gesteckt eingegeben werden.

5.2. Ausgabe

TERM-2 erzeugt 6 unterschiedliche Ausgaben:

5.2.1. Liste der Eingabekarten (siehe Abb. 9) Zur Korrektur eventueller Fehler wird jede Eingabekarte gelistet. Sie wird dabei unterteilt. Als erstes erscheint:

```
|Tätigkeits-Nr. | Zuständigkeitsbereich | Name der Tätigkeit | Dauer |
```

Darunter etwas eingerückt:

```
|Anzahl der Folgetätigkeiten | Nummern der Folgetätigkeiten|
```

Die Endmarke sowie die Trennungszeichen * oder @ werden in dieser Liste nicht ausgedruckt, jedoch werden hier mögliche Fehler nach 6.1. und 6.2. diagnostiziert.

5.2.2. Kalenderausdruck (siehe Abb. 10) Beginnend mit N-DAY=n, gibt n die Anzahl der Arbeitstage an. Danach folgt ein Kalenderausdruck für den 1. bis zum letzten Arbeitstag in folgender Form:

```
| A | B | C | D | E | F |
```

A entspricht der laufenden Nr. aller Tage.

B eine Angabe, ob an dem Tag nicht gearbeitet wird.

C entspricht der laufenden Nr. der Arbeitstage.

D entspricht dem Wochentagsnamen.

E entspricht dem Datum in der Form = Tag Monat Jahr

F entspricht der laufenden Nr. des Kalendertages.

Im Anschluß an den Kalenderausdruck erfolgt die Meldung:

KLEINE FEHLER GEFUNDEN. Möglich sind an dieser Stelle auch Nachrichten über Zeitfehler, siehe 6.3.

5.2.3. Netzplan-Gesamtausdruck (siehe Abb. 11) Dieser Ausdruck ist für den Netzplantechniker und für die Projektleitung bestimmt. Er enthält alle Tätigkeiten des Netzplanes in Wege zusammengefaßt und diese nach ansteigender totaler Pufferzeit geordnet. In einer Überschrift erscheint der Name des Projektes, das Start-, End- und das Reportdatum. Das Enddatum ist in jedem Fall der vom Programm berechnete Endtermin. Danach folgt die Liste der Tätigkeiten unter folgender Überschrift:

1. NR. : Tätigkeits-Nummer
2. ABT : Zuständigkeitsbereich
3. TAETIGKEIT : Name der Tätigkeit
4. DAUER : Dauer der Tätigkeit wie auf der Eingabekarte angegeben
5. F.BEG. : Frühester Beginn. Angabe in Arbeitswochen ab Starttag
6. ASP.END : Allerspätestes Ende der Tätigkeit =
Frühester Beginn + totale Pufferzeit + Dauer
(Angabe in Arbeitswochen ab Starttag)
7. FS.END : Frühestes Ende =
Frühester Beginn + verteilte Pufferzeit + Dauer
(in Arbeitswochen ab Starttag)
8. T.PFZ : Totale Pufferzeit (Angabe in Arbeitswochen)
9. F.PFZ : Freie Pufferzeit (Angabe in Arbeitswochen)
10. NR/WEG : Lfd. Nr. der Tätigkeit des entsprechenden Netzplanweges (beginnend mit 0).
11. VT.PFZ : Verteilte Pufferzeit (Angabe in Arbeitswochen)
12. FR.BEG. : Datum des frühesten Beginns der Tätigkeit ergibt sich, wenn alle vorangehenden Tätigkeiten auf demselben Weg ohne Pufferzeit zu verbrauchen beendet würden.
13. ASP.END : Datum des allerspätesten Endes der Tätigkeit (gebildet wie ASP.END) (nur sinnvoll für die letzte Tätigkeit auf einem Weg)
14. SP.END : Datum des Endtermins der Tätigkeit =
Sollbeginn + Dauer + verteilte Pufferzeit
(Sollbeginn s. 5.2.4.). Bei diesem Endtermin jeder Tätigkeit wird die gesamte Terminalsituation nicht gefährdet.

5.2.4. Ausdruck geordnet nach Zuständigkeiten (Abb. 12)

Für den Leiter eines Zuständigkeitsbereichs ist der Gesamtausdruck des Netzplanes unübersichtlich, deswegen wird eine Liste der Tätigkeiten inklusive Termine geordnet nach Zuständigkeitsbereichen ausgegeben. Innerhalb dieser Liste wird nach ansteigender verteilter Pufferzeit sortiert. Folgende Angaben erscheinen in dieser Liste:

1. NR. : Tätigkeitsnummer
2. ABT. : Zuständigkeitsbereich
3. TAETIGKEIT : Name der Tätigkeit
4. DAUER : Dauer der Tätigkeit, wie auf der Eingabekarte angegeben
5. FRUEH.BEGINN : Datum des frühesten Beginns der Tätigkeit (siehe oben)
6. SØLL-BEGINN : Datum des Soll-Beginns. Das ist der Termin, der sich ergibt aus
Frühestem Beginn + (n-1) • verteilte Pufferzeit,
wobei die infrage kommende Tätigkeit die n'te
Tätigkeit auf dem entsprechenden Netzplan-Weg ist.
7. SØLL-ENDE : Datum von Soll-Beginn + Dauer der Tätigkeit
8. VT.PFZ. : Verteilte Pufferzeit = Totale Pufferzeit/Anzal der
Tätigkeiten auf dem entsprechenden Netzplan-Weg
(siehe auch 2.4.)

5.2.5. Netzplanzeichnung (Beispiel siehe Abb. 13)

Man hat bei TERM-2 die Möglichkeit, einen Netzplan in der Form des Ereignisgraphen nach PERT gezeichnet zu bekommen. Die Steuerung für diese zusätzlich wählbare Ausgabeform geschieht durch Angabe eines Parameters auf der EXEC-Jobkontrollkarte:

PARAM=NETZPLAN bei einer Zeichnung oder
PARAM='NETZPLAN=XFACH'

wobei $1 \leq x \leq 5$ sein kann. Mit x wird die Anzahl der Duplikate angegeben (wenn z.B. 3 Netzpläne zur Berechnung und Zeichnung eingegeben werden und PARAM='NETZPLAN=2FACH' auf der EXEC-Karte angegeben wurde, wird jeder Netzplan 2-fach gezeichnet, der Benutzer erhält also 6 Zeichnungen)

Wege, die identisch mit anderen Wegen verlaufen würden, werden in der Plot-Routine TERMPL in Form von Rechteck-Zügen von dem zuerst gezeichneten Weg getrennt und somit übersichtlich differenziert. Wege, die zum Teil einen anderen Weg überschneiden, werden in Form eines Winkelzuges über den schon bestehenden Weg gezeichnet. Dabei ist der eine Schenkel des Winkels als Strichellinie ausgeführt. Scheintätigkeiten werden als senkrechte Strichellinie dargestellt.

Es lag nahe, in einem computergezeichneten Netzplan als Abszisse die Zeit zu wählen. Wird jedoch eine lineare Proportionalität gewünscht, steht man vor dem Dilemma, in den meisten Fällen sehr unhandlich lange Netzpläne zu bekommen, denn die Strecke für die kleinste Zeiteinheit - also 1 Tag - muß immerhin so lang sein, daß sie gut lesbar beschriftet werden kann mit:

1. Tätigkeitsnummer
2. Name der Tätigkeit
3. Dauer
4. Zuständigkeitsbereich

So erscheint es sinnvoll, eine konstante Strecke (1 inch = 25,4 mm) zu definieren für den Weg zwischen 2 zeitlich aufeinander folgende Ereignisse, unabhängig davon, wie lang die Zeitspanne zwischen diesen Ereignissen ist. Als Ereignis ist der jeweils nächste früheste Beginn einer oder mehrerer Arbeiten anzusehen.

Die Zeichnung kann nun mit einem Rahmen versehen werden, auf dem in Abszissenrichtung äquidistante Markierungen für jedes Ereignis gezeichnet werden können. Die Markierungen werden beschriftet mit dem Datum des frühesten Beginns der hier fälligen Tätigkeiten (in Ordinateenrichtung geschrieben) und mit der bis zu dem Ereignis verstrichenen Arbeitszeit (= F.BEG siehe 5.2.3.). Außerdem werden an jeder Barke die Nummern der hier beginnenden Tätigkeiten geschrieben. Auf diese Weise wird das Suchen bestimmter Tätigkeiten auch in komplizierten Netzplänen erleichtert.

Die Tätigkeitsnummern werden doppelt um 0,01 inch versetzt geschrieben in einer Schrifthöhe von 5 mm, so daß selbst bei möglicherweise dicht aneinander gesetzten Beschriftungen mit Sicherheit die Tätigkeitsnummer gelesen werden kann. (Bei komplizierten Netzplänen können manche Wege dicht nebeneinander gezeichnet werden) Die Grenze der Übersichtlichkeit eines computergezeichneten Netzplanes liegt vielleicht bei NT = 200 - 300 Tätigkeiten. Bei entsprechend günstiger Benennung von Folgetätigkeiten können jedoch erheblich mehr Tätigkeiten in einem übersichtlichen Netzplan gezeichnet werden. Die Breite der Zeichnung beträgt 70 cm, die Länge ist von der Anzahl der Ereignisse abhängig für die jeweils ein unterschiedlicher frühester Beginn der Tätigkeiten errechnet wurde.

5.2.6. Plotter-Information (siehe Abb. 14)

Eine Information über die zu plottenden Daten wird in der Ausgabeliste unter FT06FOO1 gedruckt. Hier erscheint zunächst der Name des Projektes gemäß 5.1.1. Dann werden mit NE die Anzahl der Ereignisse und NT die Anzahl der Tätigkeiten angegeben. Der vom PL/1 Programm angelegte Plot-Dataset wird danach ausgedruckt. A(n,1) und A(n,2) sind die Linienzüge, die den Tätigkeiten entsprechen. Die Koordinaten der Endpunkte U (entspricht der Abszisse und ist an dieser Stelle noch linear proportional der Zeit) und Z (zugehörige Ordinaten) wurden in der Plot-Routine TERMPL umgeformt zu X (in der oben definierten Abszissen-Einheit) und Y (eine Spreizung der Z-Werte auf den max. Zeichenbereich in Ordinatenrichtung von 70 cm). Für jeden Linienzug = Tätigkeit werden im Anschluß daran die Daten ausgedruckt:

- frühester Beginn
- Tätigkeits-Nummer
- Linienzuginformation: SHIFT, I1, I2
- Koordinaten des Anfangs- und Endpunktes der Linie
X1, Y1, X2, Y2
- Zuständigkeitsbereich = ABT

Wenn der Linienzug kleiner als 2 inch wird, muß in der Zeichnung der Tätigkeitsname, der max. 20 Zeichen lang sein darf, in 2 Zeilen getrennt geschrieben werden. Dabei werden die ersten 8 Zeichen in die erste Zeile und die nächsten 12 Zeichen in die Fortsetzungszeile geschrieben (siehe auch Abb. 13).

5.3. Eingabebeispiel

In Abb. 8 sind die Jobkontrollkarten und Eingabekarten für ein simples Beispiel wiedergegeben. Mit PARM = 'NETZPLAN=3FACH' wird verlangt, daß die Netzplanzeichnung 3 mal angefertigt wird. Alle unter 5.2. beschriebenen Ausdrücke und die Netzplanzeichnung werden für dieses Beispiel in den Abb. 9 - 14 wiedergegeben.

Man hat die Möglichkeit, in einem Job viele Netzpläne rechnen zu lassen. Dabei werden die Karten aller Modelle untereinander gesteckt. D.h. nach der letzten Tätigkeitskarte des n-ten Modells (mit Endmarke 0; siehe 5.1.2.) muß die Projektkarte des Modells (n+1) folgen.

6. Fehlermeldungen

Bis auf die Fehlermeldungen 6.3.2. und 6.4. führen alle anderen Fehlermeldungen zum Abbruch der Berechnung des jeweiligen Modells.

6.1. Formale Fehler, die während des Einlesens entdeckt und in dem Listing der Eingabekarten (siehe 5.2.1. und Abb. 9) sofort diagnostiziert werden.

I. CØNVERSIOØN ERRØR IN TAETIGKEIT:

TASK (nt) = Tätigkeitsnummer; (nt = lfd.Nr. der Tätigkeitskarten)

N = Anzahl der Folgetätigkeiten auf dieser Karte

FEHLER = 'Z'; DIESES ZEICHEN WIRD DURCH 1 ERSETZT, DIE AUFGABE WIRD ABGEBRØCHEN

Unter CØNVERSIOØN ERRØR mit dem fehlerhaften Zeichen 'Z' fallen folgende Lochungsfehler:

- a) Anstelle einer Ziffer ist irgend ein anderes Zeichen gelocht worden.
- b) Bei der Angabe Zeitdauer ist z.B. anstatt 3.2 gelocht worden: 3,2 (Dezimalkomma ist falsch).
- c) Bei nicht spaltengenauem Ablochen von Zuständigkeitsbereich und Tätigkeitsname, würde vom Programm verlangt werden, Buchstaben oder Sonderzeichen als Zahlen zu interpretieren.

- d) Wenn die letzte Tätigkeit mit der Endmarke 0 nicht auch als letzte Karte im Kartenstapel für ein Netzplanmodell liegt, wird die nächste Karte als Projektkarte für ein weiteres Netzplanmodell verstanden (gleichbedeutend mit dem Vergessen der Endmarke).
- e) Ähnlich wie d), wenn statt Endmarke 0 eine 1 gelocht wurde.
- f) Es sind mehr oder weniger Folgetätigkeiten auf einer Karte genannt als durch N angegeben.

II. INNERER LØØP: FØLGETAETIGKEIT nnn = TAETIGKEIT nnn, wenn irrtümllich als Folgetätigkeit die Tätigkeitsnummer angegeben würde.

III. IN ARBEIT: nnn IST VERKN. mm 2 x VERWENDET.
In der Liste der N Folgetätigkeiten einer Tätigkeitskarte sind eine oder mehrere Folgetätigkeiten doppelt oder noch häufiger genannt.

IV. TAETIGKEITS NR. nn 2 x VERWENDET. Das Programm prüft beim Einlesen jeder Tätigkeitskarte, ob die jeweilige Tätigkeitsnummer von den anderen bisher eingelesenen Tätigkeitsnummern verschieden ist. Es erscheint jedoch keine Fehlermeldung, wenn gleiche Tätigkeitsnamen in verschiedenen Tätigkeitskarten verwendet werden.

V. TAETIGKEIT NR. nn WURDE NICHT ALS FØLGETAETIGKEIT GENANNT.
Eine solche Meldung bedeutet, daß der Netzplan praktisch 2 Anfangstätigkeiten besitzen würde.

VI. TAETIGKEIT NR. nnn HAT KEINE FØLGETAETIGKEIT
ergibt sich für N=0 und ENDMARKE=1. Alle Tätigkeiten außer der Endtätigkeit (siehe 2.2.) müssen Folgetätigkeiten = Verknüpfungen haben.

VII. FØLGETAETIGKEIT: nnn VØN ARBEIT NR.: mmmm IST NICHT AUCH ALS ARBEIT DEFINIERT.(NC=n). Entweder ist die Zahl nnn falsch oder es wurde eine Tätigkeitskarte für die Tätigkeit mit der Nr. nnn vergessen. (NC ist die laufende Nr. aller bis hierher genannten Folgetätigkeiten. NC darf 1200 nicht überschreiten)

6.2. Loop-Fehler

Es tritt ein Kreislauf ein, wenn bei der Verknüpfung der Tätigkeiten irgendwo eine Folgetätigkeit genannt wird, die in diesem Zweig des Netzplanes schon einmal als Folgetätigkeit genannt war.

VIII. LØØP IM NETZWERK MIT FØLGENDEN KNØTEN:

nn1 nn2 nn3 nn4 nn5 nn3

SUCHEN SIE DIE URSACHE DES LØØPS IN DEN VERKNUEPFUNGEN, DIE NACH TAETIGKEIT: nn2 AUFTRETEN KØENNEN.

In jeder der Tätigkeitskarten - hier z.B. von nn2 bis nn5 - kann eine zum Loop führende falsche Folgetätigkeit genannt sein !

6.3. Zeitfehler

6.3.1. Kalenderüberschreitung. Der in das Programm eingebaute Kalender berücksichtigt die Feiertage von 1967 bis 1986. Netzpläne die über den 31.12.1986 hinausgehen, können nicht berechnet werden.

IX.TAETIGKEITEN UEBER 1986

6.3.2. Endterminüberschreitung. Kann der in der Projektkarte (5.1.1.) eingegebene Endtermin nicht eingehalten werden, folgt im Anschluß an den Kalenderausdruck (5.2.2.) die Meldung:

X. ENDTERMIN KANN NICHT EINGEHALTEN WERDEN. BERECHNETER ENDTERMIN IST n. nn. mm. Vor dieser Meldung werden nocheinmal der Projekttitel und die in der Projektkarte angegebenen Daten ausgedruckt.

Im Gegensatz zu allen bisherigen Fehlermeldungen wird die weitere Berechnung des Netzplanes nicht abgebrochen, sondern es wird mit dem berechneten Endtermin weiter gerechnet.

6.4. Fehler bei der Netzplanzeichnung

Wenn in dem zu berechnenden und zu zeichnenden Netzplanmodell keine der Fehler I.bis IX.diagnostiziert wurden und alle DD-Karten gemäß Abb. 8 vorliegen, wird eine Netzplanzeichnung grundsätzlich möglich sein bei der Angabe PARM = 'NETZPLAN=XFACH' auf der EXEC Karte.

Wenn dieser Parameter falsch geschrieben wird oder wenn X nicht innerhalb der Grenzen: $0 < x < 6$ liegt erfolgt die Meldung über FTO6FO01:

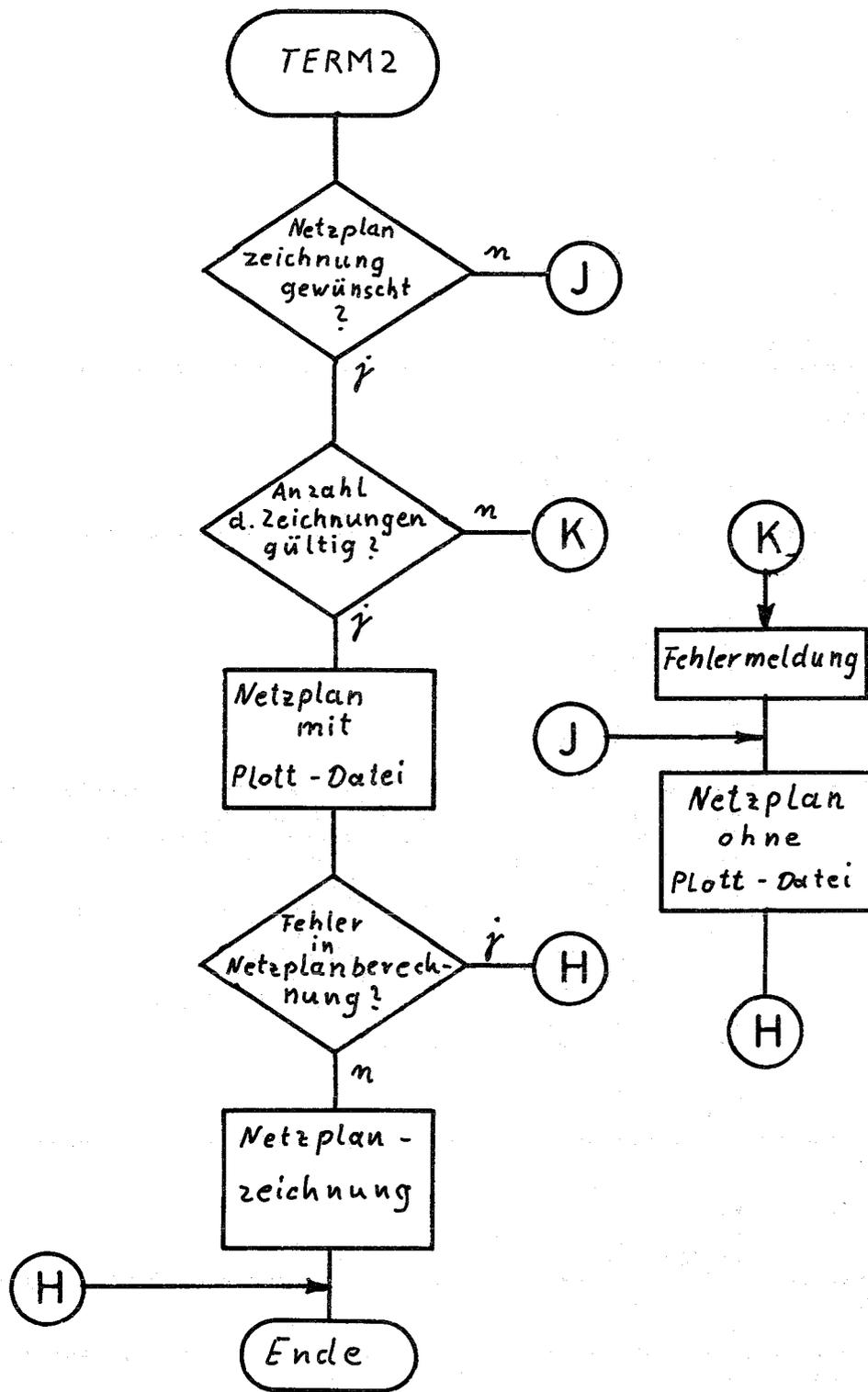
XI. ***PARAMETERANGABE FALSCH, KEINE NETZPLANZEICHNUNG,
PARAMETER MUSS LAUTEN: PARM='NETZPLAN=XFACH' $0 < x < 6$

Die Berechnung des Netzplanes bleibt davon unbeeinflusst.

Literatur

- [1] W. Gmelin
Das Programmsystem TERM-1
KFK-Bericht 462 vom Augsut 1966

- [2] Autorenkollektiv
Handbuch der Netzplantechnik
Verlag: Die Wirtschaft Berlin, 1970



TERM2 Programmablauf

Abb. 1

```

TERM2      CSECT
SAVE      (14,12),,*
BALR      10,0
USING     *,10
B         BEGINN
*
*****
*
REGSAVE   DS      18F
COUNT    DC      F'0'
NULL      DC      H'0'
FUENF     DC      F'5'
ACHT      DC      H'8'
VIERZEHN DC      H'14'
PARM      DC      H'0'
          DC      C'TERMPT '
NETZPLAN  DC      C'NETZPLAN'
*****
*
BEGINN    EQU     *
          LR      3,13
          LA      13,REGSAVE
          ST      13,8(0,3)
          ST      3,4(0,13)
*****
* PARAMETER PRUEFEN
*
          L       3,0(1)
          LH      4,0(3)
          CH      4,NULL
          BE      TERMPPT
          LA      3,2(3)
          CLC    0(8,3),NETZPLAN
          BNE     TERMPTX
*
          CH      4,ACHT          IST PARM=NETZPLAN
          BE      RC              JA , NUR EINE ZEICHNUNG WIRD VERLANGT
          CH      4,VIERZEHN PARM='NETZPLAN=XFACH' = 14 BYTES
          BNE     TERMPTX
          LA      3,9(3)          ADR. DER ANZAHL DER ZEICHNUNGEN
          MVN    COUNT+3(1),0(3)  ZAHL IN COUNT
          L       11,COUNT        ZAHL X IN REG 11
          C      11,FUENF        IST ZAHL X > 5 ?
          BC     2,TERMPTX       JA, DIES IST NICHT ERLAUBT
RA        MVI    PARM+7,X'C3'
          LINK   EP=TERMP,PARAM=(PARM),VL=1
          LR     8,15
          CH     8,ACHT
          BNL    RET
*****
*
          LINK   EP=CALCPL
RET       EQU     *
          L      13,REGSAVE+4
          RETURN (14,12),T,RC=0
*****
RC        LA     11,1(0,0)
          B      RA
*****
*
TERMPTX  OPEN  (PAPIER,(OUTPUT))
*****
          PUT   PAPIER,MESS
*****
          CLOSE PAPIER
*****
*
* PARAMETER FALSCH GESCHRIEBEN ODER MEHR ALS 5 ZEICHNUNGEN VERLANGT
TERMPPT  LINK   EP=TERMP,PARAM=(PARM),VL=1
          B      RET
*****
*
PAPIER   DCB    DDNAME=FT06F001,MACRF=(PM),DSORG=PS,BLKSIZE=931,
          LRECL=133,RECFM=FBA,EROPT=ACC
*****
MESS     DC     CL132' *** PARAMETERANGABE FALSCH, KEINE NETZPLAN-ZEICHNUNG,
          HNUMG, PARAMETER MUSS LAUTEN: PARM='NETZPLAN=XFACH'
          C<X<6
          END

```

```

TERM : PROCEDURE (INPARM) OPTIONS (MAIN);                                00000100
DECLARE INPARM CHAR(4),                                                00000200
  1 VALUES BASED (P),                                                00000300
  2   PARAM CHAR (8) ;                                               00000400
  P=ADDR(INPARM);                                                    00000500
DCL REND LABEL;                                                       00000600
REND=TERMPT;                                                           00000700
IF PARAM='TERMP' THEN REND=TERMP;                                     00000800
PUT FILE (SYSPRINT) EDIT ('PARAMETER=', PARAM)(PAGE,A,A);          00000900
DCL                                                                      00001000
(FI_W,F_C_W) BIT(8) STATIC,                                           00001100
/* G_DEF = GRAPH - DEFINITION                                         00001200
  TI = GRAPH - TITLE           ENTSpricht TITLE                       00001300
  NE = GRAPH - NUMBER OF EVENTS ENTSpricht NE                         00001400
  NT = GRAPH - NUMBER OF TASKS ENTSpricht NT */                       00001500
1 G_DEF STATIC,                                                       00001600
  2 G_TI CHAR(52),                                                    00001700
  2 G_NE BIN FIXED (17,0),                                           00001800
  2 G_NT BIN FIXED (17,0),                                           00001900
1 COOR STATIC,                                                       00002000
/* KOORDINATEN FUER PLOT DATA-SET 'GRAPH' : X1 ENTSpricht DER ZEIT  00002100
                                     X2 ENTSpricht DER Y-AXE */      00002200
  2 X1 BIN FLOAT,                                                    00002300
  2 X2 BIN FLOAT,                                                    00002400
1 G_STR STATIC,                                                       00002500
/* STRUKTUR : A_N1 BOGEN-ANFANG                                       00002600
  A_N2 BOGEN-ENDE                                                    00002700
  A_DSC BOGEN-NR = NUMERISCHER NAME DER TASK ENTSpricht. TASK      00002800
  A_ABT FIRMA, DIE DIE TASK AUSFUEHRT ENTSpricht. DEPT             00002900
  A_DES TASK, ARBEITSBESCHREIBUNG (ALPHANUMERISCH)=DESCR         00003000
  A_DAU DAUER DER ARBEIT ENTSpricht DUR                             00003100
  A_TYPE BOGEN-TYP */                                               00003200
  2 A_N1 BIN FIXED (17,0),                                           00003300
  2 A_N2 BIN FIXED (17,0),                                           00003400
  2 A_DSC BIN FIXED(17,0),                                           00003500
  2 A_ABT CHAR(5),                                                   00003600
  2 A_DES CHAR(20),                                                  00003700
  2 A_DAU BIN FLOAT,                                                 00003800
  2 A_TYPE BIN FIXED(15,0),                                          00003900
  2 A_SBEGIN(3) BIN FIXED(15,0),                                     00004000
D_N (0:6) CHAR(10) STATIC INITIAL ('SONNTAG ', 'MONTAG ',          00004100
'DIENSTAG ', 'MITTWOCH ', 'DONNERSTAG', 'FREITAG ', 'SAMSTAG '), 00004200
/* N_WAY ANZHL DER WEGE, AMPL=AMPLITUDE, WAY_TB=TABELLE DER WEGE    00004300
  W_ORD= WEGE-ORDNUNG =SORTIER-VEKTOR, SW ENTSpricht IMMER 'SWITCH',*/ 00004400
N_WAY BIN FIXED(17,0) STATIC,                                         00004500
AMPL(1000) BIN FLOAT STATIC,                                          00004600
WAY_TB(1000,2) BIN FIXED (17,0) STATIC,                               00004700
W_ORD (1000) BIN FIXED (17,0) STATIC,                                 00004800
PLT_SW (1000) BIT(8) STATIC,                                          00004900
GRAPH FILE RECORD OUTPUT SEQUENTIAL ENVIRONMENT(V(6404,64),        00005000
  CONSECUTIVE);                                                       00005100
DECLARE                                                                00005200
TITLE CHAR(50)STATIC,                                                00005300
DATE(3,3) DEC FIXED (3) STATIC, PR_SW BIT(8), W_DAY BIT (8),       00005400
TASK(1000)DEC FIXED(5) STATIC,                                       00005500
DEPT (1000) CHAR ( 5) STATIC,                                         00005600
O_DEPT CHAR(5) STATIC,                                               00005700
DESCR (1000) CHAR (20) STATIC,                                       00005800

```

Programm für die Netzplanberechnung Abb.3(1)

```

DUR          (1000) DEC FIXED(5,1) STATIC,          00005900
W(O:1) CHAR(4) STATIC INITIAL('FREI','  '),          00006000
IW          BIN FIXED (17,0) STATIC,                00006100
I2 DEC FIXED(7) STATIC,                             00006200
I1          BIN FIXED (17,0) STATIC,                00006300
           /*LATER FIXED (16,0) */                 00006400
FEHLER CHAR(1) STATIC,                              00006500
ERRCNT      BIN FIXED (17,0) STATIC EXTERNAL,       00006600
           (CNT,I,J,K,NC,NT,N,F_T_X,L_T_X,NRT,ITER,T_X,EVENT,NE) 00006700
           BIN FIXED (17,0) STATIC,                 00006800
           /* NC ANZAHL DER VERKNUEPFUNGEN T_X TASK-INDEX */ 00006900
D_CONS      (1200) DEC FIXED (5) STATIC,            00007000
           /* NT TASKS OF INDEXES 1 TO NT .          00007100
           THE INDEXES OF THE SUCCESSORS OF TASK INDEX T_X ARE STORED 00007200
           IN CONSTR(J,1) FOR J= X_CONS(T_X,1,1) TO X_CONS(T_X,2,1). 00007300
           THE INDEXES OF THE ANCESTORS OF TASK INDEX T_X ARE STORED 00007400
           IN CONSTR(J,2) FOR J=X_CONS(T_X,1,2) TO X_CONS(T_X,2,2). */ 00007500
X_CONS      (1200,2,2)BIN FIXED(17,0)STATIC,       00007600
CONSTR      (1200,2) BIN FIXED(17,0) STATIC,        00007700
TLIST(50) DEC FIXED(7)STATIC,                       00007800
FLAG DEC FIXED(1) STATIC,                           00007900
STACK      (1001)  BIN FIXED (17,0) STATIC,         00008000
ST_DPT     BIN FIXED (17,0)  STATIC,                00008100
LEVEL      (1000)  BIN FIXED (17,0)  STATIC,        00008200
T_ORD      (1000)  BIN FIXED (17,0)  STATIC,        00008300
T_ORD1(1000)  BIN FIXED (17,0)  STATIC,             00008400
(I1,JJ,KK,ICODE) BIN FIXED (17,0)  STATIC,         00008500
(I11,I12,I21,I22,I22T,J1,T_X1)
           BIN FIXED (17,0)  STATIC,                 00008600
PAGE       DEC FIXED (3)  STATIC,                   00008700
           /* CAL= KALENDER, M_LEN MONATS-LAENGE,N_DAY==ANZAHL DER TAGE,L_MN= 00008800
           MONATS-LAENGE,TDAY=HEUTIGES DATUM ZB. 12. 1. 71 = 3 WERTE,*/ 00008900
CAL( 0:1999 ,3) DEC FIXED (3,0)  STATIC,            00009000
M_LEN      (12,2)  DEC FIXED (3,0)  INITIAL((2) 31,28,29,(2) 31,(2) 30, 00009100
           (2) 31,(2) 30,(4) 31,(2) 30,(2) 31,(2) 30,(2) 31 ) STATIC, 00009200
(N_DAY,L_MN )  BIN FIXED (17,0)  STATIC,            00009300
TDAY(3) DEC FIXED (3,0)  STATIC,                   00009400
DOW        BIN FIXED (17,0)  STATIC,                00009500
           /* M_DAY = MOBILE FEIERTAGE HIER OSTERMONTAG( AUF KARFREITAG WIRD ZU- 00009600
           RUECKGERECHNET), CHRISTI-HIMMELFAHRT, PFINGSTMONTAG BIS 1986 00009700
           WAY_LE, WAY_RK = ANZAHL DER BOGEN IN EINEM WEG */ 00009800
M_DAY( 19 ,6)  FIXED DEC ( 3,0 )  STATIC INITIAL ( 00009900
15,4,23,5,3,6,7,4,15,5,26,5,30,3,7,5,18,5,12,4,20,5,31,5,3,4,11,5,22,5, 00010000
23,4,31,5,11,6,15,4,23,5,3,6,31,3,8,5,19,5,19,4,17,5,7,6,11,4,19,5,30,5, 00010100
27,3,4,5,15,5,16,4,24,5,4,6,7,4,15,5,26,5,20,4,28,5,8,6,12,4,20,5,31,5,4 00010200
,4,12,5,23,5,23,4,31,5,11,6,8,4,16,5,27,5,31,3,8,5,19,5), 00010300
P_TIME(3)  DEC FIXED(7,3)  STATIC,                 00010400
(P_DA1,P_DA2,P_DA3,E_DA)(3) DEC FIXED(3)  STATIC,  00010500
(WAY_LE,WAY_RK) (1200)  DEC FIXED (3,0)  STATIC ,  00010600
EV_X(1000)  BIN FLOAT  STATIC,                     00010700
EV_Y(1000)  BIN FLOAT  STATIC,                     00010800
EV_ARC(1200,2)  BIN FIXED(17,0)  STATIC,           00010900
TEST_X     BIN FLOAT  STATIC,                       00011000
           /* SLACK= FREIE PUFFERZEIT, T_SLK= TOTALE PUFFERZEIT, 00011100
           D_SLK= T_SLK/WAY_LE = VERTEILTE PUFFERZEIT */ 00011200
(SLACK,T_SLK,D_SLK) (1200) DEC FIXED (7,3)  STATIC, 00011300
TIME(1000,2,2) DEC FIXED (7,3)  STATIC,           00011400
(N_T,END_T,T1)  DEC FIXED(7,3)  STATIC,            00011500
           00011600

```

Abb.3(2)

```

(I_SW,M_SW) BIT(8) STATIC,                                00011700
ARC_M (1200) BIT(8),                                     00011800
PLOT_SW BIT(1),                                          00011900
TEST_DEC FIXED(5)STATIC;                                00012000
ON ENDFILE(SYSIN) GO TO FIN;                             00012100
ON CONVERSION BEGIN;                                    00012200
PUT FILE(SYSPRINT) SKIP      LIST('CONVERSION ERROR IN TAETIGKEIT :'); 00012300
ICODE=ONCODE;                                           00012400
PUT FILE(SYSPRINT) SKIP DATA (TASK(NT),N);              00012500
FEHLER=ONCHAR;                                          00012600
PUT FILE(SYSPRINT) SKIP DATA (FEHLER);                  00012700
PUT FILE(SYSPRINT) EDIT ('DIESES ZEICHEN WIRD DURCH 1 ERSETZT, DIE AUFGA00012800
BE WIRD ABGEBROCHEN')(COLUMN(15),A);                    00012900
ONCHAR='1';                                             00013000
ERRCNT=ERRCNT+1B;                                       00013100
IF ERRCNT > 200 THEN GO TO FIN;                          00013200
END;                                                     00013300
      /* INITIALIZATION.*/                                00013400
PLOT_SW = '1'B;                                         00013500
NEXT:                                                    00013600
PUT FILE(SYSPRINT) PAGE;                                 00013700
PUT FILE(SYSPRINT) EDIT ('TERM2','EINGABE:')(COLUMN(40),A,SKIP(3),A, 00013800
SKIP(2));                                                00013900
ERRCNT,NC,ST_DPT=0B;                                    00014000
      /* INPUT READING.                                    00014100
      DATE(1,*)=START                                     00014200
      DATE(2,*)=ENDE                                     00014300
      DATE(3,*)=REPORT.      */                          00014400
GET FILE(SYSIN) EDIT (TITLE,DATE) (X(1),A(50),X(1),9 F(3)); 00014500
PUT FILE(SYSPRINT) SKIP EDIT (TITLE,DATE,                'TAETIGKEITEN:') 00014600
(A,3 (3 F(3),X(2)),SKIP,      SKIP(5),A);               00014700
FLAG=1;                                                  00014800
DO NT=1B BY 1B WHILE(FLAG=1);                            00014900
GET FILE(SYSIN) EDIT (TASK(NT),DEPT(NT),DESCR(NT),DUR(NT)) 00015000
(SKIP,F(4),X(2),A(5),A(20),X(1),F(7,1));                00015100
PUT FILE(SYSPRINT) EDIT                                  00015200
(TASK(NT),DEPT(NT),DESCR(NT),DUR(NT))                   00015300
(SKIP(2),F(5),X(3),A,X(1),A,F(5,1));                    00015400
GET FILE(SYSIN) LIST (N,(TLIST(J) DO J=1B TO N),FLAG);   00015500
PUT FILE(SYSPRINT) EDIT                                  00015600
(N,(TLIST(J) DO J=1B TO N))(SKIP,F(5),10 F(8));          00015700
      /* INPUT ACQUISITION.                                00015800
      THE CONSTRAINTS ARE STORED AS FOLLOWS:             00015900
      THE TASKS WHICH CANNOT BE INITIATED BEFORE THE COMPLETION 00016000
      OF TASK(I) ARE STORED IN D_CONS FROM INDEX X_CONS(I,1) 00016100
      TO INDEX X_CONS(I,2).*/                             00016200
X_CONS(NT,1B,1B)=NC+1B; X_CONS(NT,10B,1B)=NC+N;        00016300
      /* ERROR TEST NR.0 : ARE THERE SUCCESSORS FOR EACH TASK EXEPT 00016400
      FOR THE LAST ONE, AND ARE THE SUCCESSORS ALL DIFFERENT FROM THE TASK?*/00016500
IF N=0 & FLAG=1 THEN DO ;                                00016600
ERRCNT=ERRCNT+1B;                                        00016700
      PUT FILE(SYSPRINT)EDIT ('TAETIGKEIT NR.',TASK(NT), 00016800
      ' HAT KEINE FOLGETAETIGKEIT. ')                   00016900
(SKIP,COLUMN(75),A,F(6),A);                              00017000
END;                                                       00017100
DO J=1B TO N;                                           00017200
NC=NC+1;                                                 00017300
D_CONS(NC)=TLIST(J);                                    00017400

```

Abb.3(3)

```

IF TASK(NT) = TLIST(J) THEN DO;                                00017500
ERRCNT = ERRCNT + 1B;                                         00017600
PUT FILE(SYSPRINT) EDIT ('INNERER LOOP : FOLGETAETIGKEIT ',TLIST(J), 00017700
' = TAETIGKEIT ',TASK(NT))(SKIP,COLUMN(75),A,F(4),A,F(4));    00017800
END;                                                            00017900
/* ERROR TEST NR 1:CHECK WHETHER THE ELEMENTS OF THE LIST VECTOR 00018000
ARE ALL DIFFERENT. */                                         00018100
DO K=1B TO J-1B;                                             00018200
IF TLIST(K) = TLIST(J) THEN DO;                               00018300
ERRCNT=ERRCNT+1B;                                             00018400
PUT FILE(SYSPRINT) EDIT ('IN ARBEIT:',TASK(NT),' IST VERKN.',TLIST(K), 00018500
' 2X VERWENDET.')(SKIP(3),COLUMN(80),A, F(6),A,F(6),A);      00018600
END;END;                                                       00018700
END;                                                            00018800
/* ERROR TEST NR 2:TEST WHETHER THE ELEMENTS OF THE TASK VECTOR 00018900
ARE ALL DIFFERENT. */                                         00019000
DO J=1B TO NT - 1B;                                          00019100
IF TASK(J) = TASK (NT) THEN GO TO ER1;                        00019200
END;                                                            00019300
GO TO NER1;                                                    00019400
ER1:                                                            00019500
PUT FILE(SYSPRINT) EDIT ('TAETIGKEIT NR',TASK(NT),' 2X VERWENDET.')( 00019600
(SKIP(3),COLUMN(85),A,F(6),A);                                00019700
ERRCNT=ERRCNT+1B;                                             00019800
NER1:                                                            00019900
END;                                                            00020000
/* PROCESSING. */                                             00020100
PR:                                                            00020200
NT=NT-1B;                                                      00020300
L_T_X=NT;                                                       00020400
/* NT:NUMBER OF TASKS OR NODES,                                00020500
NC:NUMBER OF CONSTRAINTS OR ARCS,                             00020600
BOTH ON THE GRAPH OF TASKS AND CONSTRAINTS.                  00020700
ERROR TEST NR.3: CHECK WHETHER ONLY ONE ELEMENT OF TASK      00020800
I.E. THE FIRST TASK DOES NOT BELONG TO D_CONS.*/             00020900
CNT=0B;                                                         00021000
DO I=1B TO NT;                                                 00021100
TEST=TASK(I);                                                  00021200
DO J=1B TO NC;                                                 00021300
IF D_CONS(J)=TEST THEN GO TO E_SK_1;                          00021400
END;                                                            00021500
/* NOT FOUND. */                                              00021600
CNT=CNT+1B; ST_DPT=ST_DPT+1B; STACK(ST_DPT)=I;              00021700
E_SK_1:                                                         00021800
END;                                                            00021900
/* CONCLUSION. */                                             00022000
IF CNT= 1B THEN DO;                                           00022100
F_T_X =STACK(ST_DPT);                                          00022200
PUT FILE(SYSPRINT) EDIT                                       00022300
('ANFANG TAETIGKEIT:',TASK(F_T_X))(SKIP(4),A,F(6));          00022400
ST_DPT=ST_DPT-1B;                                             00022500
END;                                                            00022600
ELSE DO;                                                        00022700
PUT FILE(SYSPRINT) EDIT ('FATAL ERROR.')(                     00022800
(SKIP,COLUMN(85),A);                                          00022900
DO I=1B TO CNT;                                               00023000
PUT FILE(SYSPRINT) EDIT ('TAETIGKEIT NR.',                   00023100
TASK(STACK(ST_DPT)),                                         00023200

```

Abb3(4)

```

          ' WURDE NICHT ALS FOLGETAETIGKEIT GENANNT.' )
          ( SKIP,COLUMN(65),A,F(6),A);
          ST_DPT=ST_DPT-1B;
          END;
ERRCNT=ERRCNT+1B;
          END;
          /* ERROR TEST NR.4: CHECK WHETHER ALL THE ELEMENTS OF D_CONS
          BELONG TO TASK. */
CNT=0B;
DO I=1B TO NC;
TEST=D_CONS(I);
          DO J=1B TO NT;
          IF TASK(J)=TEST THEN GO TO C_FND2 ;
          /*
          FOR CORRECT INPUT DATA,THE END IS : TASK(NT) */
          END;
          /* NOT FOUND: */
          CONSTR(I,1B)=NT+1B;
          DO J=1B TO I-1B;
          IF TEST=D_CONS(J) THEN GO TO E_SK_2;
          END;
NC=NC+1B;
          /* THIS TASK DOES NOT BELONG TO THE TASK VECTOR AND WAS NOT YET
          TAKEN FROM THE D_CONS VECTOR. */
          CNT=CNT + 1B;
          ST_DPT=ST_DPT+1B; STACK(ST_DPT)=I;
          GO TO E_SK_2;
          /* FOUND */
          C_FND2:
          CONSTR(I,1B)=J;
E_SK_2:
END;
          /* CONCLUSION. */
IF CNT = 0B THEN DO;
          PUT FILE(SYSPRINT) EDIT
          ('ENDE TAETIGKEIT: ',TASK(L_T_X))(SKIP,A,F(6));
          END;
          ELSE DO;
          PUT FILE(SYSPRINT)EDIT ('FATAL ERROR.')
          (SKIP,COLUMN(100),A);
          DO I=1B TO CNT;
          DO J = 1B TO NC;
          K=J;
          IF X_CONS(J,1B,1B) <= STACK(ST_DPT) & X_CONS(J,10B,1B) >= STACK(ST_DPT)
          THEN GO TO MESS;
          END;
          MESS:
          PUT FILE(SYSPRINT) EDIT ('FOLGETAETIGKEIT:',D_CONS(STACK(ST_DPT)),
          ' VON ARBEIT NR.:',TASK(K),' IST NICHT AUCH ALS ARBEIT DEFINIERT.(NC=',
          STACK(ST_DPT),')')
          (SKIP,COLUMN(40),A,F(4),A,F(4),A,F(4),A);
          ST_DPT=ST_DPT-1B;
          END;
          ERRCNT=ERRCNT+1B;
          END;
IF ERRCNT > 0B THEN DO;
          PUT FILE(SYSPRINT) EDIT ('AUFGABE ABGEBROCHEN.')
          (SKIP,COLUMN(100),A);
GET FILE (SYSIN) SKIP; GO TO NEXT;

```

```

00023300
00023400
00023500
00023600
00023700
00023800
00023900
00024000
00024100
00024200
00024300
00024400
00024500
00024600
00024700
00024800
00024900
00025000
00025100
00025200
00025300
00025400
00025500
00025600
00025700
00025800
00025900
00026000
00026100
00026200
00026300
00026400
00026500
00026600
00026700
00026800
00026900
00027000
00027100
00027200
00027300
00027400
00027500
00027600
00027700
00027800
00027900
00028000
00028100
00028200
00028300
00028400
00028500
00028600
00028700
00028800
00028900
00029000

```

Abb.3 (5)

```

END;
/* CORRECT X_CONS FOR THE LAST TASK. */
X_CONS(L_T_X,1B,1B)=NC;
X_CONS(L_T_X,10B,1B)=NC-1B;
PUT FILE(SYSPRINT) EDIT (NT,' TAETIGKEITEN ODER KNOTEN UND' )
(SKIP(3),F(6),A);
PUT FILE(SYSPRINT) EDIT( NC,' VERKNUEPFUNGEN ODER KANTEN ' )
(SKIP,F(6),A);
PUT FILE(SYSPRINT) EDIT(' IN EINGABE GRAPH.')(SKIP,COLUMN(24),A);
PUT FILE(SYSPRINT) PAGE;
NRT=NT;
/* ERROR TEST NR.5: CHECK WHETHER THE GRAPH TASK(NODES)
AND CONSTRAINTS ( ARCS) IS CONNECTED AND HAS
NO CIRCUITS. */
/* INITIALIZATION OF THE ARC MARKS. */
DO I=1B TO NC; ARC_M(I) ='100C0000'B;END;
/* INITIALIZE STACK DEPTH */
ST_DPT=0B;
/* SELECT THE FIRST NODE */
I=F_T_X;
G_A11:
/* STORE IN STACK AND STORE STACK DEPTH AS LEVEL FOR THE NODE.*/
ST_DPT=ST_DPT+1B; STACK(ST_DPT)=I; LEVEL(I)=ST_DPT;
/* DOES TASK(I) HAVE A SUCCESSOR THRU A NOT YET
MARKED ARC. */
G_A12:
DO J=X_CONS(I,1B,1B) TO X_CONS(I,10B,1B);
IF ARC_M(J)THEN DO;
/* FOUND: MARK THE ARC AND SELECT THE SUCCESSOR*/
ARC_M(J)='00000000'B;
I= CONSTR(J,1B);
/* CIRCUIT TEST */
DO K=10B TO ST_DPT-1B;
IF I=STACK(K) THEN DO;
ERRCNT=ERRCNT+1B;
PUT FILE(SYSPRINT) EDIT (' LOOP IM NETZWERK MIT FOLGENDEN KNOTEN: ' )
(SKIP(2),COLUMN(85),A);
PUT FILE(SYSPRINT) SKIP(2);
DO J =1B TO ST_DPT;
PUT FILE(SYSPRINT) EDIT (TASK(STACK(J)))(F(8));
END;
PUT FILE(SYSPRINT) EDIT (TASK(I),' SUCHEN SIE DIE URSACHE DES LOOPS
IN DEN VERKNUEPFUNGEN, DIE NACH TAETIGKEIT :',TASK(STACK(K-1B)),
'AUF TRETEN KOENNEN')
(F(10),SKIP,A,F(5),A);
I=STACK(ST_DPT); GO TO G_A12;
END;END;
GOTO G_A11;
END;
/* THE ARC WAS ALREADY MARKED */
END;
/* CONNECTIVITY CHECK */
CO_CK:
IF ST_DPT > 1B THEN DO;
/* SEEK IN THE STACK */
ST_DPT=ST_DPT-1B; I=STACK(ST_DPT);
GO TO G_A12;
END;

```

Abb.3 (6)

```

IF ERRCNT > OB THEN DO; GET FILE (SYSIN) SKIP; GO TO NEXT; END;
/* TASK ORDERING. */
DO I=1B TO NT; T_ORD(I) = I; END;
/* BUBBLE SORT. */
DO I=1B TO NT-1B; /*1*/
    DO K=NT-1B BY -1B TO I; /*2*/
        IF LEVEL(T_ORD(K)) < LEVEL(T_ORD(K+1)) THEN GO TO E_B1;
        IF LEVEL(T_ORD(K)) = LEVEL(T_ORD(K+1)) THEN DO; /*3*/
            IF DUR(T_ORD(K)) >= DUR(T_ORD(K+1B)) THEN
                GO TO E_B1;
            END; /*3*/
        /*EXCHANGE.*/
        JJ=T_ORD(K); T_ORD(K)=T_ORD(K+1B); T_ORD(K+1B)= JJ;
        E_B1:
        END; /*2*/
    END; /*1*/
/* ANCESTOR VECTOR */
II=OB; X_CONS(1B,1B,10B)=1B;
DO T_X=1B TO NT; /*1*/
    IF T_X > 1B THEN X_CONS(T_X,1B,10B)=X_CONS(T_X-1B,10B,10B)+1B;
X_CONS(T_X,10B,10B)=X_CONS(T_X,1B,10B)-1B;
    DO J=1B TO NC; /*2*/
        IF CONSTR(J,1B)=T_X THEN DO; /*3*/
            DO K=1B TO NT; /*4*/
                IF J >= X_CONS(K,1B,1B) THEN DO; /*5*/
                    IF J <= X_CONS(K,10B,1B) THEN GO TO A_S;
                END; /*5*/
            END; /*4*/
            A_S:
            II=II+1B;
            CONSTR(II,10B)=K;
            X_CONS(T_X,10B,10B)=II;
            END; /*3*/
        END; /*2*/
    END; /*1*/
/* SLACK=F.PFZ, T_SLK =T.PFZ, D_SLK =VF.PFZ,
F.BEG=T11, TS.END=T22 FS.END=
VF.PFZ=T.PFZ/ N, N= NUMBER OF ARCS IN A WAY..... */
/* INITIALIZATION. */
DO T_X=1B TO NT; /*1*/
TIME(T_X,1B,1B)=0;
TIME(T_X,10B,1B)=DUR(T_X);
END; /*1*/
/* FORWARD EVENT SCAN. */
I_SW='10000000'B;
/* FORD. */
DO ITER=1B BY 1B WHILE(I_SW); /*1*/
I_SW='00000000'B;
/* TAKE THE CONSTRAINTS INTO ACCOUNT. */
DO I=1B TO NT; /*2*/
T_X=T_ORD(I);
T1=TIME(T_X,1B,1B);
M_SW='00000000'B;
    DO K= X_CONS(T_X,1B,10B) TO X_CONS(T_X,10B,10B); /*3*/
J=CONSTR(K,10B);
/* J IS AN ANCESTOR. */
N_T=TIME(J,10B,1B);
IF N_T > T1 THEN DO; /*4*/

```

```

00034900
00035000
00035100
00035200
00035300
00035400
00035500
00035600
00035700
00035800
00035900
00036000
00036100
00036200
00036300
00036400
00036500
00036600
00036700
00036800
00036900
00037000
00037100
00037200
00037300
00037400
00037500
00037600
00037700
00037800
00037900
00038000
00038100
00038200
00038300
00038400
00038500
00038600
00038700
00038800
00038900
00039000
00039100
00039200
00039300
00039400
00039500
00039600
00039700
00039800
00039900
00040000
00040100
00040200
00040300
00040400
00040500
00040600

```

Abb.3(7)

```

M_SW='10000000'B;
T1=N_T;
END; /*4*/
END; /*3*/
IF M_SW THEN DO; /*3-1*/
I_SW='10000000'B;
TIME(T_X, 1B, 1B)=T1;
TIME(T_X,10B, 1B)=T1 + DUR(T_X);
END; /*3-1*/
END; /*2*/
END; /*1*/
PUT FILE (SYSPRINT) EDIT
('NUMBER OF ITERATIONS:',ITER-1B) (SKIP,A,F(10));
/* INTERMEDIATE SORT. (ON FINAL TIMES.) */
DO I=1B TO NT-1B; /*1*/
DO K=NT-1B BY -1B TO I; /*2*/
IF TIME(T_ORD(K ),10B,1B ) >
TIME(T_ORD(K+1B),10B,1B ) THEN DO; /*3*/
/* EXCHANGE. */
JJ=T_ORD(K); T_ORD(K)=T_ORD(K+1B); T_ORD(K+1B)= JJ;
END; /*3*/
END; /*2*/
END; /*1*/
/* INITIAL TIME SORT INITIALIZATION. */
DO I=1B TO NT; T_ORD1(I)=I; END;
/* SORT ON INITIAL TIMES. */
DO I=1B TO NT-1B; /*1*/
DO K=NT-1B BY -1B TO I; /*2*/
IF TIME(T_ORD1(K ),1B,1B) >
TIME(T_ORD1(K+1B ),1B,1B) THEN DO; /*3*/
/* EXCHANGE. */
JJ=T_ORD1(K); T_ORD1(K)=T_ORD1(K+1B); T_ORD1(K+1B)= JJ;
END; /*3*/
END; /*2*/
END; /*1*/
/* REVERSE EVENT SCAN. */
END_T=TIME(T_ORD(NT),10B,1B);
/* INITIALIZATION. */
DO T_X=1B TO NT; /*1*/
TIME(T_X, 1B,10B)=END_T - DUR(T_X);
TIME(T_X,10B,10B)=END_T;
END; /*1*/
I_SW='10000000'B;
/* FORD. */
DO ITER=1B BY 1B WHILE(I_SW); /*1*/
I_SW='00000000'B;
/* TAKE THE CONSTRAINTS INTO ACCOUNT. */
DO I=NT BY -1B TO 1B; /*2*/
T_X=T_ORD(I);
T1=TIME(T_X,10B,10B);
M_SW='00000000'B;
DO K= X_CONS(T_X,1B,1B) TO X_CONS(T_X,10B,1B); /*3*/
J=CONSTR(K, 1B);
/* J IS A SUCCESSOR. */
N_T=TIME(J, 1B,10B);
IF N_T < T1 THEN DO; /*4*/
M_SW='10000000'B;
T1=N_T;

```

```

00040700
00040800
00040900
00041000
00041100
00041200
00041300
00041400
00041500
00041600
00041700
00041800
00041900
00042000
00042100
00042200
00042300
00042400
00042500
00042600
00042700
00042800
00042900
00043000
00043100
00043200
00043300
00043400
00043500
00043600
00043700
00043800
00043900
00044000
00044100
00044200
00044300
00044400
00044500
00044600
00044700
00044800
00044900
00045000
00045100
00045200
00045300
00045400
00045500
00045600
00045700
00045800
00045900
00046000
00046100
00046200
00046300
00046400

```

Abb. 3(8)

```

                                END; /*4*/
                                00046500
                                00046600
                                00046700
                                00046800
                                00046900
                                00047000
                                00047100
                                00047200
                                00047300
                                00047400
                                00047500
                                00047600
                                00047700
                                00047800
                                00047900
                                00048000
                                00048100
                                00048200
                                00048300
                                00048400
                                00048500
                                00048600
                                00048700
                                00048800
                                00048900
                                00049000
                                00049100
                                00049200
                                00049300
                                00049400
                                00049500
                                00049600
                                00049700
                                00049800
                                00049900
                                00050000
                                00050100
                                00050200
                                00050300
                                00050400
                                00050500
                                00050600
                                00050700
                                00050800
                                00050900
                                00051000
                                00051100
                                00051200
                                00051300
                                00051400
                                00051500
                                00051600
                                00051700
                                00051800
                                00051900
                                00052000
                                00052100
                                00052200

                                END; /*3*/
                                00046600
                                00046700
                                00046800
                                00046900
                                00047000
                                00047100
                                00047200
                                00047300
                                00047400
                                00047500
                                00047600
                                00047700
                                00047800
                                00047900
                                00048000
                                00048100
                                00048200
                                00048300
                                00048400
                                00048500
                                00048600
                                00048700
                                00048800
                                00048900
                                00049000
                                00049100
                                00049200
                                00049300
                                00049400
                                00049500
                                00049600
                                00049700
                                00049800
                                00049900
                                00050000
                                00050100
                                00050200
                                00050300
                                00050400
                                00050500
                                00050600
                                00050700
                                00050800
                                00050900
                                00051000
                                00051100
                                00051200
                                00051300
                                00051400
                                00051500
                                00051600
                                00051700
                                00051800
                                00051900
                                00052000
                                00052100
                                00052200

                                END; /*2*/
                                00047200
                                00047300
                                00047400
                                00047500
                                00047600
                                00047700
                                00047800
                                00047900
                                00048000
                                00048100
                                00048200
                                00048300
                                00048400
                                00048500
                                00048600
                                00048700
                                00048800
                                00048900
                                00049000
                                00049100
                                00049200
                                00049300
                                00049400
                                00049500
                                00049600
                                00049700
                                00049800
                                00049900
                                00050000
                                00050100
                                00050200
                                00050300
                                00050400
                                00050500
                                00050600
                                00050700
                                00050800
                                00050900
                                00051000
                                00051100
                                00051200
                                00051300
                                00051400
                                00051500
                                00051600
                                00051700
                                00051800
                                00051900
                                00052000
                                00052100
                                00052200

                                END; /*1*/
                                00047300
                                00047400
                                00047500
                                00047600
                                00047700
                                00047800
                                00047900
                                00048000
                                00048100
                                00048200
                                00048300
                                00048400
                                00048500
                                00048600
                                00048700
                                00048800
                                00048900
                                00049000
                                00049100
                                00049200
                                00049300
                                00049400
                                00049500
                                00049600
                                00049700
                                00049800
                                00049900
                                00050000
                                00050100
                                00050200
                                00050300
                                00050400
                                00050500
                                00050600
                                00050700
                                00050800
                                00050900
                                00051000
                                00051100
                                00051200
                                00051300
                                00051400
                                00051500
                                00051600
                                00051700
                                00051800
                                00051900
                                00052000
                                00052100
                                00052200

                                /* SLACK CALCULATION. */
                                00047500
                                00047600
                                00047700
                                00047800
                                00047900
                                00048000
                                00048100
                                00048200
                                00048300
                                00048400
                                00048500
                                00048600
                                00048700
                                00048800
                                00048900
                                00049000
                                00049100
                                00049200
                                00049300
                                00049400
                                00049500
                                00049600
                                00049700
                                00049800
                                00049900
                                00050000
                                00050100
                                00050200
                                00050300
                                00050400
                                00050500
                                00050600
                                00050700
                                00050800
                                00050900
                                00051000
                                00051100
                                00051200
                                00051300
                                00051400
                                00051500
                                00051600
                                00051700
                                00051800
                                00051900
                                00052000
                                00052100
                                00052200

                                DO T_X=1B TO NT; /*1*/
                                T_SLK(T_X)=TIME(T_X,1B,10B) - TIME(T_X,1B,1B);
                                SLACK(T_X)=END_T;
                                DO K= X_CONS(T_X,1B,1B) TO X_CONS(T_X,10B,1B); /*2*/
                                J=CONSTR(K,1B);
                                /* J IS A SUCCESSOR. */
                                T1=TIME(J,1B,1B) - TIME(T_X,10B,1B);
                                IF SLACK(T_X) > T1 THEN SLACK(T_X)=T1;
                                END; /*2*/
                                END; /*1*/
                                SLACK(L_T_X)=0;
                                PUT FILE (SYSPRINT) EDIT
                                ('NUMBER OF ITERATIONS:',ITER-1B) (SKIP,A,F(10));
                                DO I=1B TO NT-1B; /*1*/
                                DO K=NT-1B BY -1B TO I; /*2*/
                                IF T_SLK(T_ORD(K)) >
                                T_SLK(T_ORD(K+1B) ) THEN DO; /*3*/
                                /* EXCHANGE. */
                                JJ=T_ORD(K); T_ORD(K)=T_ORD(K+1B); T_ORD(K+1B)= JJ;
                                END; /*3*/
                                END; /*2*/
                                END; /*1*/
                                /* INITIALIZATION OF THE WAY TABLES. */
                                DO I=1B TO NT; WAY_LE(I)=1;WAY_RK(I)=0;END;
                                ST_DPT,N_WAY=0B;
                                /* COMPUTATION OF THE VARIOUS WAYS
                                I11,I12 DEFINE A RANGE WHERE THE TOTAL SLACKS ARE EQUAL
                                I21,I22 DEFINE A WAY */
                                I11=1B;
                                W90:
                                DO I12 = I11 TO NT; /*1*/
                                T_X=T_ORD(I12);
                                IF I12 = I11 THEN GO TO W76;
                                IF T_SLK(T_X) <= T_SLK(T_ORD(I12-1B)) THEN GO TO W77;
                                W76: END; /*1*/
                                I12=NT; GO TO W81;
                                W77:I12=I12-1B;
                                W81: IF I12 = I11 THEN DO; I21,I22=I12; GO TO W_H;END;
                                I22 = I11;
                                W1:I21,I22T=I22;
                                W2: J1=I22T+1B;
                                T_X=T_ORD(I22T);
                                DO J=J1 TO I12; /*1*/
                                /* I22T AND J ARE THEY RELATED */
                                T_X1=T_ORD(J);
                                DO JJ=X_CONS(T_X,1,1) TO X_CONS(T_X,2,1); /*2*/
                                IF CONSTR(JJ,1) = T_X1 THEN GO TO W4;
                                END; /*2*/

```

Abb. 3(9)

```

END; /*1*/
T_X1= T_ORD(I21);
DO J=J1 TO I12; /*1*/
  /* J AND I21 ARE THEY RELATED */
  T_X= T_ORD(J);
  DO JJ=X_CONS(T_X,1,1) TO X_CONS(T_X,2,1); /*2*/
  IF CONSTR(JJ,1) = T_X1 THEN GO TO W6;
END; /*2*/
END; /*1*/
I22=I22T;
  /* ONE WAY HAS BEEN FOUND. */
W_H:
N_WAY=N_WAY+1B; WAY_TB(N_WAY,1B)=ST_DPT+1B; /*ORDINATES.*/
  /* INCIDENCE ON D_SLK. */
JJ=I22-I21+1B;
DO KK=I21 TO I22;
  T_X=T_ORD(KK);
  WAY_LE(T_X)=JJ; WAY_RK(T_X)=KK-I21;
  ST_DPT=ST_DPT+1B; STACK(ST_DPT)=T_X; /* THESE FOR THE ORDINATES. */
END;
  /* END OF WAY. */
WAY_TB(N_WAY,10B)=ST_DPT;
T_X=STACK(WAY_TB(N_WAY,10B));
AMPL(N_WAY)=TIME(T_X,10B,1B)+SLACK(T_X) -
TIME(STACK(WAY_TB(N_WAY,1B)),1B,1B);
IF T_SLK(T_X) = 0 THEN AMPL(N_WAY)= - AMPL(N_WAY);
I22=I22+1B;
IF I22 <= I12 THEN GO TO W1;
W91: I11= I12+1B; IF I11 <= NT THEN GO TO W90; GO TO W_END;
W4: DO K= J-1B BY -1B TO I22T+1B;
  JJ=T_ORD(K); T_ORD(K)=T_ORD(K+1B); T_ORD(K+1B)=JJ;
END;
GO TO W8;
W6: DO K= J-1B BY -1B TO I21;
  JJ=T_ORD(K); T_ORD(K)=T_ORD(K+1B); T_ORD(K+1B)=JJ;
END;
W8:
  I22T=I22T+1B; GO TO W2;
W_END:
  /* COMPUTATION OF THE DISTRIBUTED SLACKS. */
DO I=1B TO NT;
  IF T_SLK(I) = 0 THEN D_SLK(I)=0; ELSE
  D_SLK(I)=FLOAT(T_SLK(I))/FLOAT(WAY_LE(I));
END;
  /*CALENDAR PREPARATION */
TDAY(1B)=1; TDAY(10B)=1; TDAY(11B)=68; DOW=1;
N_DAY = CEIL(END_T *5+1);
I=0B; PR_SW='10000000'B; W_DAY='00000000'B;
PUT FILE (SYSPRINT) PAGE;
PUT SKIP DATA(N_DAY);
I1=0B;
I2=1;
WORK_T:
IW=0B;
  /* IS IT A WORKING DAY */
  /* IS IT A WEEK END */
IF DOW=0B | DOW= 110B THEN GO TO SLEEP;
  /* IS IT A 1 OR 6 JANUARY */

```

Abb.3(10)

```

IF TDAY(10B)=1 & ( TDAY(1B)=1 | TDAY(1B)=6 ) THEN GO TO SLEEP;          00058100
/* IS IT A FIRST MAY */                                                00058200
IF TDAY(1B)=1 & TDAY(10B)=5 THEN GO TO SLEEP;                          00058300
/* IS IT CHRISTMAS */                                                  00058400
IF TDAY(10B)=12 & TDAY(1B) >= 24 THEN GO TO SLEEP;                    00058500
/* MOBILE DAYS */                                                       00058600
JJ= TDAY(11B) - 67;                                                    00058700
IF TDAY(10B) < 3 | TDAY(10B) > 4 THEN GO TO NOOS;                    00058800
IF TDAY(1B) = M_DAY(JJ, 1B) & TDAY(10B) =M_DAY(JJ, 10B) THEN        00058900
GO TO SLEEP;                                                            00059000
IF M_DAY(JJ,10B) = 3 THEN GO TO MARCH;                                  00059100
ELSE DO;                                                                 00059200
IF M_DAY(JJ,1B) >= 4 THEN DO;                                           00059300
    MARCH: IF TDAY(1B) + 3 = M_DAY(JJ,1B) &                            00059400
        TDAY(10B) = M_DAY(JJ,10B) THEN GO TO SLEEP;                    00059500
        END;                                                            00059600
ELSE DO;                                                                 00059700
IF TDAY(1B) - 28 = M_DAY(JJ,1B) & TDAY(10B) = 3 THEN GO TO SLEEP;    00059800
END;                                                                      00059900
END;                                                                      00060000
NOOS:                                                                    00060100
IF TDAY(1B) = M_DAY(JJ, 11B) & TDAY(10B) =M_DAY(JJ,100B) THEN        00060200
GO TO SLEEP;                                                            00060300
IF TDAY(1B) = M_DAY(JJ,101B) & TDAY(10B) =M_DAY(JJ,110B) THEN        00060400
GO TO SLEEP;                                                            00060500
/*WORKING DAY*/                                                         00060600
IW=1B;                                                                    00060700
CAL(I,*)=TDAY;    W_DAY='10000000'B;                                    00060800
I=I+1B;                                                                    00060900
IF I=2000 THEN I=0B;                                                    00061000
SLEEP:                                                                    00061100
IF TDAY(11B) > 86 THEN DO;                                              00061200
PUT FILE (SYSPRINT) EDIT('TAETIGKEITEN UEBER 1986, HIERFUER MUESSEN IM 00061300
UELLPROGRAMM DIE ENTSPRECHENDEN MOBILLEN FEIERTAGE EINGEGEBEN WERDEN') 00061400
(SKIP(2),A);                                                            00061500
GET FILE (SYSIN) SKIP; GO TO NEXT; END;                                  00061600
IF PR_SW THEN DO;                                                       00061700
IF TDAY(11B)=DATE(1B,11B) THEN DO;                                       00061800
    IF TDAY(10B)=DATE(1B,10B) THEN DO;                                       00061900
        IF TDAY(1B)=DATE(1B,1B) THEN DO;                                       00062000
I1=1B; I=0B; PR_SW='00000000'B;                                       00062100
IF W_DAY THEN DO; CAL(I,*)=TDAY;I=1B; END;                               00062200
PUT FILE (SYSPRINT) SKIP EDIT (I1,W(IW),I,D_N(DOW),TDAY,I2,DOW)        00062300
(F(10),X(1),A,F(10),X(1),A,3 F(4),F(5));                                00062400
        END; END; END;                                                  00062500
END;                                                                      00062600
ELSE DO;                                                                 00062700
IF I> N_DAY THEN GO TO F_CAL;                                           00062800
PUT FILE (SYSPRINT) SKIP EDIT (I1,W(IW),I,D_N(DOW),TDAY,I2)           00062900
(F(10),X(1),A,F(10),X(1),A,3 F(4),F(5));    END;                       00063000
I1=I1+1B;                                                                00063100
I2=I2+1;                                                                00063200
IF TDAY(1B) < 28 THEN NEXT_D: TDAY(1B)=TDAY(1B)+1;                    00063300
    ELSE DO; /*2-2*/                                                    00063400
IF TDAY(10B)=2 & MOD(TDAY(11B),4)=0 THEN                                00063500
L_MN=M_LEN(TDAY(10B),10B);                                              00063600
    ELSE                                                                00063700
L_MN=M_LEN(TDAY(10B),1B);                                              00063800

```

Abb. 3(11)

```

IF TDAY(1B) < L_MN THEN GO TO NEXT_D;
IF TDAY(10B)=12 THEN DO; /*3-1:NEW YEAR*/
TDAY(10B)=1;TDAY(11B)=TDAY(11B)+1;
I2=1;
END; /*3-1*/
ELSE
TDAY(10B)=TDAY(10B)+1;
TDAY(1B)=1;
END; /*2-2*/
/* DOW: DAY OF THE WEEK. 0 FOR SUNDAY,
1 FOR MONDAY,
2 FOR TUESDAY,
3 FOR WEDNESDAY,
4 FOR THURSDAY,
5 FOR FRIDAY AND
6 FOR SATURDAY. */
/* DOW UPDATING*/
DOW=DOW+1B; IF DOW=111B THEN DOW=0;
GO TO WORK_T;
F_CAL:
/* GENERAL OUTPUT LISTING. */
CALL DA_COM(E_DA,END_T);
IF DATE(2,11B) = 0 | DATE(2,11B) > E_DA(11B) THEN GO TO KALEND;
IF DATE(2,11B) = E_DA(11B) THEN DO;
IF DATE(2,10B) > E_DA(10B) THEN GO TO KALEND;
IF DATE(2,10B) = E_DA(10B) THEN DO;
IF DATE(2, 1B) >= E_DA( 1B) THEN GO TO KALEND;
END; END;
PUT FILE (SYSPRINT) EDIT ( '*****
*****' )(SKIP(3),A);
PUT FILE (SYSPRINT) EDIT
(TITLE,'START.',DATE(1,*),'ENDE.',DATE(2,*),'REPORT.',DATE(3,*),
'ENDTERMIN KANN NICHT EINGEHALTEN WERDEN.')(SKIP(2),A,3 (A,P'ZZ',2 P'.ZZ',X(10)),SKIP(2),A);
DO I=1B TO 11B; DATE(2,I)=E_DA(I);END;
PUT FILE (SYSPRINT) EDIT ('BERECHNETER ENDTERMIN IST: ',DATE(2,*))
(SKIP(1),A,P'ZZ',2 P'.ZZ');
PUT FILE (SYSPRINT) EDIT ( '*****
*****' )(SKIP(3),A);
GO TO WEITER;
KALEND:
DO I=1B TO 11B; DATE(2,I)=E_DA(I);END;
PUT FILE (SYSPRINT) EDIT ('KEINE FEHLER GEFUNDEN.')(SKIP(3),A);
WEITER:
PAGE=0;CNT=0B;
DO KK =1B TO NT; T_X= T_ORD(KK);
IF CNT=0B THEN DO; /* HEADER */
PAGE=PAGE+1;CNT=110010B;
PUT FILE (SYSPRINT) EDIT
('PROJEKT.',TITLE,'-SORT.NACH SL.+R.-') (PAGE,A,A,COLUMN(89),A);
PUT FILE (SYSPRINT) EDIT
('START.',DATE(1,*),'ENDE.',DATE(2,*),'REPORT.',DATE(3,*),'SEITE.',PAGE)
(SKIP(2),3 (A,P'ZZ',2 P'.ZZ',X(10)),COLUMN(102),A,F(3));
PUT FILE (SYSPRINT) EDIT
('NR.','ABT.','TAETIGKEIT','DAUER','F.BEG.','ASP.END','FS.END','T.PFZ',
'F.PFZ','NR/WEG','VT.PFZ','FR.BEG.','ASP. END','SP. END')
(SKIP(3),COLUMN( 2),A,COLUMN( 6),A,COLUMN(16),A,COLUMN(35),A,
COLUMN(42),A,COLUMN(49),A,COLUMN(58),A,COLUMN(67),A,
00063900
00064000
00064100
00064200
00064300
00064400
00064500
00064600
00064700
00064800
00064900
00065000
00065100
00065200
00065300
00065400
00065500
00065600
00065700
00065800
00065900
00066000
00066100
00066200
00066300
00066400
00066500
00066600
00066700
00066800
00066900
00067000
00067100
00067200
00067300
00067400
00067500
00067600
00067700
00067800
00067900
00068000
00068100
00068200
00068300
00068400
00068500
00068600
00068700
00068800
00068900
00069000
00069100
00069200
00069300
00069400
00069500
00069600

```

Abb. 3(12)

```

COLUMN(75),A,COLUMN(82),A,COLUMN(89),A,COLUMN(97),A,COLUMN(105),A, 00069700
COLUMN(115),A); 00069800
PUT FILE(SYSPRINT) SKIP; 00069900
END; 00070000
/* BODY. */ 00070100
P_TIME(1)=TIME(T_X,1B,1B); 00070200
P_TIME(2)=TIME(T_X,10B,10B); 00070300
P_TIME(3)=TIME(T_X,10B,1B) + D_SLK(T_X); 00070400
CALL DA_COM(P_DA1,P_TIME(1)); 00070500
CALL DA_COM(P_DA2,P_TIME(2)); 00070600
CALL DA_COM(P_DA3,DECIMAL(TIME(T_X,1B,1B)+(1+WAY_RK(T_X))*D_SLK(T_X)+ 00070700
DUR(T_X),7,3)); 00070800
/* SOLL BEGINN. */ 00070900
PUT FILE (SYSPRINT) EDIT 00071000
(TASK(T_X),DEPT(T_X),DESCR(T_X),DUR(T_X),P_TIME,T_SLK(T_X),SLACK(T_X), 00071100
WAY_RK(T_X),D_SLK(T_X),P_DA1,P_DA2,P_DA3) 00071200
(SKIP,F(4),X(1),A,X(1),A, 6 F(8,1),F(6),F(9,3),3 (X(1),P'ZZ',2 P'.ZZ'));00071300
CNT=CNT-1B; 00071400
END; 00071500
/* DEPARTMENT SORT. */ 00071600
DO I=1B TO NT-1B; /*1*/ 00071700
DO K=NT-1B BY -1B TO I; /*2*/ 00071800
IF DEPT(T_ORD(K)) > 00071900
DEPT(T_ORD(K+ 1B)) THEN GO TO DEPT_X; 00072000
IF DEPT(T_ORD(K)) = 00072100
DEPT(T_ORD(K+ 1B)) THEN DO; /*3*/ 00072200
IF T_SLK(T_ORD(K)) > 00072300
T_SLK(T_ORD(K+1B)) THEN DO; /*4*/ DEPT_X; 00072400
/* EXCHANGE. */ 00072500
JJ=T_ORD(K); T_ORD(K)=T_ORD(K+1B); T_ORD(K+1B)= JJ; 00072600
END; /*4*/ 00072700
END; /*3*/ 00072800
END; /*2*/ 00072900
END; /*1*/ 00073000
/* DEPARTMENT OUTPUT LISTING. */ 00073100
PAGE=0;CNT=0B; 00073200
DO KK =1B TO NT; T_X= T_ORD(KK); 00073300
IF CNT > 0B THEN DO; 00073400
IF DEPT(T_X) = 0_DEPT THEN GO TO D_BODY; 00073500
END; 00073600
/* HEADER. */ 00073700
PAGE=PAGE+1;CNT=110010B; 00073800
PUT FILE (SYSPRINT) EDIT 00073900
('PROJEKT.',TITLE,'-ZUSTAENDIGKEITEN-') (PAGE,A,A,COLUMN(89),A); 00074000
PUT FILE (SYSPRINT) EDIT 00074100
('START.',DATE(1,*),'ENDE.',DATE(2,*),'REPORT.',DATE(3,*),'SEITE.',PAGE)00074200
(SKIP(2),3 (A,P'ZZ',2 P'.ZZ',X(10)),COLUMN(102),A,F(3)); 00074300
PUT FILE (SYSPRINT) EDIT 00074400
('AUFGABENBEREICH.',DEPT(T_X)) (SKIP(2),A,A); 00074500
PUT FILE (SYSPRINT) EDIT 00074600
('NR.','ABT.','TAETIGKEIT','DAUER','FRUEH.BEGINN','SOLL.BEGINN', 00074700
'SOLL - ENDE','VT.PFZ.') 00074800
(SKIP(3),COLUMN( 2),A,COLUMN( 7),A,COLUMN(20),A,COLUMN(38),A, 00074900
COLUMN(47),A,COLUMN(61),A,COLUMN(74),A,COLUMN(107),A); 00075000
PUT FILE (SYSPRINT) SKIP(3); 00075100
D_BODY: 00075200
P_TIME(1)=TIME(T_X,1B,1B); 00075300
P_TIME(3)=TIME(T_X,10B,1B) + D_SLK(T_X); 00075400

```

Abb. 3(13)

```

CALL DA_COM(P_DA1,P_TIME(1));
CALL DA_COM(P_DA2,DECIMAL(TIME(T_X,1B,1B) + WAY_RK(T_X) *
D_SLK(T_X),7,3));
CALL DA_COM(P_DA3,DECIMAL(TIME(T_X,1B,1B)+WAY_RK(T_X)*D_SLK(T_X)+
DUR(T_X),7,3));
PUT FILE (SYSPRINT) EDIT
(TASK(T_X),DEPT(T_X),DESCR(T_X),DUR(T_X),P_DA1,P_DA2,P_DA3,D_SLK(T_X))
(SKIP,F(4),X(1),A,X(1),A,COLUMN(35),F(9,1),3 (X( 5),P'ZZ',2 P'.ZZ'),
COLUMN(105),F(9,3));
O_DEPT=DEPT(T_X);
CNT=CNT-1B;
END;
PLOT_SW = '0'B;
GO TO REND;
/* PLOT PREPARATION FOR DSNAME = GRAPH */
TERMP:
NE=0;
DO I=1B TO NT;
T_X=T_ORD1(I);
IF NE=0 | TIME(T_X,1B,1B) = EV_X(NE) THEN DO;
NE=NE+1B; EV_X(NE)=TIME(T_X,1B,1B);
END;
EV_ARC(T_X,1B)=NE;
END;
DO I= 1B TO NT;
T_X=T_ORD1(I); TEST_X=TIME(T_X,10B,1B)+ SLACK(T_X);
DO K=1B TO NE;
IF TEST_X = EV_X(K) THEN GO TO E12;
END;
E12: EV_ARC(T_X,10B)=K;
END;
NE=NE+1B;
EV_X(NE)=END_T;
G_DEF.G_TI= TITLE || ' ';
G_DEF.G_NE=NE;
G_DEF.G_NT=NT;
WRITE FILE(GRAPH) FROM (G_DEF);
/* ORDINATES.
POINT 1. STORE FOR EACH WAY ITS AMPLITUDE.
POINT 2. SORT ON THESE AMPLITUDES.
POINT 3. ASSIGN AS ORDINATES THE RANK AFTER THIS SORT,
EXCEPT IF THE TASK BELONGS TO THE FIRST CRITICAL WAY
IN WHICH CASE,
THE ORDINATES SHOULD BE ZERO.(IN CASE OF CONFLICT,
ZERO SHOULD HAVE PRIORITY. */
/* POINT 1: */
/* SEE ABOVE: COMPUTATION OF THE VARIOUS WAYS. */
/* POINT 2 : */
/* SORT INITIALIZATION. */
DO I=1B TO N_WAY; W_ORD(I)=I; END;
DO I=1B TO N_WAY - 1B; /*1*/
DO K=N_WAY - 1B BY -1B TO I ; /*2*/
IF AMPL(W_ORD(K )) >
AMPL(W_ORD(K+1B)) THEN DO; /*3*/
/* EXCHANGE. */
JJ=W_ORD(K); W_ORD(K)=W_ORD(K+1B); W_ORD(K+1B)= JJ;
END; /*3*/
END; /*2*/
00075500
00075600
00075700
00075800
00075900
00076000
00076100
00076200
00076300
00076400
00076500
00076600
00076700
00076800
00076900
00077000
00077100
00077200
00077300
00077400
00077500
00077600
00077700
00077800
00077900
00078000
00078100
00078200
00078300
00078400
00078500
00078600
00078700
00078800
00078900
00079000
00079100
00079200
00079300
00079400
00079500
00079600
00079700
00079800
00079900
00080000
00080100
00080200
00080300
00080400
00080500
00080600
00080700
00080800
00080900
00081000
00081100
00081200

```

Abb. 3(14)

```

END; /*1*/
/* POINT 3 : */
DO I=1B TO NE; PLT_SW='00000000'B; END;
FI_W='11111111'B;
DO I=1B TO N_WAY;
KK=W_ORD(I);
I11=WAY_TB(KK,1B);I12=WAY_TB(KK,10B);
/* CHECK WHETHER THIS WAY IS A CRITICAL ONE. */
IF T_SLK(STACK(I11)) = 0 & FI_W THEN DO;
F_C_W='11111111'B; FI_W='00000000'B;
END;
ELSE F_C_W='00000000'B;
DO ST_DPT= I11 TO I12;
T_X=STACK(ST_DPT);
CALL DA_COM(P_DA2,DECIMAL(TIME(T_X,1B,1B),7,3));
II=EV_ARC(T_X,1B);
JJ=EV_ARC(T_X,10B);
G_STR.A_N1=II;
G_STR.A_N2=JJ;
G_STR.A_DSC=BINARY(TASK(T_X),31,0);
G_STR.A_ABT=DEPT(T_X);
G_STR.A_DES=DESCR(T_X);
G_STR.A_DAU=DUR(T_X);
IF I11 = I12 THEN G_STR.A_TYPE= 1 ; ELSE G_STR.A_TYPE = 0 ;
G_STR.A_SBEGIN=P_DA2;
WRITE FILE(GRAPH) FROM (G_STR);
/* ORDINATES.*/
IF PLT_SW(II) THEN GO TO PLT1;
EV_Y(II)=FLOAT(KK);PLT_SW(II)='11111111'B;
PLT1:IF PLT_SW(JJ) THEN GO TO PLT2;
EV_Y(JJ)=FLOAT(KK);PLT_SW(JJ)='11111111'B;
PLT2:IF F_C_W THEN DO; EV_Y(II),EV_Y(JJ)=0;END;
END;
END;
DO I=1B TO NE;
COOR.X1=EV_X(I);
COOR.X2=EV_Y(I);
WRITE FILE(GRAPH) FROM (COOR );
END;
/* THE END */
DA_COM: PROC(Y,X);
DCL
XX DEC FIXED(5,1),
Y(3) DEC FIXED (3) ,
X DEC FIXED (7,3) ;
XX=5*X;
JJ=ROUND(XX,0);
DO J=1B TO 11B; Y(J)= CAL(JJ,J); END;
END DA_COM;
TERMPT:
GET FILE(SYSIN) SKIP; GO TO NEXT;
FIN:
IF PLOT_SW THEN DO;
PUT FILE (SYSPRINT) EDIT ('*****') (PAGE,A);
PUT FILE (SYSPRINT) EDIT ('WEGEN FEHLERHAFTER EINGABE WIRD KEINE ZEICHNUN
NG VERSUCHT')(SKIP(3),A);
PUT FILE (SYSPRINT) EDIT ('*****') (SKIP(3),A);
SIGNAL ERROR;
END;
END TERM ;

```

```

00081300
00081400
00081500
00081600
00081700
00081800
00081900
00082000
00082100
00082200
00082300
00082400
00082500
00082600
00082700
00082800
00082900
00083000
00083100
00083200
00083300
00083400
00083500
00083600
00083700
00083800
00083900
00084000
00084100
00084200
00084300
00084400
00084500
00084600
00084700
00084800
00084900
00085000
00085100
00085200
00085300
00085400
00085500
00085600
00085700
00085800
00085900
00086000
00086100
00086200
00086300
00086400
00086500
00086600
00086700
00086800
00086900
00087000
00087100
00087200
00087300
00087400

```

Abb. 3 (15)

| | |
|--|----------|
| DIMENSION IBUF(8000) | 00000100 |
| M=8000 | 00000200 |
| CALL CALCIN (IBUF,M,1) | 00000300 |
| C H1 = ZEICHENHOEHE FUER BESCHRIFTUNG DES GRAPHEN | 00000400 |
| C H2 = ZEICHENHOEHE FUER DIE TAETIGKEITSNUMMERIERUNG UND KNOTENSYMBOL | 00000500 |
| C UND DATUM AM OBEREN ZEICHENRAND | 00000600 |
| C H3 = ABSTAND DER TAETIGKEITSNUMMERIERUNG VON DER LINIE | 00000700 |
| C H4 = ZEICHENHOEHE FUER TAETIGKEITSBESCHRIFTUNG, DAUER UND ABTEILUNG | 00000800 |
| C H5 = ZEICHENHOEHE FUER TAETIGKEITSNUMMERIERUNG AM OBEREN ZEICHENRAND | 00000900 |
| C UND ARBEITSWOCHENANGABE AM OBEREN ZEICHENRAND | 00001000 |
| C DX ALTE VERSION : MAXIMALE AUSDEHNUNG DES GRAPHEN | 00001100 |
| C DX NEUE VERSION : EINHEITSABSCHNITT AUF DER X-AXE X(N) - X(N-1) | 00001200 |
| C DY = MAXIMALE AUSDEHNUNG DER ORDINATE | 00001300 |
| H1=.49 | 00001400 |
| H2=.16 | 00001500 |
| H3=0. | 00001600 |
| H4=0.08 | 00001700 |
| H5=0.1 | 00001800 |
| DX=1. | 00001900 |
| DY=25. | 00002000 |
| L=1 | 00002100 |
| CALL ZAEHL (L) | 00002200 |
| 1 CALL TERMPL(H1,H2,H3,H4,H5,DX,DY) | 00002300 |
| L=L-1 | 00002400 |
| IF (L .GT. 0) GO TO 1 | 00002500 |
| CALL PLOTG (0.,0.,999) | 00002600 |
| STOP | 00002700 |
| END | 00002800 |

CALCPL Abb. 4

| | |
|---|----------|
| SUBROUTINE CALCIN(IBUF,N,L) | 00000100 |
| C DIESE SUBROUTINE BEWIRKT DIE INITIALISIERUNG DES CALCOMP-PLOTTERS | 00000200 |
| DIMENSION IBUF(1),JOBNAM(2),JDATUM(5) | 00000300 |
| CALL JOBN (JOBNAM,JDATUM) | 00000400 |
| CALL PLOTS (IBUF,N) | 00000500 |
| CALL PLOTG(0.,0.,-3) | 00000600 |
| CALL PLOTG(0.,1.5,3) | 00000700 |
| CALL PLOTG(0.,26.5,2) | 00000800 |
| CALL SYMBOL(3.,8.,1.,JOBNAM,90.,8) | 00000900 |
| CALL SYMBOL(5.,7.,0.5,JDATUM,90.,20) | 00001000 |
| CALL PLOTG(7.,0.,-3) | 00001100 |
| CALL WTJOBN (JOBNAM,L) | 00001200 |
| C WTJOBN SCHREIBT DIE MESSAGE AN DIE KONSOLE, | 00001300 |
| C DASS PLOTTAPE BESCHRIEBEN WIRD WENN L=1 WIRD ZUSAETZLICH 'WEISSES | 00001400 |
| C PAPIER ' AN DIE KONSOLE GESCHRIEBEN | 00001500 |
| RETURN | 00001600 |
| END | 00001700 |

CALCIN Abb. 5(1)

```

JOBN      CSECT                                00000100
*****
*
* SUBROUTINE JOBN (JOBNAM,JDATUM)                00000200
* DIESES PROGRAMM UEBERGIBT IM ARGUMENT JOBNAM DEN JOBNAMEN UND IM      00000300
* ARGUMENT JDATUM DAS DATUM UND DIE UHRZEIT DES JOBS                    00000400
* JOBNAM BENDETIGT 8 BYTES                                              00000500
* JOBNAM BENDETIGT 20 BYTES                                           00000600
*
*
*****
      SAVE (14,12),,*
      LR   3,13
      LA   13,PLSAVE-JOBN(15)
      USING PLSAVE,13
      B    BEGINN
*****
PLSAVE   DS    18F
JOBNAME  DC    D'0'
DATUM    DC    5F'0'
DAT      DC    F'0'
***
*****
BEGINN   EQU   *
      LR   4,1
      ST   13,8(0,3)      FORWARD POINTER
      ST   3,4(0,13)     BACKWARD POINTER
***
*****
* ADRESSE DER ARGUMENTE  JOBNAM UND JDATUM IN REG 2 U. 3
***
      LM   2,3,0(4)
*****
* JOBNAME AUS TIOT FISCHEN , ADR. DES JOBNAMENS STEHT IN JOBNAME
      EXTRACT JOBNAME,'S',FIELDS=(TIOT)
      L    5,JOBNAME
***
* JOBNAME IN DIE 8 BYTES DES 1. ARGUMENTES MOVEN
      MVC  0(8,2),0(5)
***
*****
* UHRZEIT UND DATUM
*
      TIME DEC
*****
*
* IN REG 1 STEHT DAS DATUM IN GEPACKTER FORM MIT 'C' AM ENDE ZUM
* ENTPACKEN OOOYDDDC
*
      ST   1,DAT
      UNPK DATUM+10(5),DAT+1(3)
      MVC  DATUM(3),DATUM+12
      MVC  DATUM+3(7),TAG
***
*****
*
* IN REG 0 STEHT DIE UHRZEIT IN GEPACKTER FORM HHMMSSZH

```

zu CALCIN Abb.5 (2)

```

* Z=ZEHNTEL-, H=HUNDERTSTELSEKUNDE
*
      ST      0,DAT
***
* Z UND H WERDEN NICHT GEBRAUCHT. UM ENTPACKEN ZU KOENNEN WIRD 'OF' IN
* DAS LETZTE BYTE GESETZT.
      MVI     DAT+3,X'OF'
      UNPK   DATUM+13(7),DAT(4)
      MVC    DATUM+18(2),DATUM+15
      MVC    DATUM+15(3),UHR
      MVI     DATUM+12,X'40'
*
*****
* DATUM UND UHRZEIT IN DIE 20 BYTES DES 2. ARGUMENTES MOVEN
      MVC     0(20,3),DATUM
*
*****
      L       13,PLSAVE+4
      RETURN (14,12),T
TAG      DC    C'.TAG 19'
UHR      DC    C'UHR'
      DROP   13
*****
* DIESES PROGRAMM UEBERGIBT EINE MESSAGE AN DEN OPERATOR IN DER DER
* JOBNAME STEHT
* AUFRUF : CALL WTJOBN (JOBNAM,L)
WTJOBN   CSECT
      SAVE   (14,12),,*
      LR     3,13
      LA     13,RGSAVE-WTJOBN(15)
      USING  RGSAVE,13
      B      BEGIN
*
RGSAVE   DC    18F'0'
EINS     DC    F'1'
WP       DC    C'/ WEISSES PAPIER'
BEGIN    EQU   *
      ST    13,8(0,3)      FORWARD POINTER
      ST    3,4(0,13)     BACKWARD POINTER
      LM    4,5,0(1)
      MVC   MESSAGEX(8),0(4)
      L     6,0(0,5)
      C     6,EINS
      BNE   WT
      MVC   MESSAGEY(16),WP
WT        WTO   '*****'
      LA    1,MESSAGE
      WTO   MF=(E,(1))
      WTO   '*****'
      L     13,RGSAVE+4
      RETURN (14,12),T
MESSAGE   WTO   '*** PLOTTE FUER  JOBNAMEX WIRD BESCHRIEBEN
              ',MF=L
MESSAGEX  EQU   MESSAGE+25
MESSAGEY  EQU   MESSAGE+52
END

```

```

00005900
00006000
00006100
00006200
00006300
00006400
00006500
00006600
00006700
00006800
00006900
00007000
00007100
00007200
00007300
00007400
00007500
00007600
00007700
00007800
00007900
00008000
00008100
00008200
00008300
00008400
00008500
00008600
00008700
00008800
00008900
00009000
00009100
00009200
00009300
00009400
00009500
00009600
00009700
00009800
00009900
00010000
00010100
00010200
00010300
00010400
00010500
00010600
00010700
00010800
00010900
*00011000
00011100
00011200
00011300
00011400

```

zu CALCIN Abb. 5 (3)

```

* ROUTINE ZAEHL(L) SUCHT IN DER HOECHSTEN SAVE/AREA DEN HIER GESPEI- 00000100
* CHERTEN WERT VON REG 11. IM ASSEMBLER-RAHMENPROGRAMM WURDE REG 11 00000200
* MIT DER ZAHL X GELADEN, DIE AUF DER EXEC - KARTE IM PARAMETER 00000300
* PARM='NETZPLAN=XFACH' ANGIBT, WIE OFT DIE ZEICHNUNG WIEDERHOLT 00000400
* WERDEN SOLL 00000500
* 00000600
ZAEHL CSECT 00000700
      SAVE (14,12),,* 00000800
      LR 3,13 00000900
      LA 13,R16-ZAEHL(15) 00001000
      USING R16,13 00001100
      B B16 00001200
* 00001300
***** 00001400
R16 DC 18F'0' 00001500
***** 00001600
B16 EQU * 00001700
      LR 2,1 00001800
      ST 13,8(0,3) 00001900
      ST 3,4(0,13) 00002000
***** 00002100
* 00002200
      L 4,0(2) ADRESSE VON L 00002300
* 00002400
* 00002500
* REG 3 ENTHAELT NOCH DIE ADRESSE DER SAVEAREA DES FORTRAN PROGRAMMS 00002600
* CALCPL 00002700
* CALCPL WURDE VOM ASSEMBLER-RAHMENPROGRAMM DURCH LINK AUFGERUFEN. 00002800
* IN REG 5 WIRD DIE ADRESSE DER SAVEAREA DES ASSEMBLER-RAHMENPROGRAMMS 00002900
* GELADEN 00003000
      L 5,4(0,3) 00003100
* IN REG 6 WIRD DIE ADRESSE DER SPEICHERSTELLE FUER REG 11 GELADEN 00003200
      LA 6,64(0,5) 00003300
***** 00003400
      MVC 0(4,4),0(6) ZAHL X IN FELD L (ARGUMENT VON ZAEHL) MOVEN 00003500
***** 00003600
      L 13,R16+4 00003700
      RETURN (14,12),T,RC=0 00003800
      END 00003900

```

ZAEHL Abb.6

```

SUBROUTINE TERMPL(H1,H2,H3,H4,H5,DX,DY) 00000100
LOGICAL *1 AP(1000) ,SHIFT,EOS,FIRST,MEC,STR(1000) ,ABTLG(1000,5) 00000200
LOGICAL *1 HX(800),STN(1000) 00000300
REAL *8 ABT 00000400
INTEGER *2 T(1000),DAT(1000,3) 00000500
INTEGER A(1000,2),DESCR(1000) 00000600
DIMENSION TITLE(13),X(1000),Y(1000),Z(1000),U(1000) 00000700
DIMENSION XE(1000),YA(1000),YE(1000),AMAC(1000) 00000800
DIMENSION XA(1000) 00000900
DIMENSION NAMED1(2),NAMED2(3),NAMET(1000,5),DAUER(1000) 00001000
C PLOT GENERAL INITIALIZATION. 00001100
C H1 = ZEICHENHOEHE FUER BESCHRIFTUNG DES GRAPHEN 00001200
C H2 = ZEICHENHOEHE FUER DIE TAETIGKEITSNUMMERIERUNG UND KNOTENSYMBOL 00001300
C UND DATUM AM OBEREN ZEICHENRAND 00001400
C H3 = ABSTAND DER TAETIGKEITSNUMMERIERUNG VON DER LINIE 00001500
C H4 = ZEICHENHOEHE FUER TAETIGKEITSBESCHRIFTUNG, DAUER UND ABTEILUNG 00001600
C H5 = ZEICHENHOEHE FUER TAETIGKEITSNUMMERIERUNG AM OBEREN ZEICHENRAND 00001700
C UND ARBEITSWOCHENANGABE AM OBEREN ZEICHENRAND 00001800
C DX ALTE VERSION : MAXIMALE AUSDEHNUNG DES GRAPHEN 00001900
C DX NEUE VERSION : EINHEITSABSCHNITT AUF DER X-AXE X(N) - X(N-1) 00002000
C DY = MAXIMALE AUSDEHNUNG DER ORDINATE 00002100
YNUM=DY+2.02 00002200
FDAT=DY-35.*H2 00002300
YDAT=DY+0.72 00002400
YDYU=DY+1.5 00002500
C STANDART WERTE FUER H1,H2,H3,H4,H5,DX,DY : 00002600
C H1=.49 00002700
C H2=.16 00002800
C H3=0. 00002900
C H4=0.08 00003000
C H5=0.1 00003100
C DX=1. 00003200
C DY=25. 00003300
R=.5*H2 00003400
H2D=2.*H2 00003500
DDX=DX+0.1 00003600
HN=-2*H4 00003700
YDAT=DY+0.72 00003800
YMA=DY+1. 00003900
C YDAT = Y-KOORDINATE FUER BESCHRIFTUNG : SOLLBEGINN AM OBEREN Z.RAND 00004000
C XDAT = X(I) - 0.03 ENTSPRECHENDE X- KOORDINATE 00004100
C CALL CALCIN (IBUF,M,N) INITIALISIERUNG DES CALCOMP-PLOTTERS 00004200
C WIRD IM HAUPTPROGRAMM GEMACHT 00004300
REWIND 1 00004400
WRITE(6,200) 00004500
C DATA READING FROM THE PL/1 PROGRAM TERM2 BUILD DATA-SET 00004600
C WITH THE DSNAME=PGRAPH 00004700
1 READ(1,101,END=99)TITLE,NE,NT 00004800
WRITE(6,201)TITLE 00004900
WRITE(6,202)NE,NT 00005000
CALL PLOTG(5.,1.5,-3) 00005100
CALL SYMBOL(0.,-1.5,H1,TITLE,0.,52) 00005200
CALL SYMBOL (-3.,YNUM,H2,'DAUER (ARBEITSWOCHEN)',0.0,21) 00005300
CALL SYMBOL (-3.,YDYU,H2,'NR.DER TAETIGKEITEN',0.0,19) 00005400
CALL SYMBOL (0.97,FDAT,H2,'FRUEHESTER BEGINN DER TAETIGKEITEN:',9000005500 00005500
1.,35) 00005600
DO 3 I=1,NT 00005700
READ(1,103)A(I,1),A(I,2),DESCR(I),ABTLG(I,1),ABTLG(I,2),ABTLG(I,3)00005800

```

Plottprogramm TERMPL Abb.7(1)

```

1),ABTLG(I,4),ABTLG(I,5),NAMET(I,1),NAMET(I,2),NAMET(I,3),NAMET(I,4)00005900
2),NAMET(I,5),DAUER(I),T(I),DAT(I,1),DAT(I,2),DAT(I,3) 00006000
3 WRITE(6,204)I,A(I,1),I,A(I,2),I,DESCR(I),ABTLG(I,1),ABTLG(I,2),ABT00006100
1LG(I,3),ABTLG(I,4),ABTLG(I,5),NAMET(I,1),NAMET(I,2),NAMET(I,3),NAM00006200
2ET(I,4),NAMET(I,5),DAUER(I),T(I),DAT(I,1),DAT(I,2),DAT(I,3) 00006300
DO 2 I=1,NE 00006400
C *****00006500
C X RANGE. 00006600
C 00006700
C *****00006800
C 00006900
C OLD VERSION WITH DIFFERENT LENGTS 00007000
C READ(1,102)X(I),Z(I) 00007100
C 2 WRITE(6,203)I,X(I),Z(I) 00007200
C DXMIN=1.E40 00007300
C DO 42 I=1,NT 00007400
C XT=ABS(X(A(I,1)) - X(A(I,2))) 00007500
C IF(XT .EQ. 0.) GO TO 42 00007600
C IF(XT .LT. DXMIN) DXMIN=XT 00007700
C 42 CONTINUE 00007800
C DX= X(NE) /{DXMIN* 3.} 00007900
C 00008000
C *****00008100
CCC CALL SC(X,NE,DX) FOR OLD VERSION ONLY 00008200
C *****00008300
C 00008400
C *****00008500
C 00008600
C NEW VERSION WITH EQUAL X-UNITS 00008700
C 00008800
READ(1,102)U(I),Z(I) 00008900
2 WRITE(6,203)I,U(I),I,Z(I) 00009000
D=1. 00009100
YRA=DY+2. 00009200
YRAP=DY+1.7 00009300
K=1 00009400
CALL PLOTG (1.,YRA,3) 00009500
DO 45 I=1,NE 00009600
EOS=.FALSE. 00009700
IF (I .EQ. 1) GO TO 46 00009800
DUU=ABS(U(I)-U(I-1)) 00009900
IF (DUU .LT. 0.001) EOS=.TRUE. 00010000
IF (EOS) GO TO 46 00010100
D=D+DX 00010200
K=K+1 00010300
46 X(I)=D 00010400
HX(K)=.FALSE. 00010500
IF (EOS) GO TO 45 00010600
C ZEICHNUNG EINES RAHMENS MIT ABSZISSENMARKIERUNGEN UM DEN GRAPHEN 00010700
C ABSZISSENMARKIERUNGEN ENTSPRECHEN DER ARBEITSDAUER IN WOCHEN 00010800
CALL PLOTG (X(I),YRA,2) 00010900
CALL AROHD (X(I),YRA,X(I),YRAP,0.16,0.05,11) 00011000
CALL NUMBER (X(I),YNUM,H2,U(I),0.0,1) 00011100
CALL PLOTG (X(I),YRA,3) 00011200
B=X(I) 00011300
45 CONTINUE 00011400
C 00011500
CALL PLOTG (B,-0.8,2) 00011600

```

Abb.7(2)

```

DO 18 I=1,K
C RAHMEN UND ABSZISSENMARKIERUNGEN AUCH AM UNTEREN RAND
CALL PLOTG (B,-0.8,2)
CALL PLOTG (B,-0.7,2)
CALL PLOTG (B,-0.8,3)
B=B-DX
18 CONTINUE
C *****
C NEW COORDINATES.
C
K=-1
KT=0
36 EOS=.TRUE.
FIRST=.TRUE.
ZT=KT
DO 38 I=1,NE
IF(Z(I)-ZT) 38,39,40
39 IF(.NOT.FIRST) GO TO 41
K=K+1
FIRST=.FALSE.
41 Y(I)=K
40 EOS=.FALSE.
38 CONTINUE
IF(EOS) GO TO 37
KT=KT+1
GO TO 36
C *****
C INDIVIDUAL PLOT.
37 CONTINUE
CALL SC(Y,NE,DY)
WRITE (6,213) (I,X(I),I,Y(I),I=1,NE)
DO 4 I=1,NE
STR(I)=.TRUE.
4 AP(I)=.FALSE.
LNR=4
C *****
C
DO 5 I =1,NT
EOS=.TRUE.
MEC=.FALSE.
SHIFT=.FALSE.
STN(I)=.FALSE.
NAMED1(1)=NAMET(I,1)
NAMED1(2)=NAMET(I,2)
NAMED2(1)=NAMET(I,3)
NAMED2(2)=NAMET(I,4)
NAMED2(3)=NAMET(I,5)
CALL ABT5 (ABTLG(I,1),ABTLG(I,2),ABTLG(I,3),ABTLG(I,4),ABTLG(I,5),
1ABT)
AMAC(I)=0.
HEIGHT=0.
LHEIGH = 0
D1=.5*(DX-H2)
I1=A(I,1)
I2=A(I,2)

```

Abb.7(3)

| | |
|---|----------|
| X1=X(I1) | 00017500 |
| X2=X(I2) | 00017600 |
| Y1=Y(I1) | 00017700 |
| Y2=Y(I2) | 00017800 |
| XA(I)=X1 | 00017900 |
| XE(I)=X2 | 00018000 |
| YA(I)=Y1 | 00018100 |
| YE(I)=Y2 | 00018200 |
| XZ=X1+H5 | 00018300 |
| YZ=DY+1.8 | 00018400 |
| XDAT=X1-0.03 | 00018500 |
| XL=0.1+(X1-1.)/DX | 00018600 |
| L=XL+1 | 00018700 |
| IF (HX(L)) GO TO 16 | 00018800 |
| DATUM1=DAT(I,1) | 00018900 |
| DATUM2=DAT(I,2) | 00019000 |
| DATUM3=DAT(I,3) | 00019100 |
| CALL NUMBER (XZ,YZ,H5,FLOAT(DESCR(I)),0.0,-1) | 00019200 |
| CALL NUMBER (XDAT,YDAT,H2,DATUM1,90.,0) | 00019300 |
| CALL NUMBER (999.,999.,H2,DATUM2,90.,0) | 00019400 |
| CALL NUMBER (999.,999.,H2,DATUM3,90.,-1) | 00019500 |
| HX(L)=.TRUE. | 00019600 |
| 16 CONTINUE | 00019700 |
| WRITE (6,104) I,DESCR(I),DAT(I,1),DAT(I,2),DAT(I,3),L | 00019800 |
| IF(I.EQ. 1) GO TO 8 | 00019900 |
| IM1=I-1 | 00020000 |
| L=I | 00020100 |
| DO 9 KK=1,IM1 | 00020200 |
| DXAN=ABS(X1-XA(KK)) | 00020300 |
| IF (DXAN .GT. 0.001) GO TO 11 | 00020400 |
| YZ=YZ-1.4*H5 | 00020500 |
| IF (STN(KK)) GO TO 11 | 00020600 |
| CALL NUMBER (XZ,YZ,H5,FLOAT(DESCR(L)),0.0,-1) | 00020700 |
| STN(KK)=.TRUE. | 00020800 |
| 11 CONTINUE | 00020900 |
| DXEN=X1-XE(KK) | 00021000 |
| IF (DXEN .GE. 0.0) GO TO 9 | 00021100 |
| DYAN=ABS(Y1-YA(KK)) | 00021200 |
| DYEN=ABS(Y2-YE(KK)) | 00021300 |
| IF (DYAN .GT. 0.001 .OR. DYEN .GT. 0.001) GO TO 9 | 00021400 |
| MEC=.TRUE. | 00021500 |
| IF(I1.NE.A(KK,1) .OR. I2.NE. A(KK,2)) GO TO 9 | 00021600 |
| LHEIGHT = 1 | 00021700 |
| HEIGHT=1.+AMAC(KK)+HEIGHT | 00021800 |
| 9 CONTINUE | 00021900 |
| IF (LHEIGHT .NE. 0.) SHIFT=.TRUE. | 00022000 |
| IF (.NOT.SHIFT .AND. MEC) AMAC(I)=3.5 | 00022100 |
| 8 CONTINUE | 00022200 |
| WRITE(6,7000)I,DESCR(I),SHIFT,I1,I2,X1,X2,Y1,Y2,ABT | 00022300 |
| C ***** | 00022400 |
| C | 00022500 |
| C | 00022600 |
| C FIRST SYMBOL. | 00022700 |
| C | 00022800 |
| C | 00022900 |
| IF (AP(I1)) GO TO 21 | 00023000 |
| CALL PLOT(X1,Y1,3) | 00023100 |
| CALL SYMBOL(X1,Y1,H2 ,1,0.,-1) | 00023200 |

Abb. 7(4)

| | | |
|----|--|----------|
| | AP(I1)=.TRUE. | 00023300 |
| 21 | CALL PLOTG (X1,Y1,3) | 00023400 |
| | OY=Y2 | 00023500 |
| | OX=X2 | 00023600 |
| | IF (.NOT. MEC) GO TO 33 | 00023700 |
| | Y2=Y2 + HEIGHT | 00023800 |
| C | IF Y2 EXCEEDS MAXIMUM PLOTTINGLENGTH IT IS REDUCED | 00023900 |
| | IF (Y2.GT.YMA) Y2=DY+HEIGHT/6. | 00024000 |
| C | ***** | 00024100 |
| C | | 00024200 |
| C | | 00024300 |
| C | LINE. | 00024400 |
| C | | 00024500 |
| C | | 00024600 |
| C | ***** | 00024700 |
| | DXP=X2-X1 | 00024800 |
| | ABY=ABS(OY-Y1) | 00024900 |
| | IF (ABY .LT. 0.001) GO TO 14 | 00025000 |
| | EOS=.FALSE. | 00025100 |
| | IF(Y1.LT.Y2) GO TO 14 | 00025200 |
| | EOS=.TRUE. | 00025300 |
| | Y2=Y1+HEIGHT | 00025400 |
| | IF (Y2.GT.YMA) Y2=DY+HEIGHT/6. | 00025500 |
| 14 | CALL PLOTG(X1,Y2,2) | 00025600 |
| | Y1=Y2 | 00025700 |
| | Y2=Y2+AMAC(I) | 00025800 |
| | IF (Y2.GT.YMA) Y2=DY+0.5 | 00025900 |
| | IF (AMAC(I) .GT. 0.0 .AND. ABY .GT. 0.001) GO TO 62 | 00026000 |
| | IF (HEIGHT .GT. 0.0) GO TO 31 | 00026100 |
| 63 | X2=.5*(X1+X2) | 00026200 |
| | CALL ARRDES(D1,HN,H3,H2D,X1,X2,Y1,Y2,DXP,DYP,X1A,X2A,Y1A,Y2A,XS,YS | 00026300 |
| | 1,XS1,XS2,YS1,YS2,AS,R,C,XS3,YS3) | 00026400 |
| | XX1A=X1A | 00026500 |
| | YY1A=Y1A | 00026600 |
| | XX2A=X2 | 00026700 |
| | YY2A=Y2 | 00026800 |
| | GO TO 51 | 00026900 |
| 62 | IF (HEIGHT .EQ. 0.0 .AND. EOS) GO TO 63 | 00027000 |
| | X2=.5*(X1+X2) | 00027100 |
| 31 | XX1A=X1 | 00027200 |
| | XX2A=X2 | 00027300 |
| | YY1A=Y1 | 00027400 |
| | YY2A=Y2 | 00027500 |
| | GO TO 51 | 00027600 |
| 33 | CALL ARRDES(D1,HN,H3,H2D,X1,X2,Y1,Y2,DXP,DYP,X1A,X2A,Y1A,Y2A,XS,YS | 00027700 |
| | 1,XS1,XS2,YS1,YS2,AS,R,C,XS3,YS3) | 00027800 |
| | XX1A=X1A | 00027900 |
| | XX2A=X2A | 00028000 |
| | YY1A=Y1A | 00028100 |
| | YY2A=Y2A | 00028200 |
| 51 | CALL AROHD (XX1A,YY1A,XX2A,YY2A,0.16,0.05,11) | 00028300 |
| | IF (.NOT.MEC) GO TO 52 | 00028400 |
| | IF (SHIFT) GO TO 53 | 00028500 |
| | CALL STRICH (X2,Y2,OX,OY,8) | 00028600 |
| | GO TO 52 | 00028700 |
| 53 | CALL PLOTG (XX2A,YY2A,3) | 00028800 |
| | CALL PLOTG (OX,OY,2) | 00028900 |
| C | | 00029000 |

Abb.7(5)

```

C *****00029100
C 00029200
C DESIGNATION. 00029300
C 00029400
C *****00029500
52 IF (SHIFT .AND. (DXP-DDX) .LE. 0.01) GO TO 56 00029600
    DYP=ABS(DYP) 00029700
    IF (DXP .GT. DDX .OR. DYP .GT. DDX) D1=2.*D1 00029800
56 CALL ARRDES(D1,HN,H3,H2D,X1,X2,Y1,Y2,DXP,DYP,X1A,X2A,Y1A,Y2A,XS,YS) 00029900
    1,XS1,XS2,YS1,YS2,AS,R,1,XS3,YS3) 00030000
    CALL NUMBER(XS,YS, H2,FLOAT(DESCR(I)),AS,-1) 00030100
    XS=XS+0.01 00030200
    YS=YS+0.01 00030300
    CALL NUMBER(XS,YS, H2,FLOAT(DESCR(I)),AS,-1) 00030400
    CALL SYMBOL (XS1,YS1,H4,NAMED1,AS,8) 00030500
    IF (D1 .LT. 0.5) GO TO 54 00030600
    CALL SYMBOL (999.0,999.0,H4,NAMED2,AS,12) 00030700
    CALL NUMBER (XS2,YS2,H4,DAUER(I),AS,1) 00030800
    GO TO 55 00030900
54 CALL SYMBOL (XS2,YS2,H4,NAMED2,AS,12) 00031000
    CALL NUMBER (XS3,YS3,H4,DAUER(I),AS,1) 00031100
55 CALL SYMBOL (999.0,999.0,H4,3HW. ,AS,3) 00031200
    CALL SYMBOL (999.0,999.0,H4,ABT,AS,5) 00031300
    IF (.NOT. SHIFT) GO TO 32 00031400
    Y2=0Y 00031500
C 00031600
C *****00031700
C 00031800
C SECOND SYMBOL. 00031900
C 00032000
C 32 IF (AP(I2)) GO TO 6 00032100
    CALL PLOT(X2,Y2,3) 00032200
    CALL SYMBOL(X2,Y2,H2 ,1,0.,-1) 00032300
    AP(I2)=.TRUE. 00032400
    6 STR(I2)=.FALSE. 00032500
    5 CONTINUE 00032600
C *****00032700
C 00032800
C STRICHEL-LINIEN = SCHEINTAETIGKEITEN DEFINIEREN U. ZEICHNEN 00032900
C 00033000
C *****00033100
    DO 22 I=1,NE 00033200
    IF (.NOT. STR(I)) GO TO 22 00033300
    V2=X(I) 00033400
    V1=V2 00033500
    W2=Y(I) 00033600
    WM=Y(I) 00033700
C SUCHE OB NOCH ANDERE SCHEINTAETIGKEITEN UEBER DER ABSZISSE X(I) ZU 00033800
C ZEICHNEN SIND, WENN JA SUCHE NACH DEM MAX. Y-WERT DER SCHEINTAETIGK. 00033900
C =W2, UND SUCHE NACH DEM MINIMALEN Y-WERT DER SCHEINTAETIGKEIT =WM 00034000
    K=I+1 00034100
    DO 23 J=K,NE 00034200
    IF (.NOT. STR(J)) GO TO 23 00034300
    ABSDX=ABS(X(J)-V2) 00034400
    IF (ABSDX .GT. 0.001) GO TO 23 00034500
    IF (Y(J).GT.W2) W2=Y(J) 00034600
    IF ( Y(J) .LT. WM) WM=Y(J) 00034700
    STR(J)=.FALSE. 00034800

```

Abb. 7 (6)

```

23 CONTINUE
C
C
C W2 AN DER STELLE X(I) IST GEFUNDEN
C AN DER STELLE X(I)=V2 GIBT ES EINEN UND NUR EINEN PUNKT MIT DER
C ORDINATE Y(IE) AN DEM 2 ECHE TAEITIGKEITEN ZUSAMMENLAUFEN.
C SUCHE NACH Y(IE). FUER Y(IE) GILT X(IE)=X(I)=V1=V2
C
    W1=-1.
    DO 24 IE=1,NE
    IF(STR(IE)) GO TO 24
    ABSXI=ABS(X(IE)-V2)
    IF (ABSXI .GT. 0.001) GO TO 24
    W1=Y(IE)
    DELTW=ABS(W2-W1)
    IF (DELTW .GT. 0.01) GO TO 25
24 CONTINUE
    GO TO 22
25 IF (W1 .GT. W2) W2=W1
    CALL STRICH (V1,W1,V2,W2,4)
    IF (W1 .LT. W2) GO TO 22
    CALL STRICH (V1,W1,V2,W2,4)
22 CONTINUE
C
C *****
C DEFINE NEW ORIGIN FOR NEXT GRAPH
C *****
C NEUE VERSION: DISTX= MAX. AUSDEHNUNG DES GRAPHEN
    DISTX=X(NE)*DX
C CALL PLOTG (DX,0.,-3) ALTE VERSION MIT DX= MAX. AUSD. D. GRAPHEN
    CALL PLOTG (DISTX,-1.5,-3)
    GO TO 1
99 CALL PLOTG( 5.,0.,-3)
    CALL PLOTG( 0.,DY,2)
    CALL PLOTG( 2.,0.,3)
C CALL PLOTG (0.,0.00,999) ABSCHLUSSAUFRUF IM HAUPTPROGRAMM
    WRITE(6,6000)
C *****
C
C
101 FORMAT(15A4)
102 FORMAT( 2A4)
C *****
C
103 FORMAT (3A4,5A1,5A4,3X,A4,4A2)
C IM FORMAT 103 SIND 3X ERFORDERLICH, UM FUER DIE VARIABLE 'DAUER'
C DIE RICHTIGE POSITIONIERUNG - FULLWORD- GRENZE ZU ERHALTEN BEIM
C LESEN DES VOM PL/1 PROGRAMM ANGELEGTE DATA-SETS AUF FTO1F001 :PGRAPHC
C *****
104 FORMAT (' DESCR(' ,I4,' )=' ,I5,' ** FRUEHESTER-BEGINN=' ,I2,' .',I2,' .',
1',I2,' BEI X1=' ,I4)
200 FORMAT('1')
201 FORMAT(' ',13A4)
202 FORMAT(' NE=' ,G7,' NT=' ,G7)
203 FORMAT(' U(' ,G4,' )=' ,F10.5,5X,' Z(' ,G4,' )=' ,F10.5)

```

Abb.7(7)

```

213 FORMAT(' X(',G4,')=',F10.5,5X,'Y(',G4,')=',F10.5) 00040700
204 FORMAT(' A(',G4,',1) =',G4,1X,',A(',G4,',2) =',G4,2X,',DESCR(',G4,00040800
1') =',G5,' ABTLG=',5A1,' ARBEIT: ',5A4,' DAUER: ',F5.1,' W. ,T=' 00040900
2,G2,' START:',I2,',',I2,',',I2) 00041000
301 FORMAT (17X,5A4,8X) 00041100
6000 FORMAT(' SUCCESSFUL PLOTTING.') 00041200
7000 FORMAT(' DESCR(',G4,')=',G5,' SHIFT=',L2,5X,' I1=',G4,', I2=',G4,5X00041300
1,'X1=',F10.5,' X2=',F10.5,5X,'Y1=',F10.5,' Y2=',F10.5,'ABT=',A8)00041400
C 00041500
C 00041600
RETURN 00041700
END 00041800
SUBROUTINE SC(V,N,DV) 00041900
DIMENSION V(1) 00042000
VMAX=V(1) 00042100
VMIN=VMAX 00042200
DO 1 I=2,N 00042300
IF(V(I).GT.VMAX) VMAX=V(I) 00042400
IF(V(I).LT.VMIN) VMIN=V(I) 00042500
1 CONTINUE 00042600
SV=(VMAX-VMIN)/DV 00042700
DO 2 I=1,N 00042800
2 V(I)=V(I)/SV 00042900
RETURN 00043000
END 00043100
SUBROUTINE ARRDES (D1,HN,H3,H2D,X1,X2,Y1,Y2,DXP,DYP,X1A,X2A,Y1A,Y200043200
1A,XS,YS,XS1,XS2,YS1,YS2,AS,R,NN,XS3,YS3) 00043300
DXP=X2-X1 00043400
DYP=Y2-Y1 00043500
IF (ABS(DXP) .LT. 0.001) GO TO 10 00043600
A1=ATAN(DYP/DXP) 00043700
AS=A1*57.2958 00043800
GO TO 11 00043900
10 AS=90. 00044000
A1=1.570797 00044100
11 SI=SIN(A1) 00044200
CO=COS(A1) 00044300
IF (NN .GT. 0) GO TO 111 00044400
X1A=X1+R*CO 00044500
Y1A=Y1+R*SI 00044600
X2A=X2-R*CO 00044700
Y2A=Y2-R*SI 00044800
111 IF (NN .EQ. 0) GO TO 112 00044900
XS= .5*(X1+X2) - H3*SI -H2D * CO 00045000
YS=.5*(Y1+Y2) +H3*CO -H2D *SI 00045100
HN1=HN*SI 00045200
HN2=HN*CO 00045300
XS1=.5*(X1+X2)-D1*CO-HN1 00045400
XS2=XS1-HN1 00045500
YS1=.5*(Y1+Y2)-D1*SI+HN2 00045600
YS2=YS1+HN2 00045700
XS3=XS2-HN1 00045800
YS3=YS2+HN2 00045900
112 RETURN 00046000
END 00046100
SUBROUTINE STRICH (VV1,WW1,VV2,WW2,L) 00046200
DIMENSION V1(120),V2(120),W1(120),W2(120) 00046300
DV=VV2-VV1 00046400

```

Abb. 7 (8)

| | |
|----------------------------|----------|
| DW=WW2-WW1 | 00046500 |
| N=ABS(DW)+1 | 00046600 |
| NV=ABS(DV)+1 | 00046700 |
| IF (N .LT. NV) N=NV | 00046800 |
| M=N*L+1 | 00046900 |
| DELV=DV/M | 00047000 |
| DELW=DW/M | 00047100 |
| V1(1)=VV1 | 00047200 |
| W1(1)=WW1 | 00047300 |
| CALL PLOTG (VV1,WW1,3) | 00047400 |
| K=2+M/2 | 00047500 |
| DO 1 I=2,K | 00047600 |
| V1(I)=V1(I-1)+DELV | 00047700 |
| W1(I)=W1(I-1)+DELW | 00047800 |
| CALL PLOTG (V1(I),W1(I),2) | 00047900 |
| V1(I)=V1(I)+DELV | 00048000 |
| W1(I)=W1(I)+DELW | 00048100 |
| CALL PLOTG (V1(I),W1(I),3) | 00048200 |
| 1 CONTINUE | 00048300 |
| RETURN | 00048400 |
| END | 00048500 |

| | |
|---|----------|
| * DIE ALS LOGICAL * 1 DEFINIERTEN ZEICHEN VON ABT5 WERDEN IN | 00000100 |
| * DAS FULLWORD ABT (IM FORTRAN PROGRAMM SO DEFINIERT) GEMOVED | 00000200 |
| * | 00000300 |
| * | 00000400 |
| ABT5 CSECT | 00000500 |
| * | 00000600 |
| SAVE (14,12),,* | 00000700 |
| LR 3,13 | 00000800 |
| LA 13,RGS-ABT5(15) | 00000900 |
| USING RGS,13 | 00001000 |
| B BEG | 00001100 |
| RGS DC 18F'0' | 00001200 |
| BEG EQU * | 00001300 |
| ST 13,8(0,3) FORWARD POINTER | 00001400 |
| ST 3,4(0,13) BACKWARD POINTER | 00001500 |
| LM 2,7,0(1) | 00001600 |
| MVC 0(8,7),BLANK | 00001700 |
| MVC 0(1,7),0(2) | 00001800 |
| MVC 1(1,7),0(3) | 00001900 |
| MVC 2(1,7),0(4) | 00002000 |
| MVC 3(1,7),0(5) | 00002100 |
| MVC 4(1,7),0(6) | 00002200 |
| L 13,RGS+4 | 00002300 |
| RETURN (14,12),T | 00002400 |
| BLANK DC C' | 00002500 |
| END | 00002600 |

Abb. 7 (9)

```

//DVZ412BP JOB (0412,541,P0000),FETTE,MSGLEVEL=(1,1),REGION=300K
/*SETUP DEVICE=TAPE9, ID=(P4128P, RING, SAVE, NL), DDNAME=PLOTTAPE
//NETZPLAN EXEC PGM=TERM2, REGION=300K, PARM='NETZPLAN=3FACH'
//STEPLIB DD DSN=LOAD.DVZ, UNIT=2314, VOL=SER=GFKLIB, DISP=SHR
//GRAPH DD DSN=&&GRAPH, UNIT=SYSDA, DISP=(NEW, PASS), SPACE=(6404, (50)),
// DCB=(RECFM=VB, LRECL=64, BLKSIZE=6404)
//PLOTTAPE DD UNIT=TAPE9, LABEL=(, NL), DISP=(NEW, DELETE)
//FT06F001 DD SYSOUT=A, DCB=(RECFM=FBA, LRECL=133, BLKSIZE=931),
// SPACE=(TRK, 100)
//SYSPRINT DD SYSOUT=A, DCB=(RECFM=VBA, LRECL=133, BLKSIZE=935),
// SPACE=(935, 4000)
//FT01F001 DD DSN=*.GRAPH, UNIT=SYSDA, DISP=(OLD, DELETE),
// VOLUME=REF=*.GRAPH
//SYSIN DD *
* NETZPLAN - TESTZEICHNUNG
210 @ELEKTARBEIT NR. 1 @ 10. 1 100 1
220 @ELEKTARBEIT NR. 2 @ 20. 1 100 1
230 @ELEKTARBEIT NR. 3 @ 30. 1 100 1
10 @ELEKTARBEIT NR.4 @ 83. 3 210 220 230 1
20 @BETONARBEIT NR.5 @ 27. 2 110 100 1
30 @PLAN ARBEIT NR. 6 @ 19.0 2 20 10 1
40 @BETONARBEIT NR. 7 @ 73. 2 50 44 1
50 @BETONARBEIT NR. 8 @ 22. 1 60 1
70 @PLAN ARBEIT NR. 9 @ 37. 3 80 30 90 1
80 @PLAN ARBEIT NR. 10 @ 35. 3 50 10 40 1
90 @AUSH ARBEIT NR. 11 @ 5. 3 110 20 300 1
100 @ABN ARBEIT NR. 12 @ 10. 1 60 1
110 @ELEKTARBEIT NR. 13 @ 25. 1 60 1
44 @ELEKTARBEIT NR. 14 @ 5. 1 60 1
300 @BETONARBEIT NR. 15 @ 4.0 2 50 110 1
60 @ABN ARBEIT NR. 16 @ 5.0 0 0
/*
//

```

Eingabebeispiel mit Steuerkarten Abb. 8

TERM2

EINGABE:
NETZPLAN - TESTZEICHNUNG

29 9 71 0 0 0 29 4 71

TAETIGKEITEN:

| | | | | |
|-----|-------|--------|--------|------|
| 210 | ELEKT | ARBEIT | NR. 1 | 10.0 |
| 1 | 100 | | | |
| 220 | ELEKT | ARBEIT | NR. 2 | 20.0 |
| 1 | 100 | | | |
| 230 | ELEKT | ARBEIT | NR. 3 | 30.0 |
| 1 | 100 | | | |
| 10 | ELEKT | ARBEIT | NR.4 | 83.0 |
| 3 | 210 | 220 | 230 | |
| 20 | BETON | ARBEIT | NR.5 | 27.0 |
| 2 | 110 | 100 | | |
| 30 | PLAN | ARBEIT | NR. 6 | 19.0 |
| 2 | 20 | 10 | | |
| 40 | BETON | ARBEIT | NR. 7 | 73.0 |
| 2 | 50 | 44 | | |
| 50 | BETON | ARBEIT | NR. 8 | 22.0 |
| 1 | 60 | | | |
| 70 | PLAN | ARBEIT | NR. 9 | 37.0 |
| 3 | 80 | 30 | 90 | |
| 80 | PLAN | ARBEIT | NR. 10 | 35.0 |
| 3 | 50 | 10 | 40 | |
| 90 | AUSH | ARBEIT | NR. 11 | 5.0 |
| 3 | 110 | 20 | 300 | |
| 100 | ABN | ARBEIT | NR. 12 | 10.0 |
| 1 | 60 | | | |
| 110 | ELEKT | ARBEIT | NR. 13 | 25.0 |
| 1 | 60 | | | |
| 44 | ELEKT | ARBEIT | NR. 14 | 5.0 |
| 1 | 60 | | | |
| 300 | BETON | ARBEIT | NR. 15 | 4.0 |
| 2 | 50 | 110 | | |
| 60 | ABN | ARBEIT | NR. 16 | 5.0 |
| 0 | | | | |

Liste der Eingabekarten Abb. 9

N_DAY= 1001;

| | | | | | | | | |
|------|------|------|------------|----|----|----|-----|---|
| 1 | | 1 | MITTWOCH | 29 | 9 | 71 | 272 | 3 |
| 2 | | 2 | DONNERSTAG | 30 | 9 | 71 | 273 | |
| 3 | | 3 | FREITAG | 1 | 10 | 71 | 274 | |
| 4 | FREI | 3 | SAMSTAG | 2 | 10 | 71 | 275 | |
| 5 | FREI | 3 | SONNTAG | 3 | 10 | 71 | 276 | |
| 6 | | 4 | MONTAG | 4 | 10 | 71 | 277 | |
| 7 | | 5 | DIENSTAG | 5 | 10 | 71 | 278 | |
| 8 | | 6 | MITTWOCH | 6 | 10 | 71 | 279 | |
| 9 | | 7 | DONNERSTAG | 7 | 10 | 71 | 280 | |
| 10 | | 8 | FREITAG | 8 | 10 | 71 | 281 | |
| 11 | FREI | 8 | SAMSTAG | 9 | 10 | 71 | 282 | |
| 12 | FREI | 8 | SONNTAG | 10 | 10 | 71 | 283 | |
| 13 | | 9 | MONTAG | 11 | 10 | 71 | 284 | |
| 14 | | 10 | DIENSTAG | 12 | 10 | 71 | 285 | |
| 15 | | 11 | MITTWOCH | 13 | 10 | 71 | 286 | |
| 16 | | 12 | DONNERSTAG | 14 | 10 | 71 | 287 | |
| 17 | | 13 | FREITAG | 15 | 10 | 71 | 288 | |
| 18 | FREI | 13 | SAMSTAG | 16 | 10 | 71 | 289 | |
| 19 | FREI | 13 | SONNTAG | 17 | 10 | 71 | 290 | |
| 20 | | 14 | MONTAG | 18 | 10 | 71 | 291 | |
| 21 | | 15 | DIENSTAG | 19 | 10 | 71 | 292 | |
| 22 | | 16 | MITTWOCH | 20 | 10 | 71 | 293 | |
| 23 | | 17 | DONNERSTAG | 21 | 10 | 71 | 294 | |
| : | | : | : | : | : | : | : | |
| : | | : | : | : | : | : | : | |
| : | | : | : | : | : | : | : | |
| 1439 | FREI | 980 | SAMSTAG | 6 | 9 | 75 | 249 | |
| 1440 | FREI | 980 | SONNTAG | 7 | 9 | 75 | 250 | |
| 1441 | | 981 | MONTAG | 8 | 9 | 75 | 251 | |
| 1442 | | 982 | DIENSTAG | 9 | 9 | 75 | 252 | |
| 1443 | | 983 | MITTWOCH | 10 | 9 | 75 | 253 | |
| 1444 | | 984 | DONNERSTAG | 11 | 9 | 75 | 254 | |
| 1445 | | 985 | FREITAG | 12 | 9 | 75 | 255 | |
| 1446 | FREI | 985 | SAMSTAG | 13 | 9 | 75 | 256 | |
| 1447 | FREI | 985 | SONNTAG | 14 | 9 | 75 | 257 | |
| 1448 | | 986 | MONTAG | 15 | 9 | 75 | 258 | |
| 1449 | | 987 | DIENSTAG | 16 | 9 | 75 | 259 | |
| 1450 | | 988 | MITTWOCH | 17 | 9 | 75 | 260 | |
| 1451 | | 989 | DONNERSTAG | 18 | 9 | 75 | 261 | |
| 1452 | | 990 | FREITAG | 19 | 9 | 75 | 262 | |
| 1453 | FREI | 990 | SAMSTAG | 20 | 9 | 75 | 263 | |
| 1454 | FREI | 990 | SONNTAG | 21 | 9 | 75 | 264 | |
| 1455 | | 991 | MONTAG | 22 | 9 | 75 | 265 | |
| 1456 | | 992 | DIENSTAG | 23 | 9 | 75 | 266 | |
| 1457 | | 993 | MITTWOCH | 24 | 9 | 75 | 267 | |
| 1458 | | 994 | DONNERSTAG | 25 | 9 | 75 | 268 | |
| 1459 | | 995 | FREITAG | 26 | 9 | 75 | 269 | |
| 1460 | FREI | 995 | SAMSTAG | 27 | 9 | 75 | 270 | |
| 1461 | FREI | 995 | SONNTAG | 28 | 9 | 75 | 271 | |
| 1462 | | 996 | MONTAG | 29 | 9 | 75 | 272 | |
| 1463 | | 997 | DIENSTAG | 30 | 9 | 75 | 273 | |
| 1464 | | 998 | MITTWOCH | 1 | 10 | 75 | 274 | |
| 1465 | | 999 | DONNERSTAG | 2 | 10 | 75 | 275 | |
| 1466 | | 1000 | FREITAG | 3 | 10 | 75 | 276 | |
| 1467 | FREI | 1000 | SAMSTAG | 4 | 10 | 75 | 277 | |
| 1468 | FREI | 1000 | SONNTAG | 5 | 10 | 75 | 278 | |
| 1469 | | 1001 | MONTAG | 6 | 10 | 75 | 279 | |

KEINE FEHLER GEFUNDEN.

Beispiel für den Kalenderausdruck Abb. 10

PROJEKT. NETZPLAN - TESTZEICHNUNG

-SORT.NACH SL.+R.-

START.29. 9.71

ENDE. 6.10.75

REPORT.29. 4.71

SEITE. 1

| NR. | ABT. | TAETIGKEIT | DAUER | F.BEG. | ASP.END | FS.END | T.PFZ | F.PFZ | NR/WEG | VT.PFZ | FR.BEG. | ASP. END | SP. END |
|-----|-------|---------------|-------|--------|---------|--------|-------|-------|--------|---------|----------|----------|----------|
| 70 | PLAN | ARBEIT NR. 9 | 37.0 | 0.0 | 37.0 | 37.0 | 0.0 | 0.0 | 0 | 0.000 | 29. 9.71 | 30. 6.72 | 30. 6.72 |
| 80 | PLAN | ARBEIT NR. 10 | 35.0 | 37.0 | 72.0 | 72.0 | 0.0 | 0.0 | 1 | 0.000 | 30. 6.72 | 12. 3.73 | 12. 3.73 |
| 10 | ELEKT | ARBEIT NR.4 | 83.0 | 72.0 | 155.0 | 155.0 | 0.0 | 0.0 | 2 | 0.000 | 12. 3.73 | 6.11.74 | 6.11.74 |
| 230 | ELEKT | ARBEIT NR. 3 | 30.0 | 155.0 | 185.0 | 185.0 | 0.0 | 0.0 | 3 | 0.000 | 6.11.74 | 23. 6.75 | 23. 6.75 |
| 100 | ABN | ARBEIT NR. 12 | 10.0 | 185.0 | 195.0 | 195.0 | 0.0 | 0.0 | 4 | 0.000 | 23. 6.75 | 1. 9.75 | 1. 9.75 |
| 60 | ABN | ARBEIT NR. 16 | 5.0 | 195.0 | 200.0 | 200.0 | 0.0 | 0.0 | 5 | 0.000 | 1. 9.75 | 6.10.75 | 6.10.75 |
| 220 | ELEKT | ARBEIT NR. 2 | 20.0 | 155.0 | 185.0 | 185.0 | 10.0 | 10.0 | 0 | 10.000 | 6.11.74 | 23. 6.75 | 23. 6.75 |
| 30 | PLAN | ARBEIT NR. 6 | 19.0 | 37.0 | 72.0 | 72.0 | 16.0 | 0.0 | 0 | 16.000 | 30. 6.72 | 12. 3.73 | 12. 3.73 |
| 210 | ELEKT | ARBEIT NR. 1 | 10.0 | 155.0 | 185.0 | 185.0 | 20.0 | 20.0 | 0 | 20.000 | 6.11.74 | 23. 6.75 | 23. 6.75 |
| 40 | BETON | ARBEIT NR. 7 | 73.0 | 72.0 | 173.0 | 159.0 | 28.0 | 0.0 | 0 | 14.000 | 12. 3.73 | 24. 3.75 | 4.12.74 |
| 50 | BETON | ARBEIT NR. 8 | 22.0 | 145.0 | 195.0 | 181.0 | 28.0 | 28.0 | 1 | 14.000 | 28. 8.74 | 1. 9.75 | 1. 9.75 |
| 44 | ELEKT | ARBEIT NR. 14 | 5.0 | 145.0 | 195.0 | 195.0 | 45.0 | 45.0 | 0 | 45.000 | 28. 8.74 | 1. 9.75 | 1. 9.75 |
| 20 | BETON | ARBEIT NR.5 | 27.0 | 56.0 | 170.0 | 126.5 | 87.0 | 0.0 | 0 | 43.500 | 10.11.72 | 3. 3.75 | 17. 4.74 |
| 110 | ELEKT | ARBEIT NR. 13 | 25.0 | 83.0 | 195.0 | 151.5 | 87.0 | 87.0 | 1 | 43.500 | 1. 6.73 | 1. 9.75 | 1. 9.75 |
| 90 | AUSH | ARBEIT NR. 11 | 5.0 | 37.0 | 143.0 | 143.0 | 101.0 | 0.0 | 0 | 101.000 | 30. 6.72 | 14. 8.74 | 14. 8.74 |
| 300 | BETON | ARBEIT NR. 15 | 4.0 | 42.0 | 170.0 | 170.0 | 124.0 | 37.0 | 0 | 124.000 | 4. 8.72 | 3. 3.75 | 3. 3.75 |

Ausdruck der Wege durch den Netzplan geordnet nach ansteigender totaler Pufferzeit Abb. 11

PROJEKT. NETZPLAN - TESTZEICHNUNG

-ZUSTAENDIGKEITEN-

START.29. 9.71

ENDE. 6.10.75

REPORT.29. 4.71

SEITE. 4

AUFGABENBEREICH. ELEKT

| NR. | ABT. | TAETIGKEIT | DAUER | FRUEH.BEGINN | SOLL.BEGINN | SOLL - ENDE | VT.PFZ. |
|-----|-------|---------------|-------|--------------|-------------|-------------|---------|
| 10 | ELEKT | ARBEIT NR.4 | 83.0 | 12. 3.73 | 12. 3.73 | 6.11.74 | 0.000 |
| 230 | ELEKT | ARBEIT NR. 3 | 30.0 | 6.11.74 | 6.11.74 | 23. 6.75 | 0.000 |
| 220 | ELEKT | ARBEIT NR. 2 | 20.0 | 6.11.74 | 6.11.74 | 9. 4.75 | 10.000 |
| 210 | ELEKT | ARBEIT NR. 1 | 10.0 | 6.11.74 | 6.11.74 | 27. 1.75 | 20.000 |
| 44 | ELEKT | ARBEIT NR. 14 | 5.0 | 28. 8.74 | 28. 8.74 | 2.10.74 | 45.000 |
| 110 | ELEKT | ARBEIT NR. 13 | 25.0 | 1. 6.73 | 17. 4.74 | 14.10.74 | 43.500 |

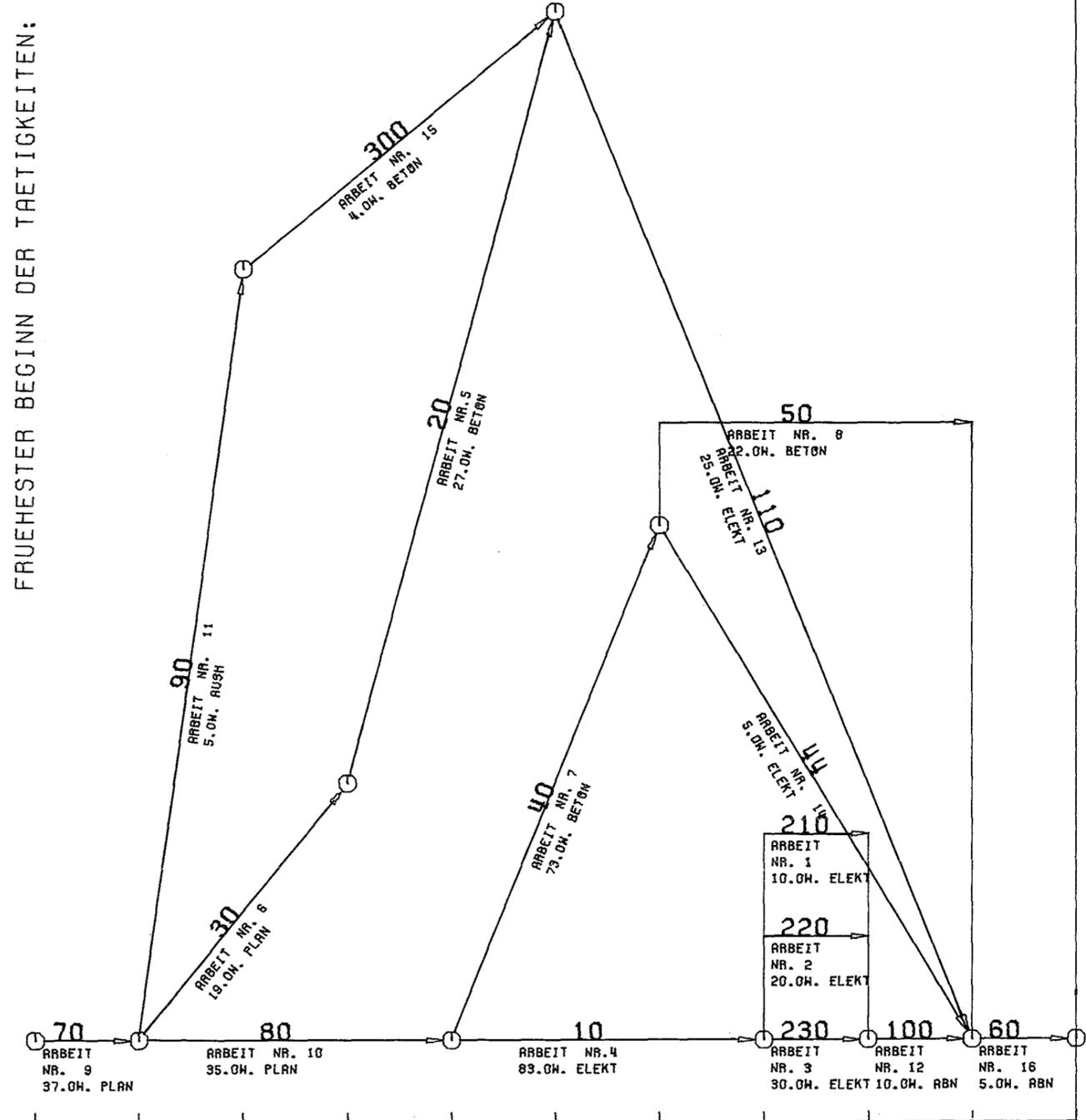
Beispiel für den Ausdruck geordnet nach Zuständigkeiten Abb.12

DVZ412BP

232. TAG 1971 11UHR46

DAUER (ARBEITSWÖCHEN)
NR. DER TÄTIGKEITEN

0.0 37.0 42.0 56.0 72.0 83.0 145.0 155.0 185.0 195.0 200.0
 29.9.71 30.6.72 4.8.72 10.11.72 12.3.73 1.6.73 28.8.74 6.11.74 23.6.75 1.9.75



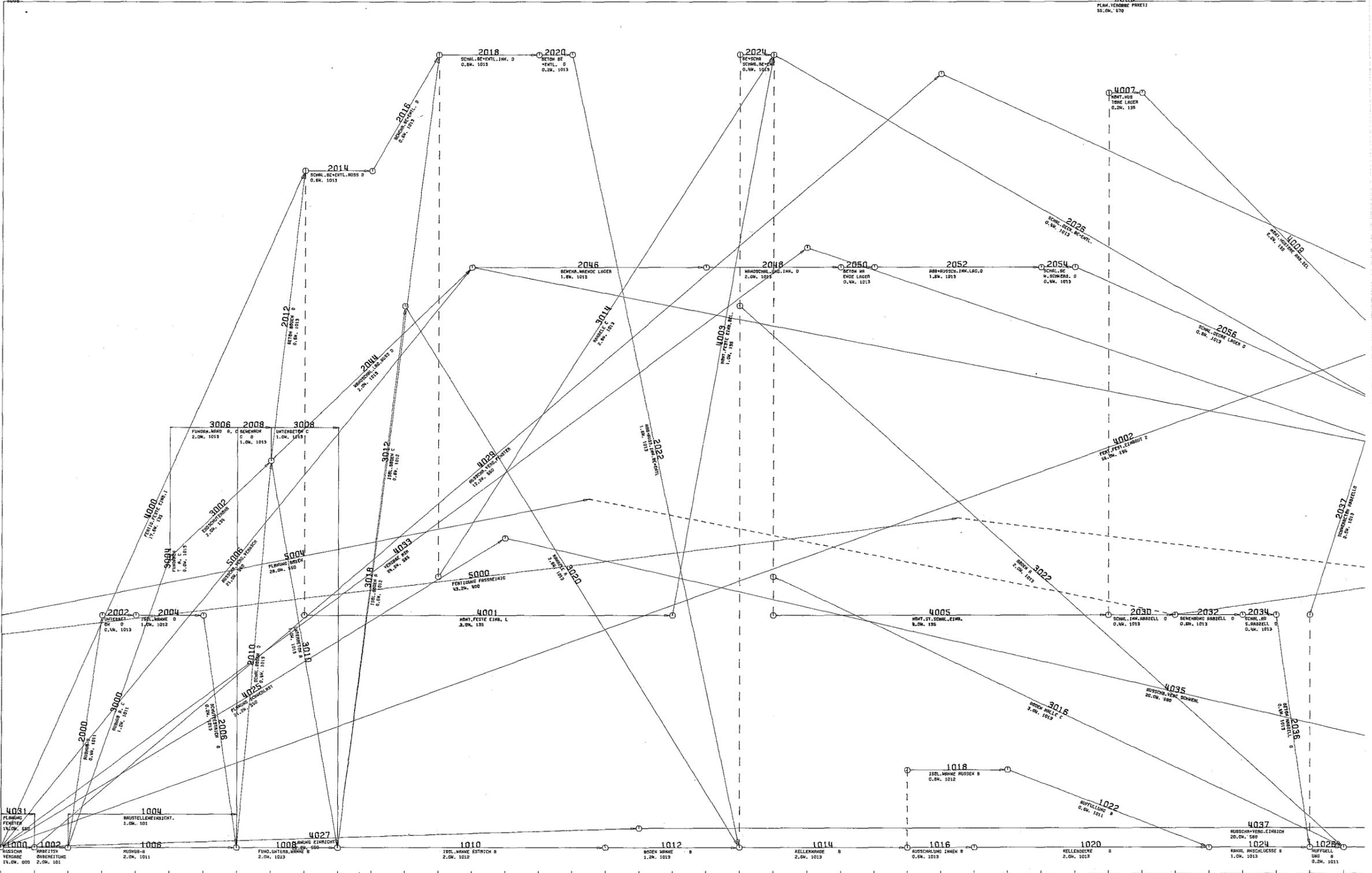
NETZPLAN - TESTZEICHNUNG

Abb. 13(1)

DAUER (ARBEITSWOCHEN)
NR. DER TÄTIGKEITEN

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------|--------|---------|---------|--------|--------|--------|---------|---------|---------|---------|--------|--------|--------|--------|--------|---------|---------|---------|---------|---------|---------|---------|---------|---------|--------|--------|--------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|--------|--------|--------|------|------|------|------|------|------|------|------|
| 0.0 | 14.0 | 28.0 | 30.0 | 30.4 | 30.8 | 31.0 | 31.8 | 32.0 | 33.0 | 33.6 | 34.0 | 34.2 | 34.6 | 34.8 | 35.0 | 35.2 | 35.4 | 35.6 | 36.0 | 36.2 | 36.6 | 36.8 | 37.2 | 37.6 | 38.2 | 38.8 | 39.2 | 39.6 | 40.2 | 40.4 | 40.6 | 40.8 | 41.2 | 41.6 | 41.8 | 42.0 | 42.4 | 42.6 | 43.0 | 43.4 | 43.6 | | | | | | | | |
| 1001 | 1002 | 1003 | 1004 | 1005 | 1006 | 1007 | 1008 | 1009 | 1010 | 1011 | 1012 | 1013 | 1014 | 1015 | 1016 | 1017 | 1018 | 1019 | 1020 | 1021 | 1022 | 1023 | 1024 | 1025 | 1026 | 1027 | 1028 | 1029 | 1030 | 1031 | 1032 | 1033 | 1034 | 1035 | 1036 | 1037 | 1038 | 1039 | 1040 | 1041 | 1042 | 1043 | 1044 | 1045 | 1046 | 1047 | 1048 | 1049 | 1050 |
| 18.9.69 | 3.1.69 | 16.4.69 | 30.4.69 | 5.5.69 | 7.5.69 | 8.5.69 | 14.5.69 | 16.5.69 | 23.5.69 | 29.5.69 | 2.6.69 | 3.6.69 | 5.6.69 | 6.6.69 | 9.6.69 | 10.6.69 | 11.6.69 | 12.6.69 | 16.6.69 | 17.6.69 | 19.6.69 | 20.6.69 | 24.6.69 | 26.6.69 | 1.7.69 | 4.7.69 | 8.7.69 | 11.7.69 | 15.7.69 | 16.7.69 | 17.7.69 | 18.7.69 | 22.7.69 | 24.7.69 | 25.7.69 | 28.7.69 | 30.7.69 | 31.7.69 | 4.8.69 | 6.8.69 | 7.8.69 | | | | | | | | |

FRÜHESTER BEGINN DER TÄTIGKEITEN:



BAU 536 FERAB ANLAGE NETZWERKPLAN ABB. 13 (2)

NETZPLAN - TESTZEICHNUNG

NE= 11 NT= 16

| | |
|---|------------------------------------|
| A(1,1) = 1 ,A(1,2) = 2 ,DESCR(1) = 70 ABTLG=PLAN ARBEIT: ARBEIT NR. 9 | DAUER: 37.0 W. ,T= 0START:29. 9.71 |
| A(2,1) = 2 ,A(2,2) = 5 ,DESCR(2) = 80 ABTLG=PLAN ARBEIT: ARBEIT NR. 10 | DAUER: 35.0 W. ,T= 0START:30. 6.72 |
| A(3,1) = 5 ,A(3,2) = 8 ,DESCR(3) = 10 ABTLG=ELEKT ARBEIT: ARBEIT NR. 4 | DAUER: 83.0 W. ,T= 0START:12. 3.73 |
| A(4,1) = 8 ,A(4,2) = 9 ,DESCR(4) = 230 ABTLG=ELEKT ARBEIT: ARBEIT NR. 3 | DAUER: 30.0 W. ,T= 0START: 6.11.74 |
| A(5,1) = 9 ,A(5,2) = 10 ,DESCR(5) = 100 ABTLG=ABN ARBEIT: ARBEIT NR. 12 | DAUER: 10.0 W. ,T= 0START:23. 6.75 |
| A(6,1) = 10 ,A(6,2) = 11 ,DESCR(6) = 60 ABTLG=ABN ARBEIT: ARBEIT NR. 16 | DAUER: 5.0 W. ,T= 0START: 1. 9.75 |
| A(7,1) = 2 ,A(7,2) = 3 ,DESCR(7) = 90 ABTLG=AUSH ARBEIT: ARBEIT NR. 11 | DAUER: 5.0 W. ,T= 1START:30. 6.72 |
| A(8,1) = 2 ,A(8,2) = 4 ,DESCR(8) = 30 ABTLG=PLAN ARBEIT: ARBEIT NR. 6 | DAUER: 19.0 W. ,T= 1START:30. 6.72 |
| A(9,1) = 8 ,A(9,2) = 9 ,DESCR(9) = 220 ABTLG=ELEKT ARBEIT: ARBEIT NR. 2 | DAUER: 23.0 W. ,T= 1START: 6.11.74 |
| A(10,1) = 8 ,A(10,2) = 9 ,DESCR(10) = 210 ABTLG=ELEKT ARBEIT: ARBEIT NR. 1 | DAUER: 10.0 W. ,T= 1START: 6.11.74 |
| A(11,1) = 3 ,A(11,2) = 6 ,DESCR(11) = 300 ABTLG=BETON ARBEIT: ARBEIT NR. 15 | DAUER: 4.0 W. ,T= 1START: 4. 8.72 |
| A(12,1) = 7 ,A(12,2) = 10 ,DESCR(12) = 44 ABTLG=ELEKT ARBEIT: ARBEIT NR. 14 | DAUER: 5.0 W. ,T= 1START:28. 8.74 |
| A(13,1) = 5 ,A(13,2) = 7 ,DESCR(13) = 40 ABTLG=BETON ARBEIT: ARBEIT NR. 7 | DAUER: 73.0 W. ,T= 0START:12. 3.73 |
| A(14,1) = 7 ,A(14,2) = 10 ,DESCR(14) = 50 ABTLG=BETON ARBEIT: ARBEIT NR. 9 | DAUER: 22.0 W. ,T= 0START:28. 8.74 |
| A(15,1) = 4 ,A(15,2) = 6 ,DESCR(15) = 20 ABTLG=BETON ARBEIT: ARBEIT NR.5 | DAUER: 27.0 W. ,T= 0START:10.11.72 |
| A(16,1) = 6 ,A(16,2) = 10 ,DESCR(16) = 110 ABTLG=ELEKT ARBEIT: ARBEIT NR. 13 | DAUER: 25.0 W. ,T= 0START: 1. 6.73 |

| | |
|--------------------|------------------|
| U(1) = 0.0 | Z(1) = 0.0 |
| U(2) = 37.00000 | Z(2) = 0.0 |
| U(3) = 42.00000 | Z(3) = 8.00000 |
| U(4) = 56.00000 | Z(4) = 3.00000 |
| U(5) = 72.00000 | Z(5) = 0.0 |
| U(6) = 83.00000 | Z(6) = 9.00000 |
| U(7) = 145.00000 | Z(7) = 6.00000 |
| U(8) = 155.00000 | Z(8) = 0.0 |
| U(9) = 185.00000 | Z(9) = 0.0 |
| U(10) = 195.00000 | Z(10) = 0.0 |
| U(11) = 200.00000 | Z(11) = 0.0 |
| X(1) = 1.00000 | Y(1) = 0.0 |
| X(2) = 2.00000 | Y(2) = 0.0 |
| X(3) = 3.00000 | Y(3) = 7.50000 |
| X(4) = 4.00000 | Y(4) = 2.50000 |
| X(5) = 5.00000 | Y(5) = 0.0 |
| X(6) = 6.00000 | Y(6) = 10.00000 |
| X(7) = 7.00000 | Y(7) = 5.00000 |
| X(8) = 8.00000 | Y(8) = 0.0 |
| X(9) = 9.00000 | Y(9) = 0.0 |
| X(10) = 10.00000 | Y(10) = 0.0 |
| X(11) = 11.00000 | Y(11) = 0.0 |

| | | | |
|---|--------------|--------------|-----------|
| DESCR(1) = 70 ** FRUEHESTER-BEGINN=29. 9.71 BEI X1= 1 | Y1= 0.0 | Y2= 0.0 | ART=PLAN |
| DESCR(1) = 70 SHIFT= F I1= 1,I2= 2 X1= 1.00000 X2= 2.00000 | Y1= 0.0 | Y2= 0.0 | ART=PLAN |
| DESCR(2) = 80 ** FRUEHESTER-BEGINN=30. 6.72 BEI X1= 2 | Y1= 0.0 | Y2= 0.0 | ART=PLAN |
| DESCR(2) = 80 SHIFT= F I1= 2,I2= 5 X1= 2.00000 X2= 5.00000 | Y1= 0.0 | Y2= 0.0 | ART=PLAN |
| DESCR(3) = 10 ** FRUEHESTER-BEGINN=12. 3.73 BEI X1= 5 | Y1= 0.0 | Y2= 0.0 | ART=ELEKT |
| DESCR(3) = 10 SHIFT= F I1= 5,I2= 8 X1= 5.00000 X2= 8.00000 | Y1= 0.0 | Y2= 0.0 | ART=ELEKT |
| DESCR(4) = 230 ** FRUEHESTER-BEGINN= 6.11.74 BEI X1= 8 | Y1= 0.0 | Y2= 0.0 | ART=ELFKT |
| DESCR(4) = 230 SHIFT= F I1= 8,I2= 9 X1= 8.00000 X2= 9.00000 | Y1= 0.0 | Y2= 0.0 | ART=ELFKT |
| DESCR(5) = 100 ** FRUEHESTER-BEGINN=23. 6.75 BEI X1= 9 | Y1= 0.0 | Y2= 0.0 | ART=ABN |
| DESCR(5) = 100 SHIFT= F I1= 9,I2= 10 X1= 9.00000 X2= 10.00000 | Y1= 0.0 | Y2= 0.0 | ART=ABN |
| DESCR(6) = 60 ** FRUEHESTER-BEGINN= 1. 9.75 BEI X1= 10 | Y1= 0.0 | Y2= 0.0 | ART=ABN |
| DESCR(6) = 60 SHIFT= F I1= 10,I2= 11 X1= 10.00000 X2= 11.00000 | Y1= 0.0 | Y2= 0.0 | ART=ABN |
| DESCR(7) = 90 ** FRUEHESTER-BEGINN=30. 6.72 BEI X1= 2 | Y1= 0.0 | Y2= 7.50000 | ART=AUSH |
| DESCR(7) = 90 SHIFT= F I1= 2,I2= 3 X1= 2.00000 X2= 3.00000 | Y1= 0.0 | Y2= 7.50000 | ART=AUSH |
| DESCR(8) = 30 ** FRUEHESTER-BEGINN=30. 6.72 BEI X1= 2 | Y1= 0.0 | Y2= 2.50000 | ART=PLAN |
| DESCR(8) = 30 SHIFT= F I1= 2,I2= 4 X1= 2.00000 X2= 4.00000 | Y1= 0.0 | Y2= 2.50000 | ART=PLAN |
| DESCR(9) = 220 ** FRUEHESTER-BEGINN= 6.11.74 BEI X1= 8 | Y1= 0.0 | Y2= 0.0 | ART=ELEKT |
| DESCR(9) = 220 SHIFT= T I1= 8,I2= 9 X1= 8.00000 X2= 9.00000 | Y1= 0.0 | Y2= 0.0 | ART=ELEKT |
| DESCR(10) = 210 ** FRUEHESTER-BEGINN= 6.11.74 BEI X1= 8 | Y1= 0.0 | Y2= 0.0 | ART=ELEKT |
| DESCR(10) = 210 SHIFT= T I1= 8,I2= 9 X1= 8.00000 X2= 9.00000 | Y1= 0.0 | Y2= 0.0 | ART=ELEKT |
| DESCR(11) = 300 ** FRUEHESTER-BEGINN= 4. 8.72 BEI X1= 3 | Y1= 7.50000 | Y2= 10.00000 | ART=BETON |
| DESCR(11) = 300 SHIFT= F I1= 3,I2= 6 X1= 3.00000 X2= 6.00000 | Y1= 7.50000 | Y2= 10.00000 | ART=BETON |
| DESCR(12) = 44 ** FRUEHESTER-BEGINN=28. 8.74 BEI X1= 7 | Y1= 5.00000 | Y2= 0.0 | ART=ELEKT |
| DESCR(12) = 44 SHIFT= F I1= 7,I2= 10 X1= 7.00000 X2= 10.00000 | Y1= 5.00000 | Y2= 0.0 | ART=ELEKT |
| DESCR(13) = 40 ** FRUEHESTER-BEGINN=12. 3.73 BEI X1= 5 | Y1= 0.0 | Y2= 5.00000 | ART=BETON |
| DESCR(13) = 40 SHIFT= F I1= 5,I2= 7 X1= 5.00000 X2= 7.00000 | Y1= 0.0 | Y2= 5.00000 | ART=BETON |
| DESCR(14) = 50 ** FRUEHESTER-BEGINN=28. 8.74 BEI X1= 7 | Y1= 5.00000 | Y2= 0.0 | ART=BETON |
| DESCR(14) = 50 SHIFT= T I1= 7,I2= 10 X1= 7.00000 X2= 10.00000 | Y1= 5.00000 | Y2= 0.0 | ART=BETON |
| DESCR(15) = 20 ** FRUEHESTER-BEGINN=10.11.72 BEI X1= 4 | Y1= 2.50000 | Y2= 10.00000 | ART=BETON |
| DESCR(15) = 20 SHIFT= F I1= 4,I2= 6 X1= 4.00000 X2= 6.00000 | Y1= 2.50000 | Y2= 10.00000 | ART=BETON |
| DESCR(16) = 110 ** FRUEHESTER-BEGINN= 1. 6.73 BEI X1= 6 | Y1= 10.00000 | Y2= 0.0 | ART=ELEKT |
| DESCR(16) = 110 SHIFT= F I1= 6,I2= 10 X1= 6.00000 X2= 10.00000 | Y1= 10.00000 | Y2= 0.0 | ART=ELEKT |

SUCCESSFUL PLOTTING.