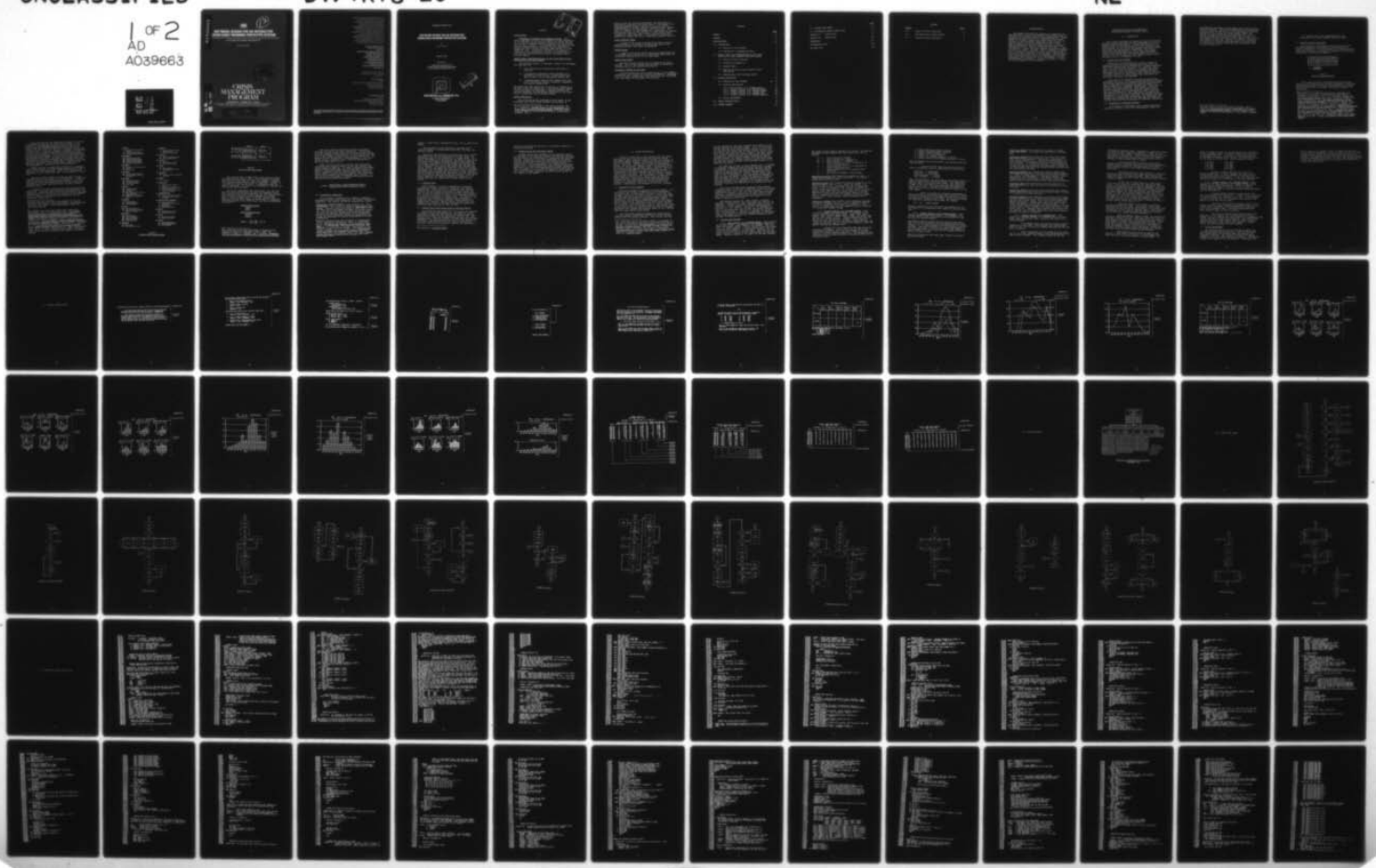
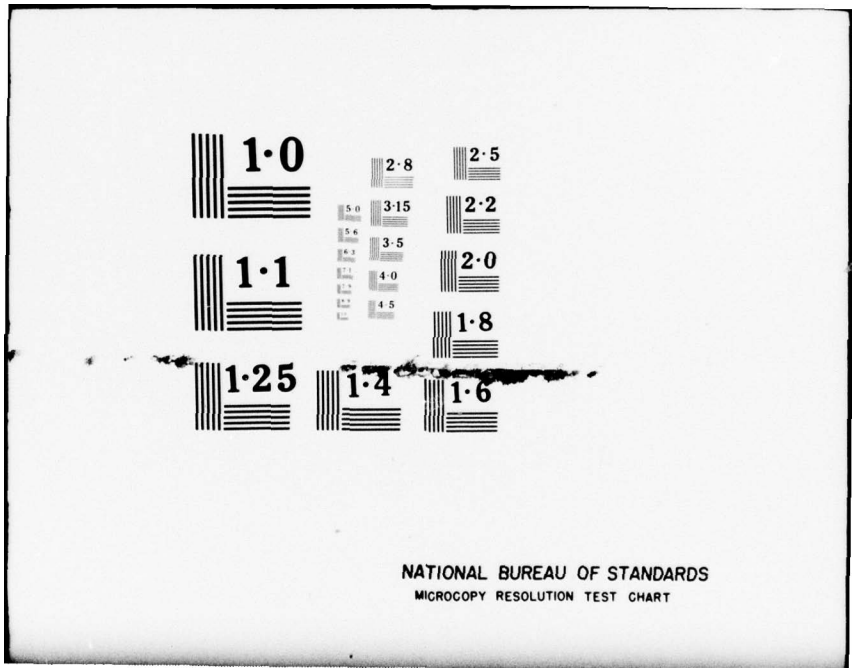
AD-A039 663 DECISIONS AND DESIGNS INC MCLEAN VA F/G 5/2
SOFTWARE DESIGN FOR AN INTERACTIVE CRISIS EARLY WARNING PROTOTY--ETC(U)
DEC 76 J F WITTMEYER N00014-76-C-0712
UNCLASSIFIED DT/TR76-20 NL

1 OF 2
AD
A039663




1·0
1·1
1·25
1·4
1·6
1·8
2·0
2·2
2·5
2·8
3·15
3·5
4·0
4·5
5·0
5·6
6·3
NATIONAL BUREAU OF STANDARDS
MICROCOPY RESOLUTION TEST CHART

AD A039663

SOFTWARE DESIGN FOR AN INTERACTIVE CRISIS EARLY WARNING PROTOTYPE SYSTEM

DECISIONS AND DESIGNS INCORPORATED

James F. Wittmeyer

DDC
RECEIVED
MAY 19 1977
C

CRISIS MANAGEMENT PROGRAM

CYBERNETICS TECHNOLOGY OFFICE
DEFENSE ADVANCED RESEARCH PROJECTS AGENCY
Office of Naval Research • Organizational Effectiveness Research Programs

AU NO.
DDC FILE COPY

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

The objective of the Crisis Management Program is to develop and transfer to users in the Department of Defense advanced technologies and methodologies for crisis warning and decision-making. The technologies and methodologies draw upon research in the social, behavioral, and computer sciences, including quantitative forecasting, decision analysis, and cybernetics. The program is sponsored by the Cybernetics Technology Office of the Defense Advanced Research Projects Agency (ARPA). Technical progress is monitored by the Office of Naval Research – Organizational Effectiveness Research Programs. Current and past participants in the program include:

Analytical Assessments, Inc.
The Brookings Institution
CACI, Inc.
California State College, Dominguez Hills
Carnegie-Mellon University
Decisions and Designs, Inc.
Human Sciences Research, Inc.
The Ohio State University
The RAND Corporation
U.S. Naval Postgraduate School
University of Maryland
University of Southern California
Yale University

Inquiries and comments regarding the program or this report should be addressed to:

Dr. John A. Nagay
Director,
Organizational Effectiveness Research Programs
Office of Naval Research
800 North Quincy Street
Arlington, VA 22217

or

Dr. Robert A. Young
Director,
Cybernetics Technology Office
Defense Advanced Research Projects Agency
1400 Wilson Boulevard
Arlington, VA 22209

The views and conclusions contained in this document are those of the author(s) and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the U.S. Government.

TECHNICAL REPORT 76-20

SOFTWARE DESIGN FOR AN INTERACTIVE CRISIS EARLY WARNING PROTOTYPE SYSTEM

by

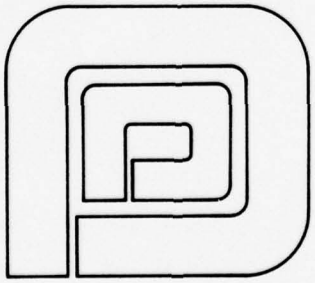
James F. Wittmeyer

December 1976

Sponsored by

Cybernetics Technology Office
Defense Advanced Research Projects Agency
Under Contract No. N00014-76-C-0712

DDC
MAY 19 1977
C



DECISIONS and DESIGNS, INC.
Suite 600, 8400 Westpark Drive
McLean, Virginia 22101
(703) 821-2828

SUMMARY

Introduction

Statement of problem and proposed solution. This report describes in detail one component of the development of an integrated crisis warning system, the design and use of interactive computer software necessary for the development of such a system.[1] The component was developed for the Defense Advanced Research Projects Agency's Cybernetics Technology Office during the period from January 1 to September 30, 1976. Interactive software is an essential component of the development, testing, and evaluation of a user-oriented integrated system and of prototype crisis warning systems.

Design, Test, and Demonstration of the First-Stage Crisis Early Warning Prototype System

The prototype system, as designed, tested, and developed, was comprised of:

(1) five quantitative international political indicators;

(2) a forecasting capability which includes an international crisis object, a short-range retrospective goal, and an associative method; and

(3) a sophisticated computer base endowed with efficient information storage, retrieval, processing, and display capabilities.

The system has been tested for its ability or inability to forecast crises retrospectively. The demonstration system for the prototype is comprised of computer hardware and software necessary for the design, installation, and operation of interactive user-oriented systems.

System Description

The prototype system described in this report is the foundation upon which future systems will be built in

[1] For an overview of progress on the entire project, see S. J. Andriole, Progress Report on the Development of a Crisis Early Warning Prototype System, Technical Report 76-19 (McLean, VA: Decisions and Designs, Incorporated, December, 1976).

parallel with the on-going development and application of quantitative indicators for early warning. The system is an overlayed FORTRAN program consisting of 1 main overlay, PROGRAM ANALYZ; 4 primary overlays; PROGRAMS DIALOG, SEARCH, PLOT, AND TABULAR; 29 subroutines; and 2 functions. The system is written in Super FORTRAN (a United Computing Systems, Incorporated compiler), and requires a time-sharing driver resident on a host mainframe. The program is relatively inefficient without the use of a Tektronix graphics terminal.

Sample Program Output

A sample of the output from the crisis early warning prototype system is notated to describe the area in the program from which the output is generated.

System Design

A graph of the crisis early warning prototype shows the relationship each overlay and its associated subroutines has to the main overlay as well as to each other.

System Flow Charts

The main overlay and each of the primaries are flow-charted. Calls to the subroutines are shown as predefined processes and are not charted individually.

Documented FORTRAN Source Code

To allow access and ease of modification to any FORTRAN programmer, each routine is internally documented. Appendixes A and B contain, respectively, the NATION file and the RWINDX file, the index for the events file by month.

CONTENTS

		Page
SUMMARY		ii
FIGURES		vi
ACKNOWLEDGMENT		vii
1.0	INTRODUCTION	1
1.1	Overview of the Problem	1
1.2	Overview of a Proposed Solution	1
2.0	DESIGN, TEST, AND DEMONSTRATION OF THE FIRST-STAGE CRISIS EARLY WARNING PROTOTYPE SYSTEM	3
2.1	Range of Crisis Indicator	3
2.2	Forecasting Capability	7
2.3	Computer Base	8
2.4	Testing the Crisis Early Warning Prototype System	8
2.5	Demonstrating the Prototype System	9
3.0	SYSTEM DESCRIPTION	10
3.1	Objective of the Program	10
3.2	Overlay Descriptions	11
3.2.1	Main Overlay (0,0) PROGRAM ANALYZ	11
3.2.2	Primary Overlay (1,0) PROGRAM DIALOG	12
3.2.3	Primary Overlay (2,0) PROGRAM SEARCH	13
3.2.4	Primary Overlay (4,0) PROGRAM PLOT	14
3.2.5	Primary Overlay (5,0) PROGRAM TABULAR	16
3.3	Future Enhancement	16
4.0	SAMPLE PROGRAM OUTPUT	18
5.0	SYSTEM DIAGRAM	42

	Page
6.0 SYSTEM FLOW CHARTS	44
7.0 DOCUMENTED FORTRAN SOURCE CODE	60
APPENDIX A: NATION FILE	97
APPENDIX B: RWINDX FILE	101
REFERENCES	104
DISTRIBUTION LIST	106
DD FORM 1473	110

FIGURES

Figure		Page
2-1	Range of Crisis Indicators	3
2-2	The WEIS Events Coding Scheme	5
2-3	Political Event Indicators	6

ACKNOWLEDGMENT

This report describes the product of a dedicated and combined effort of many individuals without whose ideas and participation nothing would have been accomplished. Acknowledgments to those whose work is presented here include thanks to Richard W. Parker and Judith Ayres Daly who contributed to the substantive component of the project. Special thanks to L. Scott Randall and James J. Allen, Jr. whose contributions were the foundations upon which the software was built. Thanks is also due to Michael L. Hays and Theresa A. Buxton for their helpful editorial assistance as well as to John F. Correnti, John E. Billheimer, Judith Mial, and Phyllis Hutchison for their design and advice in the artwork and production of the document. Acknowledgment also to Lee Decker who took on the responsibility of the preparation and coordination of all the efforts. Finally, acknowledgment to Stephen J. Andriole and Robert A. Young for their leadership in this project.

SOFTWARE DESIGN FOR AN INTERACTIVE CRISIS EARLY WARNING PROTOTYPE SYSTEM

1.0 INTRODUCTION

This report describes the design and use of interactive computer software necessary for the development of an integrated crisis warning system. Interactive software is an essential part of the development of a user-oriented integrated system, as well as the development and test of prototype crisis warning systems. Accordingly, this report summarizes the development and test of the first-stage crisis early warning prototype system during the period January 1 to June 30 and, to a much greater extent, during the period July 1 to September 30, 1976.

1.1 Overview of the Problem

Recent history has acquainted us with a variety of unique problems, including resource and energy scarcity, population expansion and maldistribution, nuclear proliferation, and international terrorism. As a dominant global actor, the U.S. must deal with such problems, especially when they evolve into threatening international crises. Procedures designed to anticipate, avert, and manage problems and crises of all sorts should thus continue to be developed, evaluated, and improved.

In an effort to respond to the defense imperatives connected with intra- and international problems and crises, the defense community has devoted attention to the development of approaches and methods designed to enhance its overall warning and management capabilities. While much of this work has been productive, much remains to be done. Specific problems include the systematic identification, use, and combination of quantitative military, political, and economic indicators; the integration of quantitative estimates of U.S. military, political and economic interests abroad; and, perhaps most importantly, the development and integration of techniques for the rapid and efficient retrieval, manipulation, and display of important information.

1.2 Overview of a Proposed Solution

In an effort to contribute to the defense community's ability to forecast intra- and international crises, DDI

has developed and tested a first-stage crisis early warning prototype system.[1] The testing addressed two considerations. First, it involved the ability of the prototype to forecast retrospectively a number of international crises; and, second, it involved demonstrating precisely how crisis warning systems such as the prototype might be used. The interactive software discussed in this report is relevant to both aspects of the test. Section 2.0 briefly introduces the crisis early warning prototype system, while sections 3.0 through 7.0 of this report describe and display these software capabilities.

[1] For more detail on the first-stage prototype, see S. J. Andriole, Progress Report on the Development of a Crisis Early Warning Prototype System, Technical Report 76-19 (McLean, VA: Decisions and Designs, Incorporated, December, 1976).

2.0 DESIGN, TEST, AND DEMONSTRATION OF THE FIRST-STAGE CRISIS EARLY WARNING PROTOTYPE SYSTEM

2.1 Range of Crisis Indicators

The prototype is comprised of quantitative political indicators developed with reference to the concept of "event interaction." Future prototypes will be comprised of additional indicators, as Figure 2-1 suggests with reference to the range of potential indicators.

MILITARY				POLITICAL				ECONOMIC			
Static		Dynamic		Static		Dynamic		Static		Dynamic	
I	E	I	E	I	E	I	E	I	E	I	E

I – INTERNAL
E – EXTERNAL

Figure 2-1

RANGE OF CRISIS INDICATORS

Event-interactions may be defined as official statements or actions which cross national boundaries.[1] They are particularly suited to the task of short-range crisis forecasting because they are dynamic and readily receptive to measurement.

---

[1] The concept of an event-interaction was developed with reference to the study of international relations by Charles A. McClelland. See C. A. McClelland, Theory and the International System (New York: Macmillan, 1966), and C. A. McClelland et al., World Event/Interaction Survey (WEIS) Handbook and Codebook, WEIS Technical Report (Los Angeles: University of Southern California, January, 1969). Also see P. M. Burgess and R. W. Lawton, Indicators of International Behavior: An Assessment of Events Data Research, Sage International Studies Series, Vol. 1 (Beverly Hills, CA, 1972); C. W. Kegley et al., eds., International Events and the Comparative Analysis of Foreign Policy (Columbia, SC: University of South Carolina Press, 1975); and E. E. Azar and J. D. Ben-Dak, eds., Theory and Practice of Events Research (New York: Gordon and Breach, 1975); among many others.

Event-interactions are generated via public or private (classified) sources and an "event coding scheme." For example, if the New York Times reported that the Soviet Union warned the Chinese to withdraw from a particular border, the event would be coded as a "warning" and assigned a numeric code. Similarly, requests, proposals, accusations, and the like are coded numerically and stored for future retrieval and processing. The event coding scheme employed for prototypical purposes is the product of the World Event Interaction Survey (WEIS), an event data project which, with ARPA support, refined the methodology associated with the collection and treatment of event data. Cnrrently, the WEIS data file consists of over 78,000 events coded from the New York Times for the period 1966 to the present.[2]

Events may thus be classified according to a predetermined coding scheme. The WEIS scheme recognizes 63 kinds of events in 22 broad categories. There are 10 cooperative categories and 12 conflictual ones. All 22 appear in Figure 2-2.

The basic coding elements include the type of event observed as well as the time of its observation. Actors and targets constitute the remaining two elements. In a very real sense, then, the coding elements provide information regarding who (actor) does what (event) to whom (target) and when (time).

A number of crisis indicators have been developed from event data. Ours have been organized with reference to the concepts of volume and variety and the realities of one- and two-way flows.[3] As Figure 2-3 illustrates, we have entered activity, tension, and uncertainty indicators into the volume and variety cells.

---

[2]The WEIS data set continues to be updated and used by scholars, R&D analysts, and policy-makers, and constitutes one of the very few--if only--consistently coded ten-year-plus data set.

[3]The concepts of volume and variety were outlined by McClelland. See C. A. McClelland et al., The Management and Analysis of International Event Data: A Computerized System for Monitoring and Projecting Event Flows, WEIS Technical Report (Los Angeles: University of Southern California, September, 1971). Also see R. A. Young et al., The Utilization of ARPA-Supported Research for National Security Planning, Technical Report (Arlington, VA: CACI, Incorporated, October, 1972) and B. I. Spector et al., Quantitative Indicators for Defense Analysis, Final Technical Report (Arlington, VA: CACI, Incorporated, June, 1975). We have, however, made a number of adjustments to the political indicators developed in the above reports.

1. YIELD

011 Surrender, yield to order, submit to arrest, etc.
012 Yield position; retreat; evacuate
013 Admit wrongdoing; retract statement

2. COMMENT

021 Explicit decline to comment
022 Comment on situation-pessimistic
023 Comment on situation-neutral
024 Comment on situation-optimistic
025 Explain policy or future position

3. CONSULT

031 Meet with; at a neutral site; or send note
032 Visit; go to
033 Receive visit; host

4. APPROVE

041 Praise, hail, applaud, condolences
042 Endorse other policy or position; give verbal support

5. PROMISE

051 Promise own policy support
052 Promise material support
053 Promise other future support
054 Assure; reassure

6. GRANT

061 Express regret; apologize
062 Give state invitation
063 Grant asylum
064 Grant privilege, diplomatic recognition de facto relations, etc.
065 Suspend negative sanctions; truce
066 Release and/or return persons or property

7. REWARD

071 Extend economic aid
072 Extend military assistance
973 Give other assistance

8. AGREE

081 Make substantive agreement
082 Agree to future action or procedure; agree to meet, to negotiate

9. REQUEST

091 Ask for information
092 Ask for policy assistance
093 Ask for material assistance
094 Request action; call for
095 Entreat; plead; appeal to

10. PROPOSE

101 Offer proposal
102 Urge or suggest action or policy

11. REJECT

111 Turn down proposal; reject protest demand, threat, etc.
112 Refuse; oppose, refuse to allow

12. ACCUSE

121 Charge; criticize; blame; disapprove
122 Denounce; denigrate; abuse

13. PROTEST

131 Make complaint (not formal)
132 Formal complaint or protest

14. DENY

141 Deny an accusation
142 Deny an attributed policy, action, role or position

15. DEMAND

150 Issue order or command, insist; demand compliance, etc.

16. WARN

160 Give warning

17. THREATEN

171 Threat without specific negative sanctions
172 Threat with specific negative sanctions
173 Threat with force specified
174 Ultimatum; threat with time limit and negative sanctions specified

18. DEMONSTRATE

181 Nonmilitary demonstration; walk-out on
182 Armed force mobilization, exercise and/or display

19. REDUCE RELATIONSHIP

191 Cancel or postpone event
192 Reduce routine international activity; recall officials, etc.
194 Halt negotiations
195 Break diplomatic relations

20. EXPEL

201 Order personnel out of country
202 Expel organization or group

21. SEIZE

211 Seize position or possessions
212 Detain or arrest person(s)

22. FORCE

221 Non-injury destructive act
222 Nonmilitary injury/descruction
223 Military engagement

Figure 2-2

THE WEIS EVENTS CODING SCHEME

	Volume	Variety
One Way Flow (A ⟶ B) (B ⟶ A)	Total Activity Level Cooperative Activity Level Conflictual Activity Level	Tension Level Uncertainty Level
Two Way Flow (A ⟷ B)	Total Activity Level Cooperative Activity Level Conflictual Activity Level	Tension Level Uncertainty Level

Figure 2-3

POLITICAL EVENT INDICATORS

One- and two-way (volume) flows yield activity levels of three distinct types. First, it is possible to monitor the volume of total activity (the total number of cooperative and conflictual events) sent or exchanged. Second, it is possible to monitor the level of cooperative activity sent or exchanged. And third, it is possible to monitor the conflictual activity level.

The total, cooperative and conflictual, one- and two-way activity levels suggest the need for indicators which will calculate the mix of cooperative and conflictual events. This derivative indicator may be regarded as a tension level. Tension ranges from 0 to 100 and reflects the percentage of the total number of events sent or exchanged that are cooperative and conflictual, and may be calculated as follows:[4]

COOPERATIVE BEHAVIOR (CO)

PROPOSE
AGREE
PRAISE . . .

CONFLICTUAL BEHAVIOR (CON)

DENOUNCE
ACCUSE
THREATEN . . .

$$\text{TENSION} = \left\{ \frac{CO - CON}{CO + CON} + 1 \right\} \times 50$$

[4]This algorithm is derived from those, including policy style and relations, developed at CACI. See, for example, T. Rubin, D. Peasley, and L. Hopkins, Dissemination and Evaluation of Quantitative International Affairs

The third indicator measures the range of actions occurring in each of the event categories. Actors which initiate numerous events across the 22 event categories may be said to behave "inconsistently," while those who initiate only a small number of event types are more consistent. The measure of this consistency or inconsistency is, in information theory, regarded as Hrel (relative uncertainty),[5] which may range from 0 to +1.

These political activity, tension, and uncertainty indicators are tracked according to how they deviate from normal ranges of behavior. In order to measure the precise degree of abnormality exhibited by individual nations, or pairs or groups of nations, standard deviations from the normal (mean) political activity, tension, and uncertainty levels are calculated and standardized into Z-scores. Z-scores indicate the number of standard deviations from the mean and are calculated as follows:

$$\text{Z-SCORE} = \frac{\text{(SCORE AT TIME n)} - \text{(MEAN OF SCORES UP TO TIME n-1)}}{\text{STANDARD DEVIATION FOR ALL SCORES UP TO TIME n-1}}$$

2.2 Forecasting Capability

The prototype is endowed with a number of specific forecasting options which constitute its overall forecasting capability. The options include an international crisis

---

Indicators, Technical Report (Santa Barbara, CA: CACI, Incorporated, July, 1972) and Spector, Quantitative Indicators. Previous indicators ranged in value from -1 to +1. They also acknowledged the impact of neutral events. Our 0 to 100 one- and two-way tension levels have been modified with special reference to crisis forecasting. We are currently examining a number of aggregations and weighting techniques for specific application to crisis forecasting.

[5] Research regarding Hrel was pioneered in international crisis research by McClelland. See C. A. McClelland et al., Management and Analysis of International Event Data; C. A. McClelland, "The Beginning, Duration, and Abatement of International Crises: Comparisons in Two Conflict Arenas," in C. F. Hermann, ed., International Crises: Insights from Behavioral Research (New York: Free Press, 1972, pp. 18-108); and D. A. Zinnes, Contemporary Research in International Politics (New York: Free Press, 1976).

object, a short-range retrospective goal, and an associative method.

The prototype is thus designed to generate short-range forecasts of international political, military, and economic crises.

Insofar as the monitoring of events does not in itself shed much light upon the explanation of event flows, the methodology used is not objective in a causal sense. Instead, the method is an associative one which is less objective than those based, for example, upon causal modeling. Accordingly, the method involves monitoring Z-scores (for all of the indicators, all of the flows, and for specific time increments) and linking the scores with previous pre-crisis, crisis, and non-crisis behavior (again for all the indicators, all the flows, and for specific time increments). The method thus requires that pre-crisis, crisis, and non-crisis periods be identified, separated, and then associated with the Z-scores observed during past periods. This procedure enables us to assign probabilistic meanings to Z-scores when they are observed at certain levels in the future.

2.3 Computer Base

The prototype is currently endowed with efficient information storage, retrieval, processing, and display capabilities. With the aid of a Tektronix 4051 graphics terminal and the software described at length below, the user is able to command the rapid calculation and display of one- and two-way activity, tension, and uncertainty levels on a monthly, quarterly, or yearly basis. In addition, the system has the capability to present the indicators in different forms, including numerical tables comprised of frequencies, Z-scores, tension levels, and the like, with graphs or bar-charts of the same plotted over extended or relatively short periods of time, or both.

2.4 Testing the Crisis Early Warning Prototye System

The system has been tested for its ability or inability to forecast crises retrospectively. The specific crises forecasted include the 1967 Sino-Soviet border clash, the 1968 Soviet invasion of Czechoslovakia, and the 1971 Indo-Pakistani War. Detailed results of these tests may be found elsewhere.[6] We need to note here only that the analysis of the three crises revealed that the use of quantitative political indicators of activity, tension, and uncertainty

---

[6]See Andriole, Progress Report.

would have improved the ability of the defense community to forecast the crises.

2.5 Demonstrating the Prototype System

Another operational dimension of the prototype system is comprised of the computer hardware and software necessary for the design and installation of interactive user-oriented systems. Accordingly, the major components of the demonstration system now include an event-interaction data base, interactive software, and graphic and tabular display capabilities. As described below, the interactive software enables one to issue commands and receive prompt responses. While the first cut of the demonstration system requires the user to respond to a good deal of prompting, later versions will require minimal prompting and permit quick access to all aspects of the system.

3.0 SYSTEM DESCRIPTION

"-CRISIS" is the name of the command file that causes execution of the crisis early warning prototype system. The system itself, although called a system, is actually an overlayed FORTRAN program. The program currently resides on disk in two forms: FORTRAN source code and relocatable binary. There are also several archived copies maintained on 800 BPI, odd parity, magnetic tape. The portability of the program itself depends upon several factors: the similarity of ANSI FORTRAN IV to United Computing Systems, Incorporated (UCS) Super FORTRAN, the availability of Tektronix Plot-10 Software, and the access to certain library and intrinsic functions. The fulfillment of these requirements makes the crisis early warning prototype system a highly efficient, convertable, stand-alone program that provides all functions pertaining to the selection and display of quantitative political indicators for early warning. Consequently, the program is, in fact, a system.

3.1 Objective of the Program

Our goal has been very specific from the outset. It is to parallel the research connected with the development and application of quantitative (military, political, and economic) indicators for early warning.[1] As new indicators are developed and integrated into the research design, the program will be modified to accommodate their display. Accordingly, the prototype system described in this report may be viewed as the foundation upon which future systems will be built. The system may or may not remain intact as it currently exists; at the same time, it will remain either as the primary vehicle through which the computer sciences department will perform the required tasks, or as a smaller part of a more sophisticated whole.

The first-stage prototype system was configured by using the modular programming technique to yield one main overlay and four primary overlays. Each overlay contains

---

[1] For a detailed look at the scope, method, and purpose of the crisis forecasting research program see Decisions and Designs, Incorporated, The Development of a Crisis Early Warning Prototype System (Technical Proposal, McLean, VA, December, 1975); Decisions and Designs, Incorporated, The Development of a Crisis Early Warning Prototype System (Technical Proposal, McLean, VA, July, 1976); and S. J. Andriole, Progress Report.

all the subroutines required to ensure the autonomy of the overlay resident at any given moment. It is also important to note that the main overlay always remains in core; that all overlays are called from the main overlay and not from each other; that overlays may be called in any order regardless of their number; that once an overlay has been called, it resides in core below main; that variables are passed between overlays via labeled COMMON; that subroutines may be called only by the overlay program in which they reside; and that the primary purpose of an overlayed program is to conserve core space.

The CEWPS was written in Super FORTRAN (SFO), a FORTRAN VI extension language offered by United Computing Systems, Incorporated. The CEWPS can be executed by using the SFO compiler on any Control Data 6000 series computer system running the KRONOS operating system. KRONOS itself, however, is not a requirement. Any timesharing driver should suffice since the system utilizes the primitive FORTRAN I/O READ and WRITE functions. It must be emphasized, however, that the availability of TEKTRONIX Plot-10 (or similar graphics software) and compatibility of FORTRAN Library and intrinsic functions are requisite to the operation of the program.

Since the prototype system generates sophisticated graphical output, it would be inefficient to execute without a TEKTRONIX 4010, 4012, 4014, or 4051 terminal. It is also important to provide a user with 1200 baud communications, although a degredated version can be made to plot at 300. The primary overlay which produces the graphics is hard-coded to execute at 1200 baud.

The program consists of 1 main overlay, 4 primary overlays, 26 subroutines, and 2 functions (see Section 4.0 diagrams). Section 3.2 briefly describes each overlay and subroutine. These comments are further supplemented by Sections 4.0, 5.0, and 6.0. Section 6.0 is incomplete in that it only charts the flow of main and primary overlays (not the subroutines). Section 7.0 includes all source code. Each routine is internally documented, a step deliberately undertaken to allow any FORTRAN programmer access and ease of modification.

3.2 Overlay Descriptions

3.2.1 Main overlay (0,0) PROGRAM ANALYZ. This routine is the main driver. It consists of the program ANALYZ and three subroutines, BLANC, CREDITS, and INSTRNS. (See Section 4.0, System Diagram.) Program ANALYZ prints the system title and calls CREDITS. It then calls overlay 1 and 2 to initiate the selection of crisis cases. Once this is complete, overlay 2 returns to main and displays the user commands. Through a series of "if tests" on the user response,

the proper overlay and/or subroutine is called. The possible responses to user commands and the related calls are as follows:

'I' -- calls subroutine INSTRNS
'G' -- calls Overlay 4,0 (graphics)
'T' -- calls Overlay 5,0 (tabular O/P)
'F' -- calls Overlays 1,0 & 2,0 (a variation of initial case selection
'A' -- calls overlays 1,0 & 2,0 (a variation of initial case selection)
'C' -- calls overlays 1,0 & 2,0 (initial case selection
'E' -- Go to Print Statement & END program

Subroutine Credits--This subroutine contains no logic. It simply writes formatted output to the CRT. The routine immediately returns to Main.

Subroutine Blanc--This subroutine consists primarily of two calls to Plot-10 (AG II) (i.e., INITT and FINITT). By utilizing these calls, the subroutine can accomplish two tasks. First, it blanks the CRT. Second, it moves the cursor to the home position. In the near future, we plan to replace these calls with the simple output of a control character to blank and home the screen; however, at 1200 baud, it becomes difficult to accomplish this task without running into a timing problem (note the existing print statement).

Subroutine INSTRNS--Like CREDITS, this subroutine contains no logic. It simply writes to output a formatted description of the operation of Overlay 4,0 (graphics). This routine is called when the user responds with the character 'I' to the USER COMMAND request.

3.2.2 Primary overlay (1,0) PROGRAM DIALOG. This routine is simply a primary overlay. It consists of the program DIALOG and ten subroutines: NCODE, CASE, SETUP, TSETUP, SSBC, CZEINV, INDWAR, USSOV, USPEO, and SOVPEO (see Section 5.0, System Diagram). Program DIALOG does none of the actual user interaction. However, it does contain almost all the logic to call the proper subroutine under the current parameters, and it initiates the reading of both the NATION file and the RWINDEX file.

DIALOG will call Subroutine CASE and return the selected case number. It will then call any one of the subroutines associated with that case number to provide a heading to further parameter selection. The subroutines are associated with the menu provided in subroutine CASE in the following manner:

1 = SSBC (Sino-Soviet Border Clashes)
2 = CZEINV (Czechoslovakian Invasion)
3 = INDWAR (India-Pakistani War)
4 = USSOV (U.S.-Soviet Union)
5 = USPEO (U.S.-Peoples Republic of China)
6 = SOVPEO (Soviet Union-Peoples Republic of China)

Once the headers have been printed, the overlay calls subroutine SETUP.

Subroutine SETUP asks the user for the following information, which it passes back to the Overlay:

Initiator - 3 characters
Recipient - 3 characters
Flow Direction - 1 integer
Time Increment - 1 integer

When these parameters are converted to the proper values in common, DIALOG calls subroutine TSETUP. This subroutine asks the user for the start date (SDATE) and finish date (FDATE) of the time period for which the user desires to collect event data. The SDATE and FDATE parameters are asked for in the form MMM DD and returned to dialog in the form YYMMDD.

Having collected all necessary information to begin processing, the program then calls subroutines NCODE in order to validate the 3-character country code, look up the associated integer nation number, and store the 3-word (30 character) expanded country name, for example,

USA 002 UNITED STATES.

Finally, DIALOG will open and read file RWINDEX and find the address of the beginning date (SDATE) on the main data base.

3.2.3 Primary Overlay (2,0) PROGRAM SEARCH. This routine is another primary overlay. It consists of the program SEARCH, eight subroutines, and one function (see Section 5.0, System Diagram).

The SEARCH program collects the data from the main data base into arrays according to the parameters derived from Overlay 1,0. It also maintains the accumulation of several counters. The function NXTINC is a very integral part of the data collection. This routine signals the change of time increment from day to day, month to month, period to period, or year to year.

Once all calculations have been made, control is usually returned to MAIN.

Subroutine CONVRT--This subroutine is passed an integer array from SEARCH. It returns the same array in the real mode.

Subroutine ZSCORP--This subroutine calculates the progressive means and progressive Z-scores for a collection of items (array passed to it). The arrays passed should have data collected in increments starting at the beginning of the data base. It is also the function of this subroutine to use this data to calculate the Z-score and to delete all Z-scores except those that fall between SDATE and FDATE once the calculations are made.

Subroutine PTABLE--This subroutine assigns probabilities from Z-scores according to the type of event data passed to it. There are three types of event data Z-scores: TOTAL, CONFLICTUAL, and COOPERATIVE. A parameter passed from SEARCH tells the routine which type of data it has and thereby begins a table lookup by using the proper boundaries.

Subroutine HREL--This subroutine calculates the Hrel of total activity event data for each of a collection of time intervals.

Subroutine TENSION--This subroutine calculates the Tension Factor (ratio of cooperative to conflictual events) for each of a collection of time intervals.

Subroutine SMLTAB--This subroutine prints the 30-day probabilistic tables from the probabilities calculated by ZSCORP for the total events array. This routine calls DEKODE and DEKODA subroutines. DEKODE splits the time interval date (SDATE and FDATE) into two parts (month and year). Subroutine DEKODA does essentially the same although it is sent an array in I6 format (YYMMDD) and returns the array in the format (MMM YY). Although these routines will execute for all possible time increment selections, they were primarily designed for monthly increments.

3.2.4 Primary Overlay (4,0) PROGRAM PLOT. This routine is a primary overlay. It consists of the program PLOT, seven subroutines, and a function (see Section 5.0, System Diagram).

The SEARCH program provides the graphic display capability to the CEWPS. This program calls subroutines which in turn call the PLOT-10 package from both the TCS and AG-II functions.

After rearranging all the arrays built in Overlay 2,0 in a more readable fashion, the program executes the PLOT-10 INITT command. This command blanks the CRT and

initiates the graphics mode. Immediately, subroutine CHOOSE is called to enter graphics parameters. CHOOSE solicits from the user the number of plots desired, type of plot, and the arrays to be plotted. A matrix is drawn by CHOOSE to provide an association between the arrays constructed in Overlay 2,0 and a unique name or number.

Subroutine PTABLE is called to recalculate probabilities, a necessary step since the plots generated by PLOTIT allow for only one character after the decimal point, whereas TABULAR output specifications require two characters.

Subroutine PLOTIT is then called in a loop from one to six times depending again upon the number of user selections made in subroutine CHOOSE. When the desired number of graphs have been drawn, subroutine TITLE is called to print a title.

If the type of graph drawn is a multi-graph (more than one plot on a single axis), subroutine LGND is called. This subroutine will draw the appropriate line and notate the associated array to which it belongs. There are two other subroutines that are used while in graphics mode and are called by the other subroutines, not the overlay. Subroutine USERS is a routine taken from the PLOT-10 AG II manual (p. 191) and modified slightly. This routine gives the capability of doing a point plot without an axis; in the CEWPS, we use it to display the probability values directly above the bar on a bar-chart or above the point on a graph with an arrow drawn to it.

Subroutine CHCNVT, like USERS, is called by subroutines other than PLOT. This routine takes a text character string of N length and converts it to USASCII code functions for Plot-10(AG II) plotter output. This routine is necessary for all special labeling and titles because Plot-10 will not accept hollerith constants.

When all graphs, labels, titles, and legends are completed, program PLOT exits the graph mode with a call to the PLOT-10 FINITT routine. This routine not only exits in graph mode but ensures that all buffers are dumped. After each graph(s) is drawn, the program reads from input to allow for a pause, and to give the user an opportunity to study the display or make a hard copy of it. Normal continuation involves simply hitting the RETURN key; however, one might also continue through the NUMB function.

NUMB reads the value which is typed in and conducts a table look-up check for a match between that input and any of the array numbers or unique names. If

there is indeed a match, the entire sequence for generating a plot (discussed above) is invoked without going through the CHOOSE subroutine. In effect, it executes a blow-up of the name or numbered array choosen. All possible arrays can be blown-up at this time. Valid responses include:

1 or TOT	6 or CONZ
2 or TOTZ	7 or TEN
3 or COO	8 or TENZ
4 or COOZ	9 or UNC
5 or CON	10 or UNCZ

Finally, program PLOT asks the user if he wishes another plot. A "yes" response will start him at the beginning with the call to CHOOSE, and the matrix of political indicators will appear. A "no" will exit from the overlay and return him to MAIN and to the user commands.

3.2.5 Primary Overlay (5,0) PROGRAM TABULAR. This routine is a primary overlay. It consists of the program TABULAR and one subroutine DEKODE. DEKODE is exactly like subroutine DEKODE in overlay 2,0. It is used to convert SDATE and FDATE to the more readable alpha characters used in the headings.

Program TABULAR produces printed output from the arrays generated in Overlay 2. In order to fit the output onto one CRT page, all tables are in a loop. If the number of rows is greater than 18, the tables are printed in two pages (1 to 18 and 19 to the end).

The first table printed includes output of the number of events, their Z-scores, and probabilities for each of the TOTAL, COOPERATIVE, and CONFLICTUAL activities as well as the increment date. The second table shows the date, TENSION, HREL, and their Z-scores.

The user is now asked if he desires event frequencies. If he answers "yes," TABULAR prints two more tables, the first showing the raw collection of cooperative events, by date, in their ten categories. The second table displays the conflictual events, again by date, in their twelve categories. The overlay then returns to "ANALYZ" (main) and again displays the user commands.

3.3 Future Enhancement

The prototype system will be modified to ensure the greatest possible code optimization. Code inefficiencies were caused solely by stringent deadlines and were not intended or desired. The current utilization of random FORTRAN I/O will be either deleted or improved. The general format of all graphics will be improved. A decision will

also be made as to whether or not to discontinue the use of Plot-10 (AG II) and revert to Plot-10 (TCS) in order to create a more tailored output than can be generated by the AG II package. New statistical indicators will also be incorporated to analyze expanded data bases. In short, the system will soon evolve into a more efficient, more attractive, expanded and highly polished product.

4.0 SAMPLE PROGRAM OUTPUT

********** CRISIS EARLY WARNING PROTOTYPE SYSTEM **********

OVERLAY 0,0

THE CRISIS EARLY WARNING PROTOTYPE SYSTEM WAS DEVELOPED FOR THE ADVANCED RESEARCH PROJECTS AGENCY'S CYBERNETICS TECHNOLOGY OFFICE BY DECISIONS AND DESIGNS, *INCORPORATED*.

THE SYSTEM IS DESIGNED TO SIMULATE HOW QUANTITATIVE POLITICAL INDICATORS MIGHT HAVE CONTRIBUTED TO THE U.S. DEFENSE COMMUNITY'S ABILITY TO FORECAST A NUMBER OF IMPORTANT INTERNATIONAL CRISES. IT IS ALSO DESIGNED TO ILLUSTRATE HOW THE GENERAL NATURE AND DIRECTION OF INTERNATIONAL AFFAIRS MAY BE MONITORED USING POLITICAL INDICATORS.

Subroutine "CREDITS"

OVERLAY 1,0

```
THE FOLLOWING INTERNATIONAL CRISIS CASES ARE NOW AVAILABLE
ON THE DEMONSTRATION SYSTEM :

  1.  SINO - SOVIET BORDER CLASHES,
      JANUARY , 1967 AND MARCH , 1969

  2.  CZECHOSLOVAKIAN INVASION,
      AUGUST , 1968

  3.  INDO - PAKISTANI WAR,
      NOVEMBER , 1971

THE SYSTEM ALSO CONTAINS THE FOLLOWING OTHER CASES :

  4.  U.S. - SOVIET UNION,
      JANUARY , 1966 - DECEMBER , 1975

  5.  U.S. - PEOPLES REPUBLIC OF CHINA,
      JANUARY , 1966 - DECEMBER , 1975

  6.  SOVIET UNION - PEOPLES REPUBLIC OF CHINA,
      JANUARY , 1966 - DECEMBER , 1975


PLEASE SELECT ONE CASE NUMBER :2
```

Subroutine "CASE"

OVERLAY 1,0

```
*** CZECHOSLOVAKIAN INVASION , AUGUST , 1968 ***

    PRIMARY ACTORS :
        SOVIET UNION (USR)
        CZECHOSLOVAKIA (CZE)
    OTHER ACTORS :
        UNITED STATES (USA)
        PEOPLES REPUBLIC OF CHINA (CHN)

PLEASE SELECT TWO ACTORS :(E.G. XXX,YYY )USR,CZE

SPECIFY ACTIVITY FLOW :
    0. ONE WAY (USR >>> CZE)
    1. ONE WAY (USR <<< CZE)
    2. TWO WAY (USR <-> CZE)2

SELECT TIME INCREMENT :
    1. MONTHLY
    2. QUARTERLY
    3. YEARLY      1

SET TIME PARAMETERS (JAN66,DEC75) :JAN68-DEC68

DO YOU DESIRE 30 DAY PROBABLISTIC FORCASTS (Y OR N) : Y
```

Subroutine "CZEINV"

Subroutine "SETUP"

Subroutine "TSETUP"

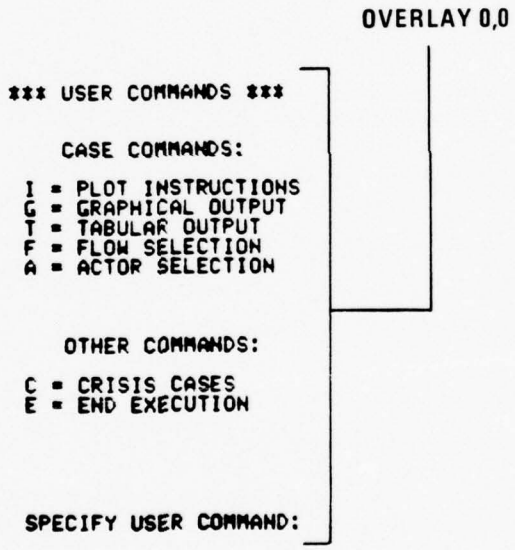
OVERLAY 2,0

MONTHLY PROBABILITY
JAN, 1968 - DEC, 1968

*** USR <<<>>> CZE ***

DATE	PROBABILITY
JAN 68	.10
FEB 68	.50
MAR 68	.70
APR 68	.70
MAY 68	.80
JUN 68	.40
JUL 68	.70
AUG 68	.60
SEP 68	.40
OCT 68	.05
NOV 68	.01
DEC 68	.01

Subroutine "SMLTAB"


OVERLAY 0,0
*** USER COMMANDS ***
CASE COMMANDS:
I = PLOT INSTRUCTIONS
G = GRAPHICAL OUTPUT
T = TABULAR OUTPUT
F = FLOW SELECTION
A = ACTOR SELECTION
OTHER COMMANDS:
C = CRISIS CASES
E = END EXECUTION
SPECIFY USER COMMAND:

OVERLAY 0,0

Subroutine "INSTRNS"

```
                    *** PLOTTER INSTRUCTIONS ***

UPON INITIAL ENTRY INTO THE GRAPHICS SECTION, THE POLITICAL
INDICATOR MATRIX WILL BE DISPLAYED.  THIS MATRIX DESCRIBES
EACH PLOT AVAILABLE TO THE USER AND IS NUMBERED ONE THROUGH
TEN (TEN POSSIBLE PLOTS).

IT IS THE USERS DECISION WHICH PLOTS ARE TO BE DISPLAYED,
HOW MANY, WHAT PLOT TYPE (BAR OR GRAPH), AND THE DESIRED
POSITION ON THE CRT.  THE FOLLOWING ARE VALID RESPONSES TO
QUERIES WHICH IMMEDIATELY FOLLOW THE INDICATOR MATRIX:

     1)  DO YOU WANT MORE THAN ONE PLOT PER GRAPH (Y OR N)

--THE 'Y' (YES) ANSWER WILL PROVIDE THE USER WITH CHOICES
  AVAILABLE FOR MORE THAN ONE GRAPH PLOTTED ON A SINGLE
  PAIR OF AXIS.

--THE 'N' (NO) ANSWER WILL LEAD THE USER THROUGH A GROUP OF
  QUESTIONS THAT WILL ALLOW HIM TO GENERATE FROM ONE TO SIX
  SIMULTANEOUS PLOTS, EACH HAVING ITS OWN AXIS.
```

OVERLAY 0,0

AFTER ANY GRAPH IS DISPLAYED THE SYSTEM AWAITS THE INPUT OF A 'RETURN' TO CONTINUE.

- OR -

THE USER MAY SPECIFY ANY OF THE FOLLOWING PLOT NAMES OR NUMBERS TO CREATE A BLOW-UP OF AN INDIVIDUAL GRAPH.

1	OR	TOT	6	OR	CONZ
2	OR	TOTZ	7	OR	TEN
3	OR	COO	8	OR	TENZ
4	OR	COOZ	9	OR	UNC
5	OR	CON	10	OR	UNCZ

2) ANOTHER DISPLAY (Y OR N)

--THE 'Y' (YES) ANSWER WILL CAUSE THE INDICATOR MATRIX TO BE RE-DRAWN.

--THE 'N' (NO) ANSWER WILL TERMINATE THIS OVERLAY AND RE-INITIATE THE 'SPECIFY USER COMMAND' SECTION.

Subroutine "INSTRNS"

OVERLAY 4,0

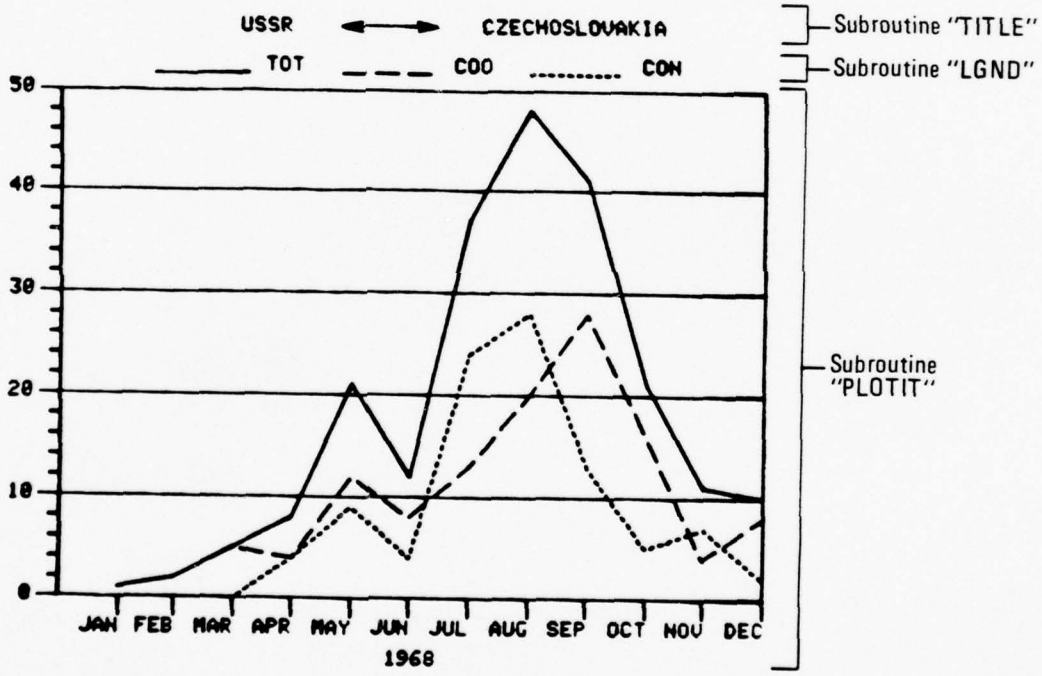
POLITICAL INDICATORS

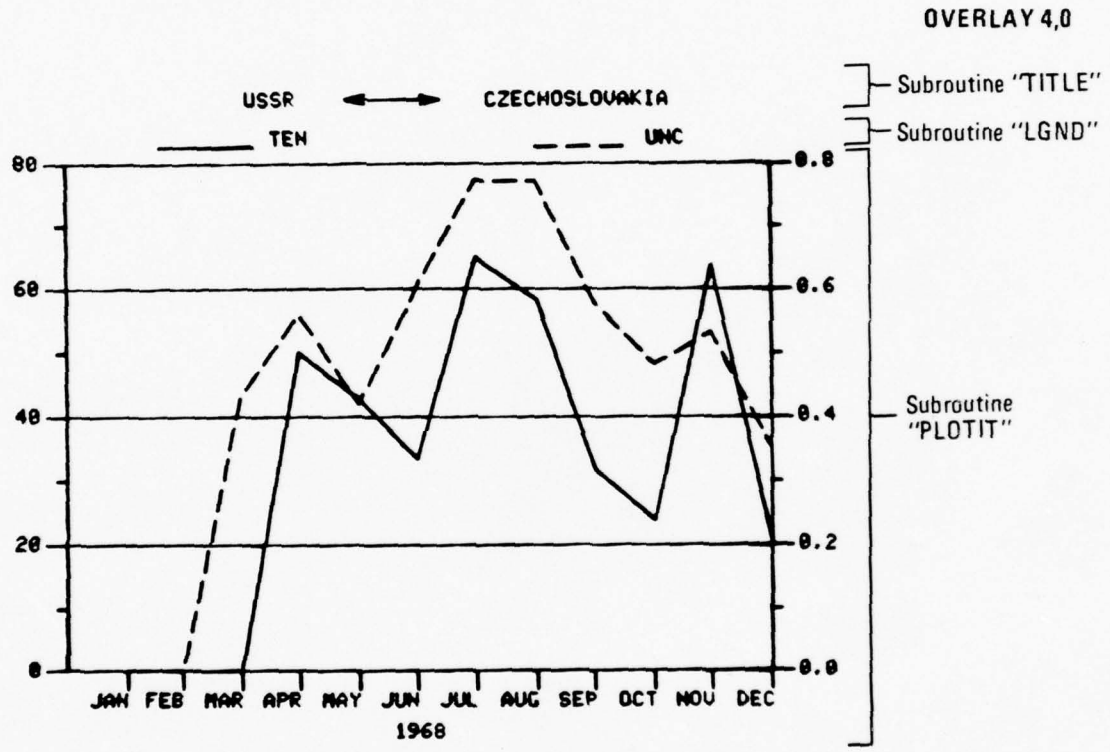
	TOTAL ACTIVITY	COOP ACTIVITY	CONFLICT ACTIVITY	TENSION LEVEL	UNCERT LEVEL
RAW DATA	1	3	5	7	9
Z-SCORE	2	4	6	8	10

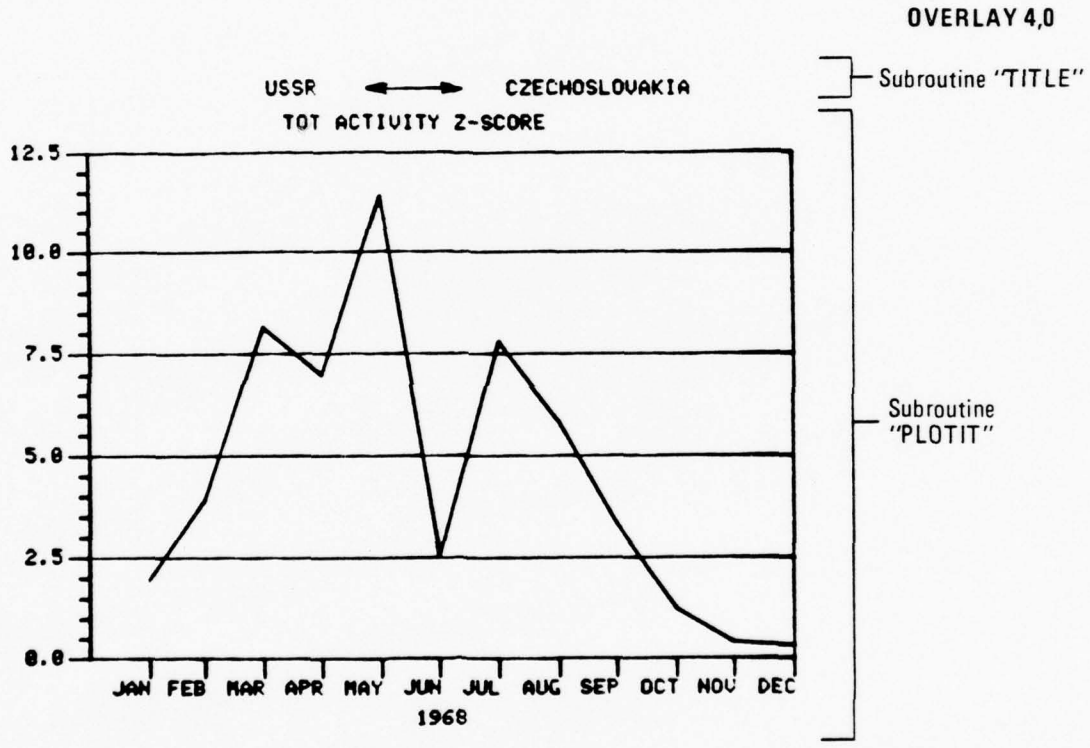
DO YOU WANT MORE THAN ONE PLOT PER GRAPH (Y OR N) Y
YOU HAVE THESE CHOICES:
1 = GRAPHS 1,3 + 5
2 = GRAPHS 2,4 + 6
3 = GRAPHS 7 + 9
4 = GRAPHS 8 + 10
PLEASE PICK ONE 1

Subroutine "CHOOSE"

OVERLAY 4,0


USSR ⟷ CZECHOSLOVAKIA
TOT
COO
CON
50
40
30
20
10
0
JAN FEB MAR APR MAY JUN JUL AUG SEP OCT NOV DEC
1968
Subroutine "TITLE"
Subroutine "LGND"
Subroutine "PLOTIT"


OVERLAY 4,0
Subroutine "TITLE"
Subroutine "LGND"
USSR ⟷ CZECHOSLOVAKIA
TEN
UNC
80
60
40
20
0
0.8
0.6
0.4
0.2
0.0
Subroutine "PLOTIT"
JAN FEB MAR APR MAY JUN JUL AUG SEP OCT NOV DEC
1968


OVERLAY 4,0
Subroutine "TITLE"
USSR ⟷ CZECHOSLOVAKIA
TOT ACTIVITY Z-SCORE
12.5
10.0
7.5
5.0
2.5
0.0
JAN FEB MAR APR MAY JUN JUL AUG SEP OCT NOV DEC
1968
Subroutine "PLOTIT"

OVERLAY 4,0

POLITICAL INDICATORS

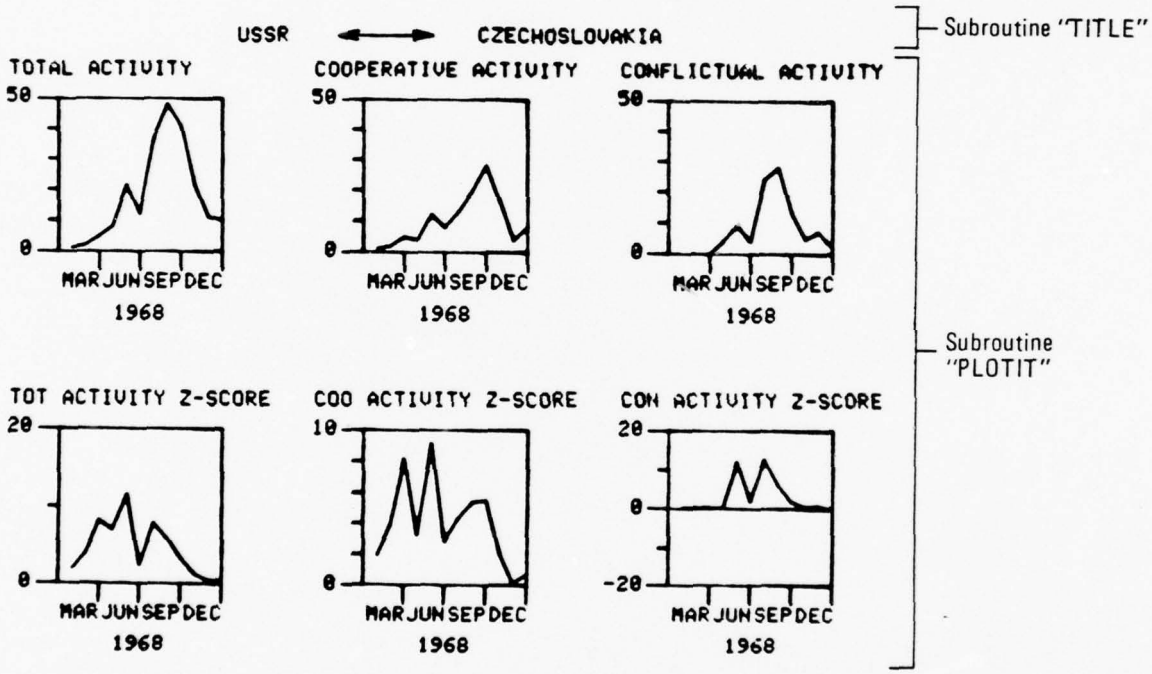
	TOTAL ACTIVITY	COOP ACTIVITY	CONFLICT ACTIVITY	TENSION LEVEL	UNCERT LEVEL
RAW DATA	1	3	5	7	9
Z-SCORE	2	4	6	8	10

DO YOU WANT MORE THAN ONE PLOT PER GRAPH (Y OR N) N
DO YOU WANT GRAPHS OR BAR-CHARTS (G OR B) G
HOW MANY GRAPHS DO YOU WANT (1-6) 6

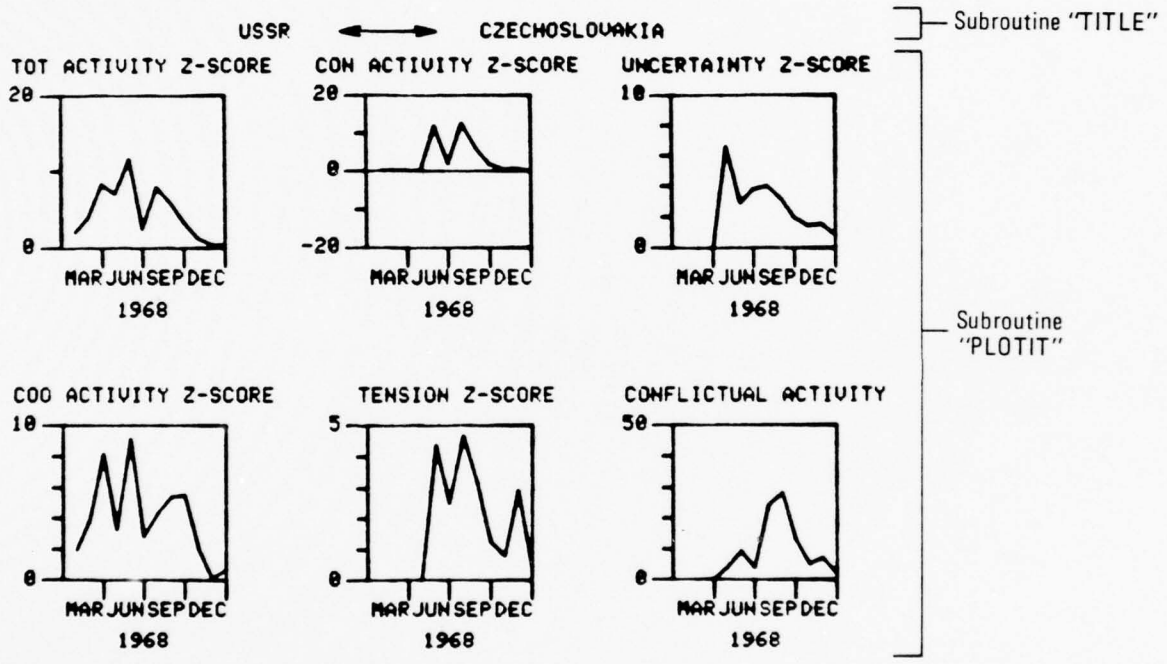
PLEASE TYPE IN THE 6 GRAPH NUMBERS YOU
WOULD LIKE DISPLAYED IN THE FORM (1,2, ... 6) 1 2 3 4 5 6

Subroutine "CHOOSE"

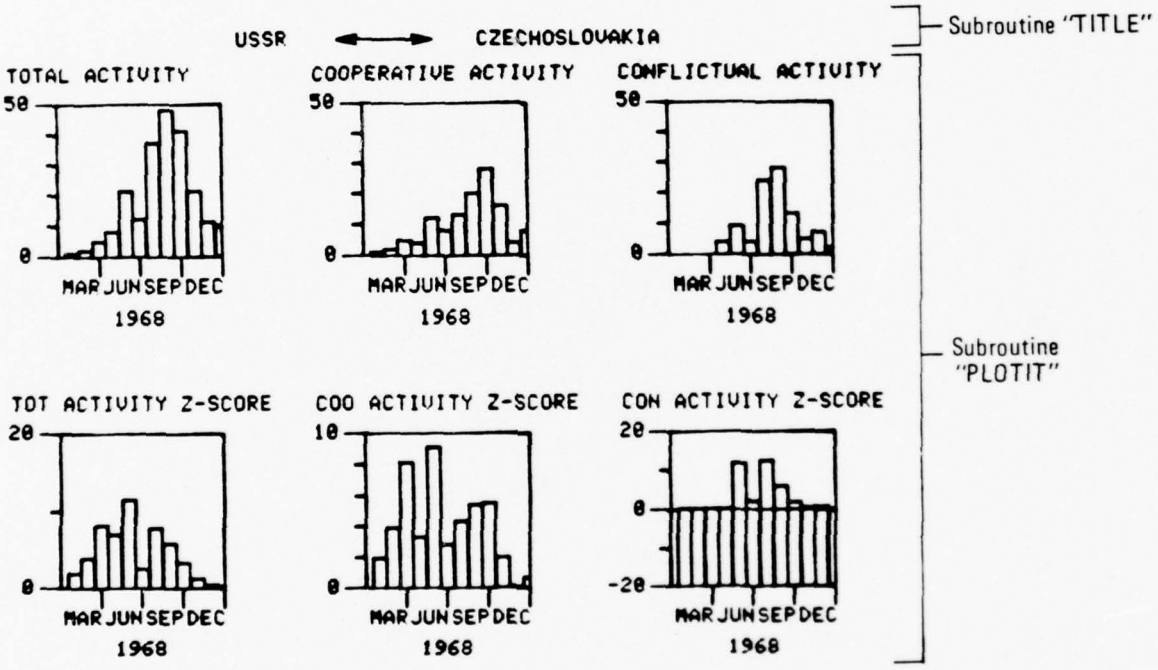
OVERLAY 4,0


USSR ⟷ CZECHOSLOVAKIA
Subroutine "TITLE"
TOTAL ACTIVITY
COOPERATIVE ACTIVITY
CONFLICTUAL ACTIVITY
50
0
MAR JUN SEP DEC
1968
TOT ACTIVITY Z-SCORE
COO ACTIVITY Z-SCORE
CON ACTIVITY Z-SCORE
20
10
-20
Subroutine "PLOTIT"

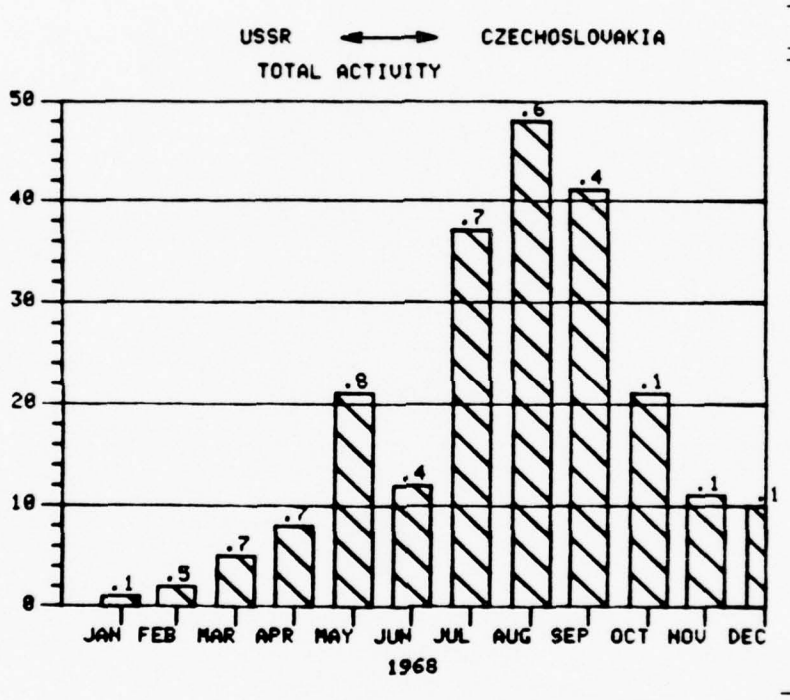
OVERLAY 4,0


USSR ⟷ CZECHOSLOVAKIA
Subroutine "TITLE"
TOT ACTIVITY Z-SCORE
20
0
MAR JUN SEP DEC
1968
CON ACTIVITY Z-SCORE
20
0
-20
MAR JUN SEP DEC
1968
UNCERTAINTY Z-SCORE
10
0
MAR JUN SEP DEC
1968
Subroutine "PLOTIT"
COO ACTIVITY Z-SCORE
10
0
MAR JUN SEP DEC
1968
TENSION Z-SCORE
5
0
MAR JUN SEP DEC
1968
CONFLICTUAL ACTIVITY
50
0
MAR JUN SEP DEC
1968

OVERLAY 4,0


USSR ⟷ CZECHOSLOVAKIA
Subroutine "TITLE"
TOTAL ACTIVITY
COOPERATIVE ACTIVITY
CONFLICTUAL ACTIVITY
50
0
MAR JUN SEP DEC
1968
TOT ACTIVITY Z-SCORE
COO ACTIVITY Z-SCORE
CON ACTIVITY Z-SCORE
20
10
0
-20
Subroutine "PLOTIT"

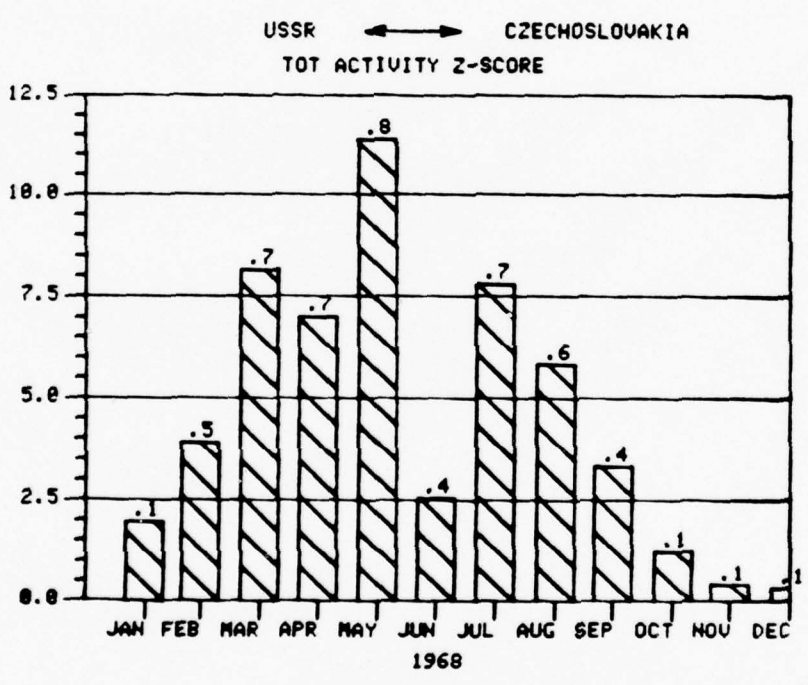
OVERLAY 4,0



Subroutine "TITLE"

Subroutines:
"PLOTIT"
"USERS"
"PTABLE"

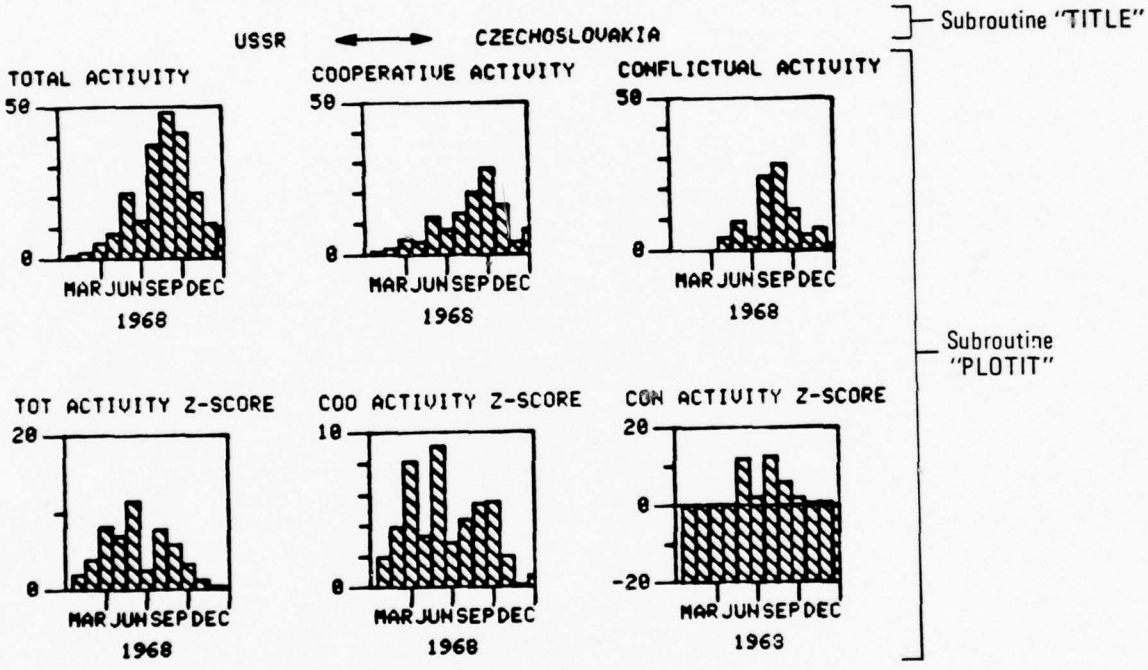
OVERLAY 4,0



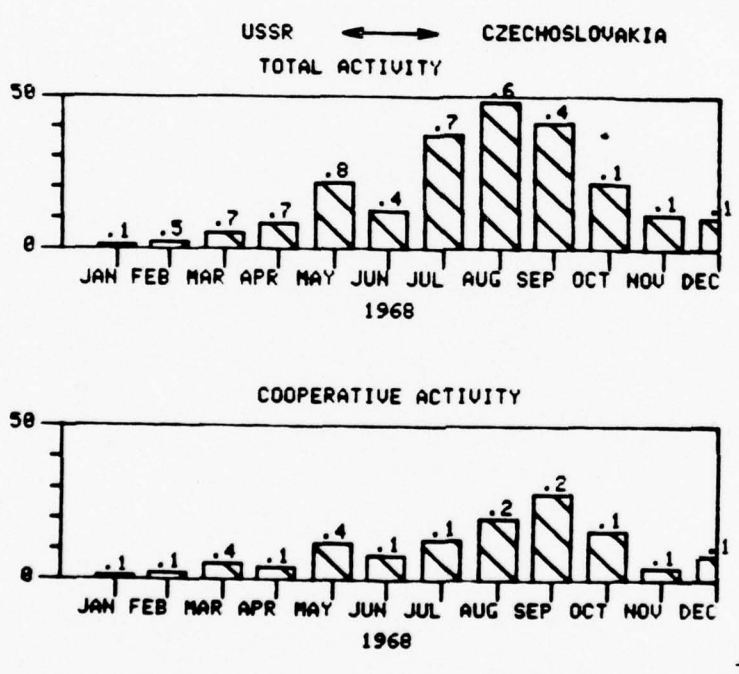
Subroutine "TITLE"

Subroutines:
"PLOTIT"
"USERS"
"PTABLE"

OVERLAY 4,0

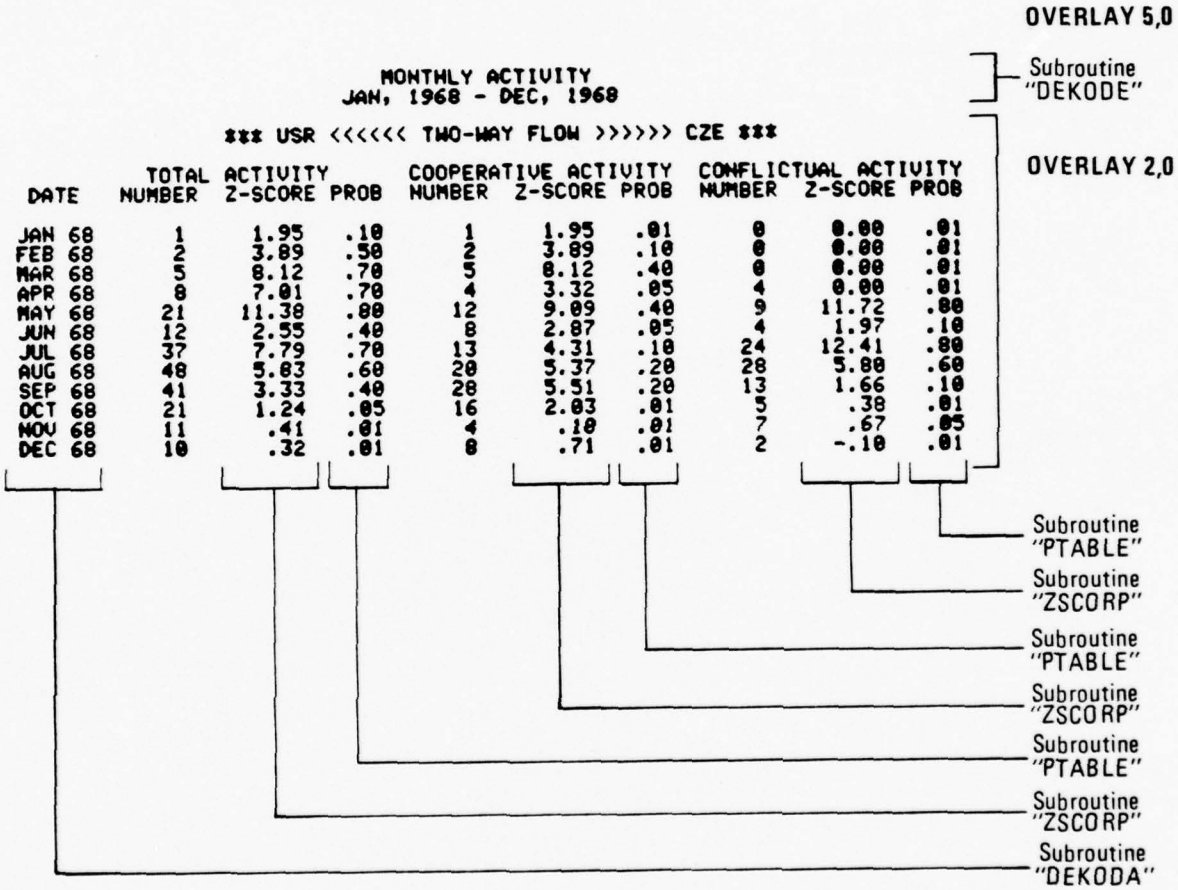

Subroutine "TITLE"
USSR ⟷ CZECHOSLOVAKIA
TOTAL ACTIVITY
COOPERATIVE ACTIVITY
CONFLICTUAL ACTIVITY
50
0
MAR JUN SEP DEC
1968
Subroutine "PLOTIT"
TOT ACTIVITY Z-SCORE
COO ACTIVITY Z-SCORE
CON ACTIVITY Z-SCORE
20
10
-20
1963

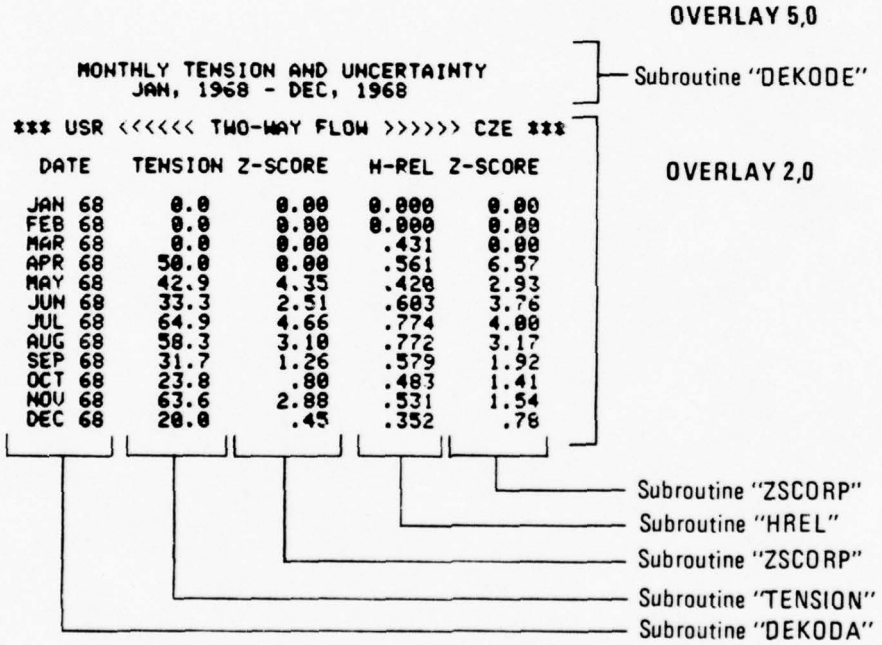
OVERLAY 4,0

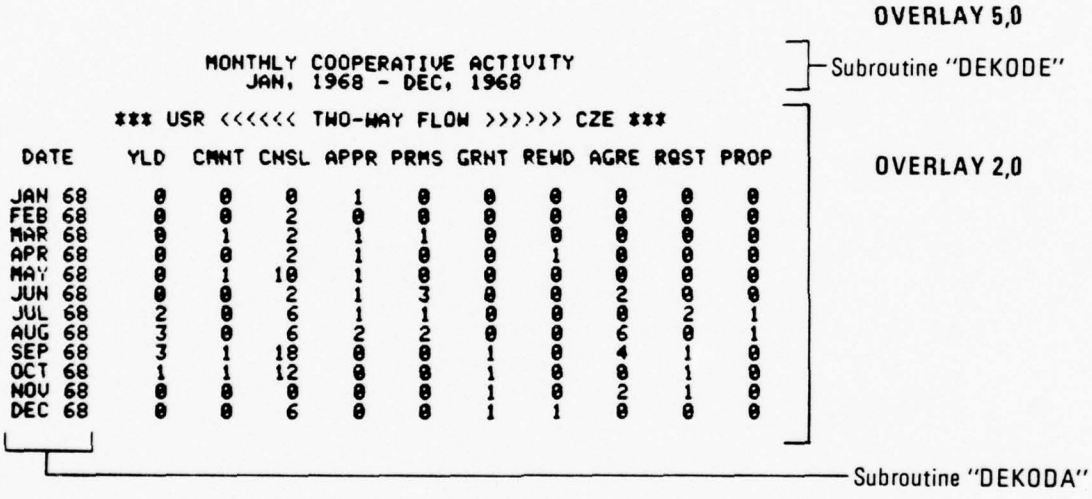


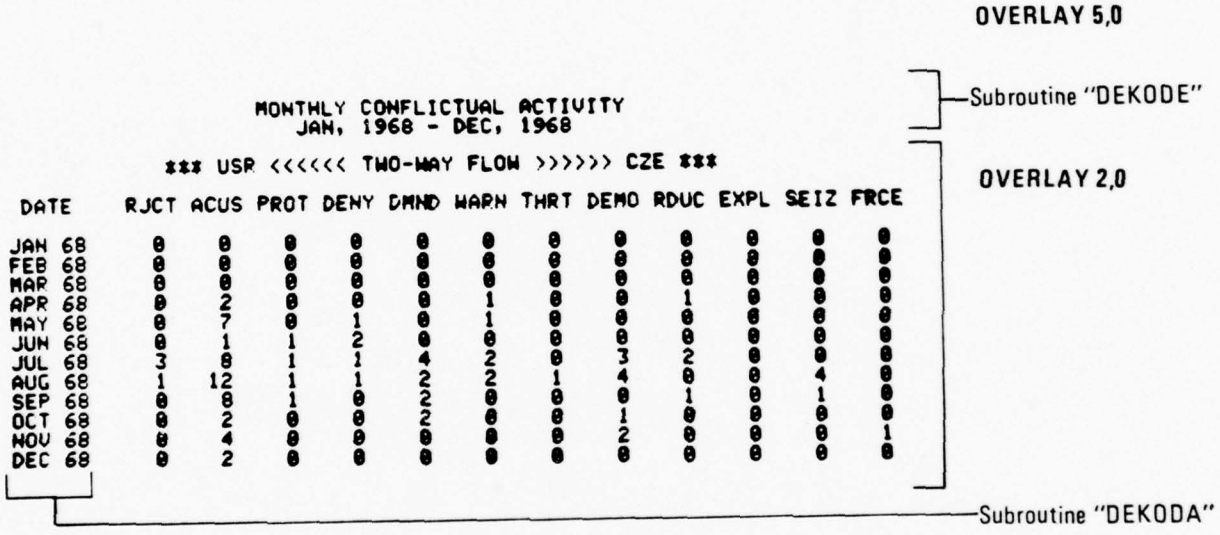
Subroutine "TITLE"

Subroutines:
"PLOTIT"
"USERS"
"PTABLE"

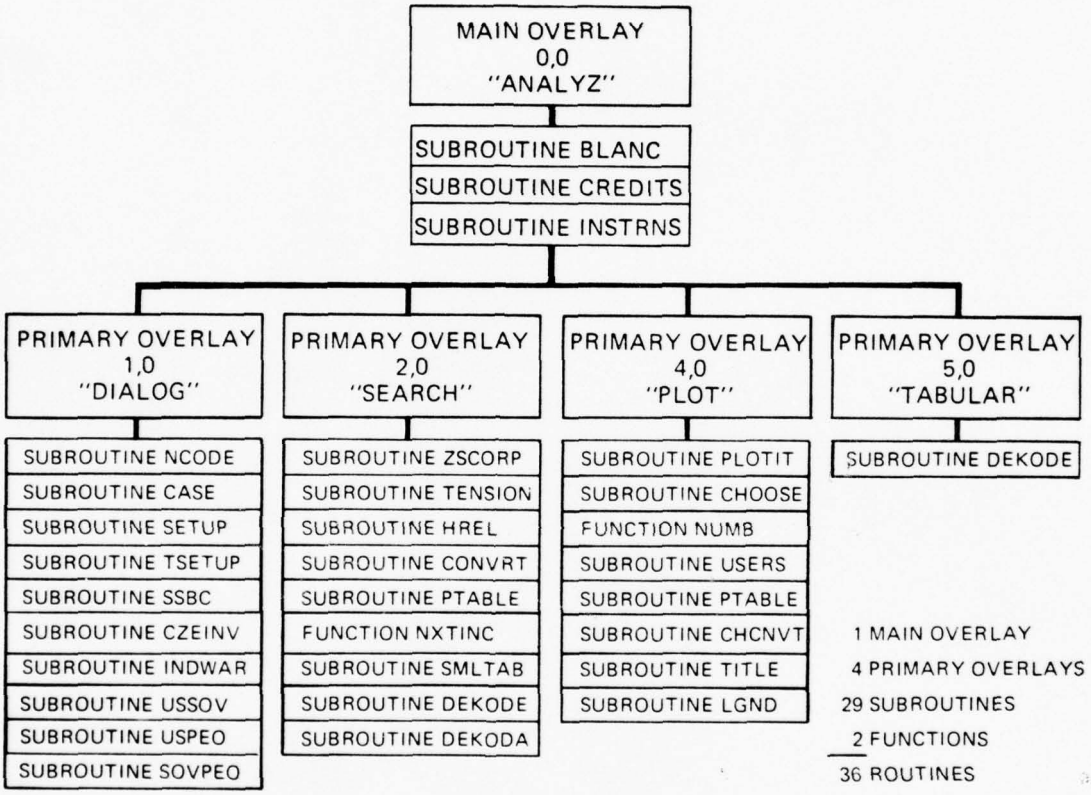

OVERLAY 5,0
Subroutine "DEKODE"
MONTHLY ACTIVITY
JAN, 1968 - DEC, 1968
*** USR <<<<<< TWO-WAY FLOW >>>>>> CZE ***
OVERLAY 2,0
DATE
TOTAL ACTIVITY NUMBER Z-SCORE PROB
COOPERATIVE ACTIVITY NUMBER Z-SCORE PROB
CONFLICTUAL ACTIVITY NUMBER Z-SCORE PROB
JAN 68 1 1.95 .10 1 1.95 .01 0 0.00 .01
FEB 68 2 3.89 .50 2 3.89 .10 0 0.00 .01
MAR 68 5 8.12 .70 5 8.12 .40 0 0.00 .01
APR 68 8 7.01 .70 4 3.32 .05 4 0.00 .01
MAY 68 21 11.38 .80 12 9.09 .40 9 11.72 .80
JUN 68 12 2.55 .40 8 2.87 .05 4 1.97 .10
JUL 68 37 7.79 .70 13 4.31 .10 24 12.41 .80
AUG 68 48 5.83 .60 20 5.37 .20 28 5.80 .60
SEP 68 41 3.33 .40 28 5.51 .20 13 1.66 .10
OCT 68 21 1.24 .05 16 2.03 .01 5 .38 .01
NOV 68 11 .41 .01 4 .10 .01 7 .67 .05
DEC 68 10 .32 .01 8 .71 .01 2 -.10 .01
Subroutine "PTABLE"
Subroutine "ZSCORP"
Subroutine "PTABLE"
Subroutine "ZSCORP"
Subroutine "PTABLE"
Subroutine "ZSCORP"
Subroutine "DEKODA"


OVERLAY 5,0
MONTHLY TENSION AND UNCERTAINTY
JAN, 1968 - DEC, 1968
Subroutine "DEKODE"
*** USR <<<<<< TWO-WAY FLOW >>>>>> CZE ***
OVERLAY 2,0
DATE TENSION Z-SCORE H-REL Z-SCORE
JAN 68 0.0 0.00 0.000 0.00
FEB 68 0.0 0.00 0.000 0.00
MAR 68 0.0 0.00 .431 0.00
APR 68 50.0 0.00 .561 6.57
MAY 68 42.9 4.35 .420 2.93
JUN 68 33.3 2.51 .603 3.76
JUL 68 64.9 4.66 .774 4.00
AUG 68 58.3 3.10 .772 3.17
SEP 68 31.7 1.26 .579 1.92
OCT 68 23.8 .80 .483 1.41
NOV 68 63.6 2.88 .531 1.54
DEC 68 20.0 .45 .352 .76
Subroutine "ZSCORP"
Subroutine "HREL"
Subroutine "ZSCORP"
Subroutine "TENSION"
Subroutine "DEKODA"


OVERLAY 5,0
MONTHLY COOPERATIVE ACTIVITY
JAN, 1968 - DEC, 1968
Subroutine "DEKODE"
*** USR <<<<<< TWO-WAY FLOW >>>>>> CZE ***
OVERLAY 2,0
DATE YLD CMNT CNSL APPR PRMS GRNT REWD AGRE RQST PROP
JAN 68 0 0 0 1 0 0 0 0 0 0
FEB 68 0 0 2 0 0 0 0 0 0 0
MAR 68 0 1 2 1 1 0 0 0 0 0
APR 68 0 0 2 1 0 0 1 0 0 0
MAY 68 0 1 10 1 0 0 0 0 0 0
JUN 68 0 0 2 1 3 0 0 2 0 0
JUL 68 2 0 6 1 1 0 0 0 2 1
AUG 68 3 0 6 2 2 0 0 6 0 1
SEP 68 3 1 18 0 0 1 0 4 1 0
OCT 68 1 1 12 0 0 1 0 0 1 0
NOV 68 0 0 0 0 0 1 0 2 1 0
DEC 68 0 0 6 0 0 1 1 0 0 0
Subroutine "DEKODA"

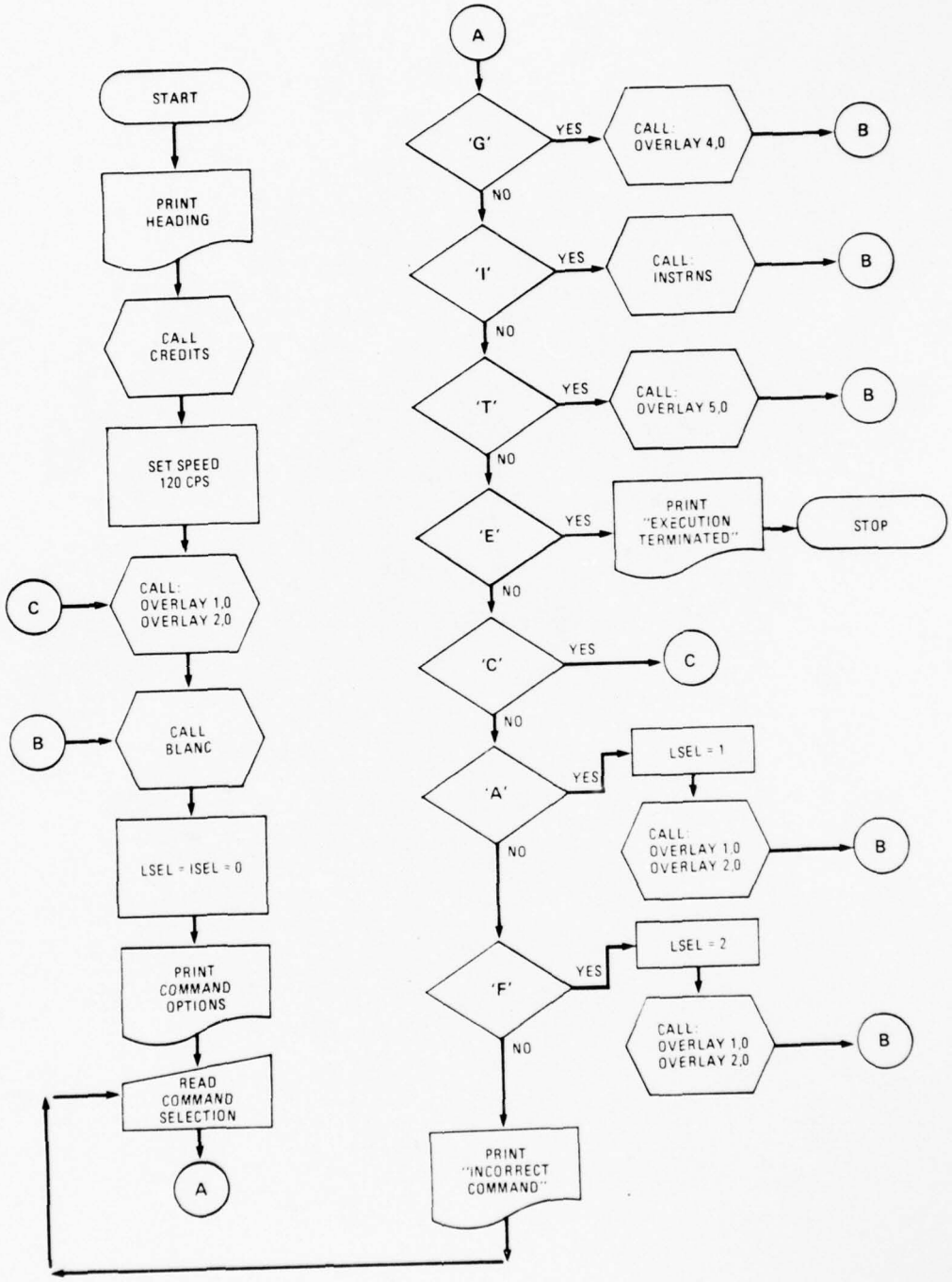

OVERLAY 5,0
Subroutine "DEKODE"
MONTHLY CONFLICTUAL ACTIVITY
JAN, 1968 - DEC, 1968
*** USR <<<<<< TWO-WAY FLOW >>>>>> CZE ***
OVERLAY 2,0
DATE RJCT ACUS PROT DENY DMND WARN THRT DEMO RDUC EXPL SEIZ FRCE
JAN 68 0 0 0 0 0 0 0 0 0 0 0 0
FEB 68 0 0 0 0 0 0 0 0 0 0 0 0
MAR 68 0 0 0 0 0 0 0 0 0 0 0 0
APR 68 0 2 0 0 0 1 0 0 1 0 0 0
MAY 68 0 7 0 1 0 1 0 0 0 0 0 0
JUN 68 0 1 1 2 0 0 0 0 0 0 0 0
JUL 68 3 8 1 1 4 2 0 3 2 0 0 0
AUG 68 1 12 1 1 2 2 1 4 0 0 4 0
SEP 68 0 8 1 0 2 0 0 0 1 0 1 0
OCT 68 0 2 0 0 2 0 0 1 0 0 0 0
NOV 68 0 4 0 0 0 0 0 2 0 0 0 1
DEC 68 0 2 0 0 0 0 0 0 0 0 0 0
Subroutine "DEKODA"

5.0 SYSTEM DIAGRAM

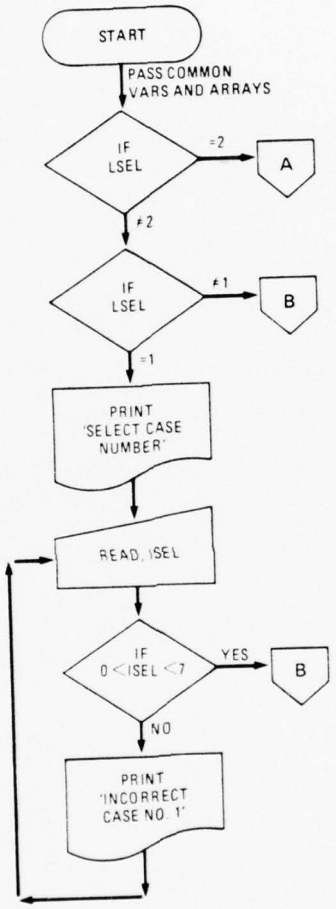


CRISIS EARLY WARNING PROTOTYPE SYSTEM
SEPTEMBER, 1976

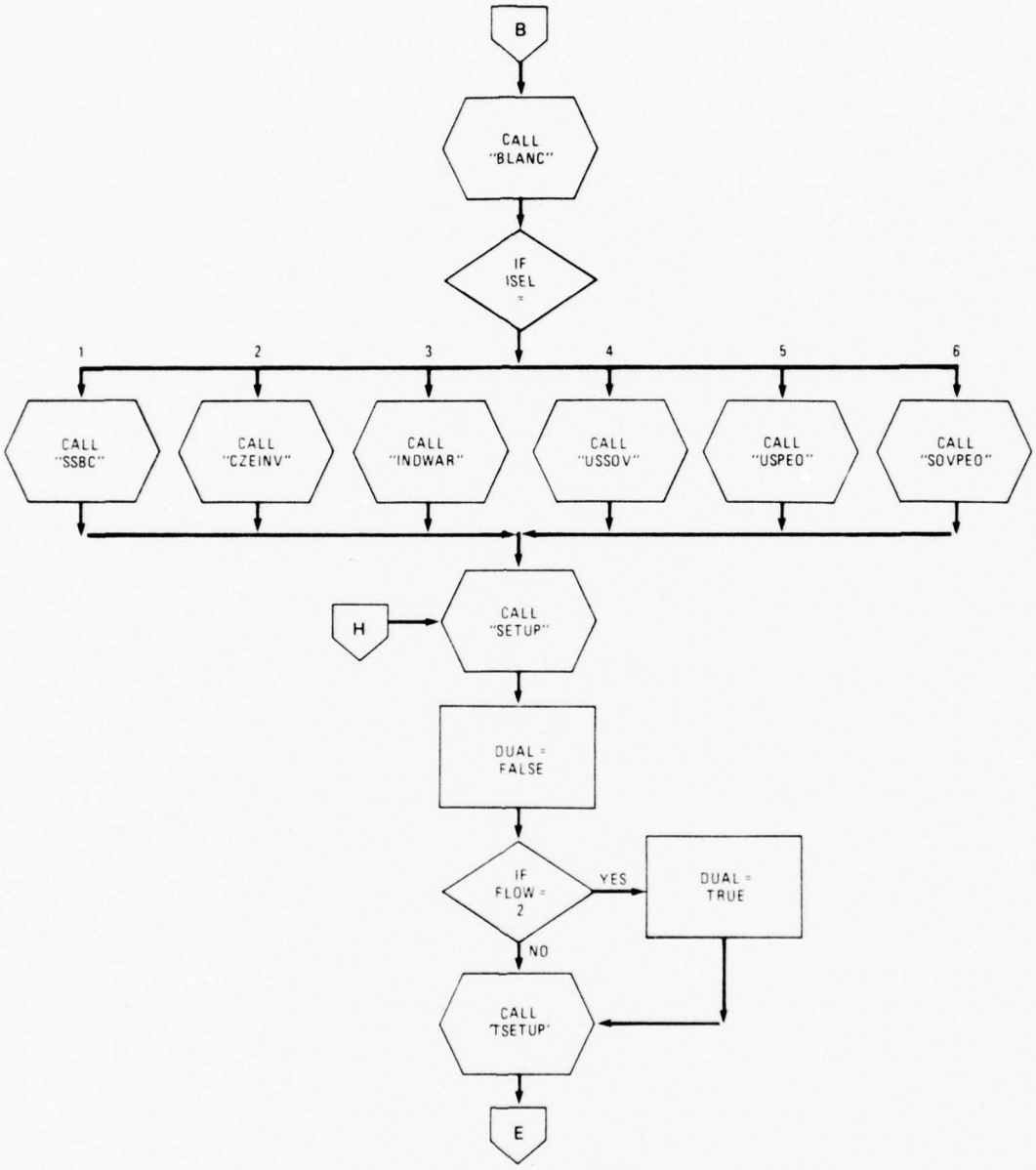
6.0 SYSTEM FLOW CHARTS



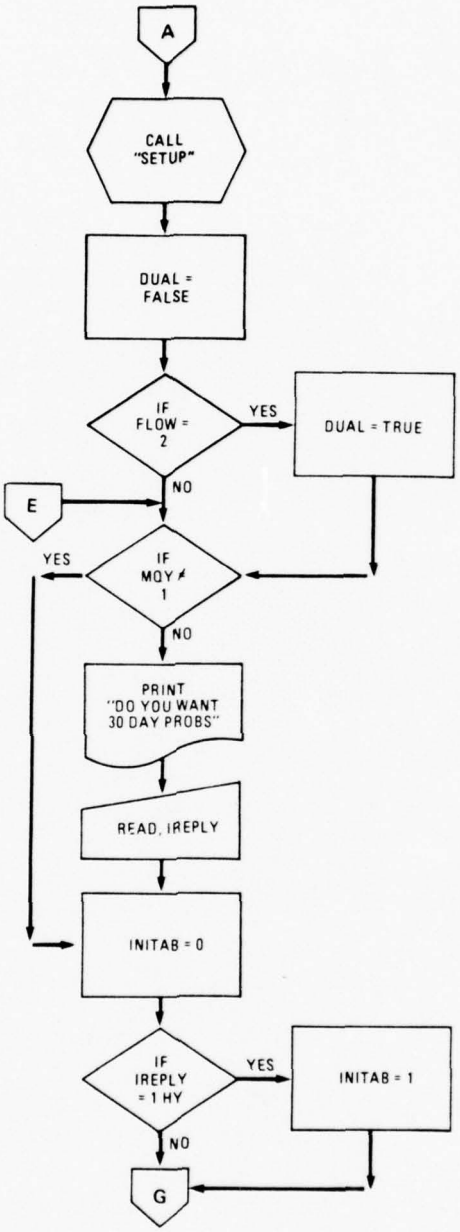
OVERLAY 0,0 (Main) "ANALYZ"



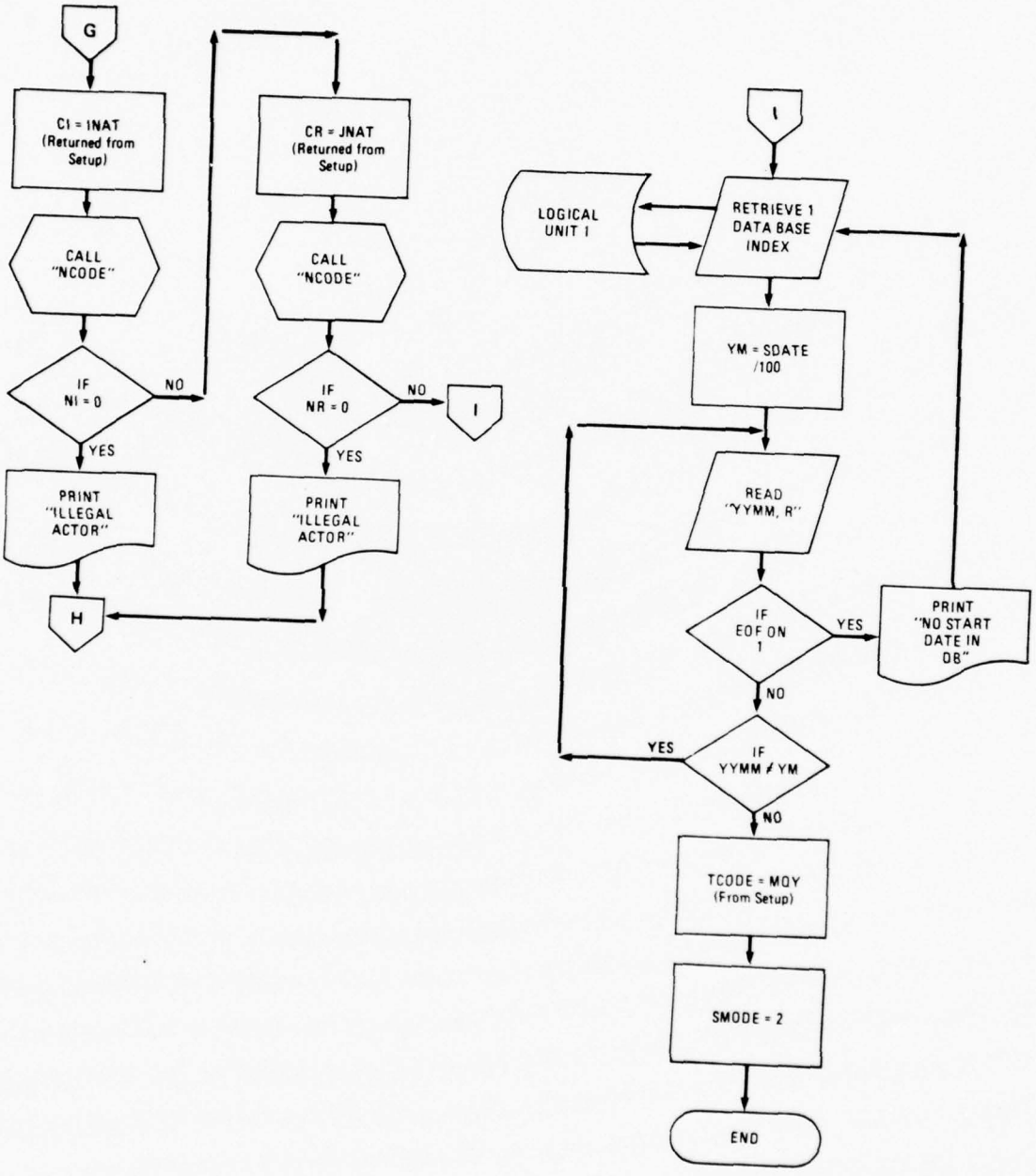
OVERLAY 1,0 (Primary) "DIALOG"



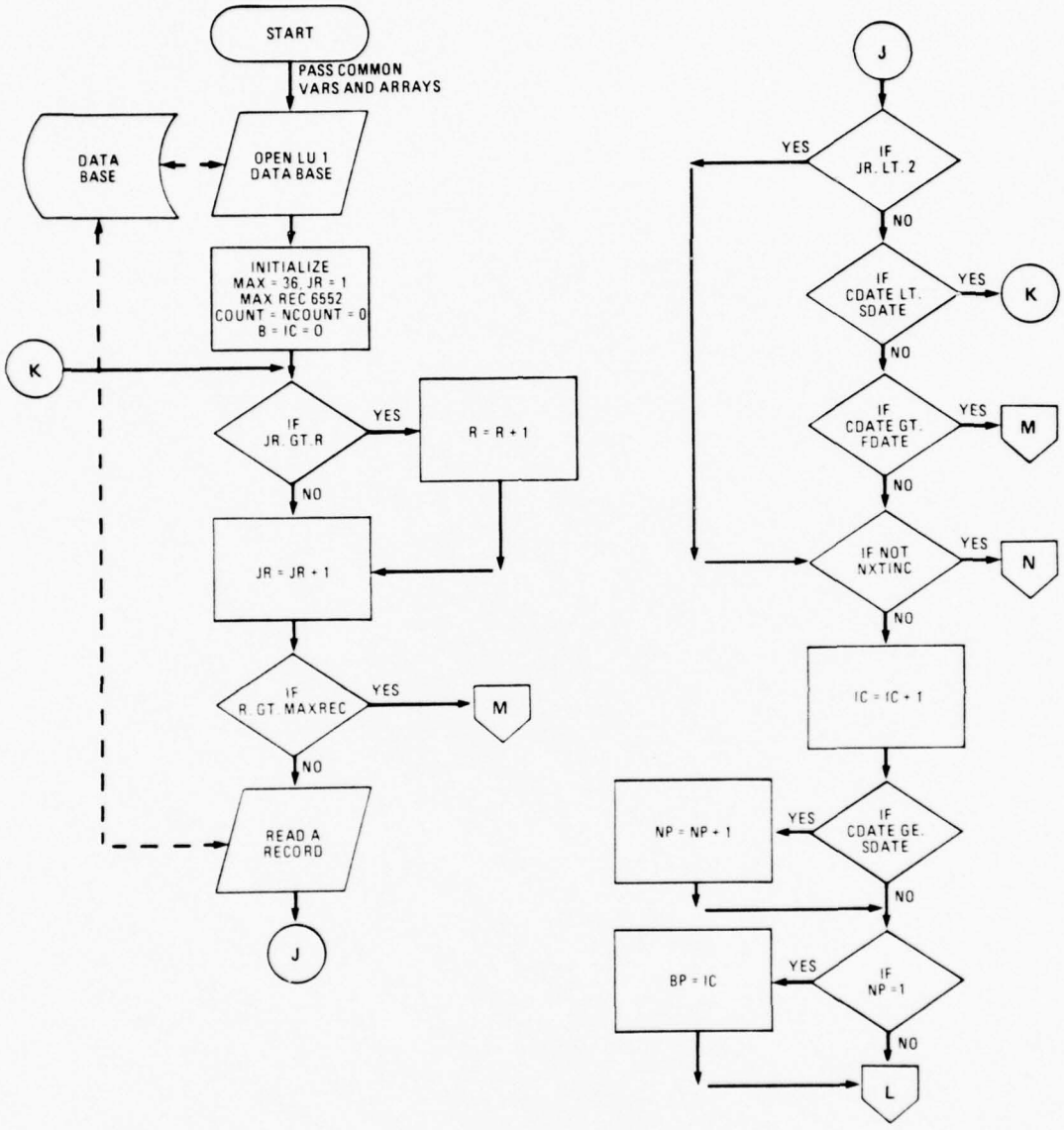
OVERLAY 1,0 (cont.)



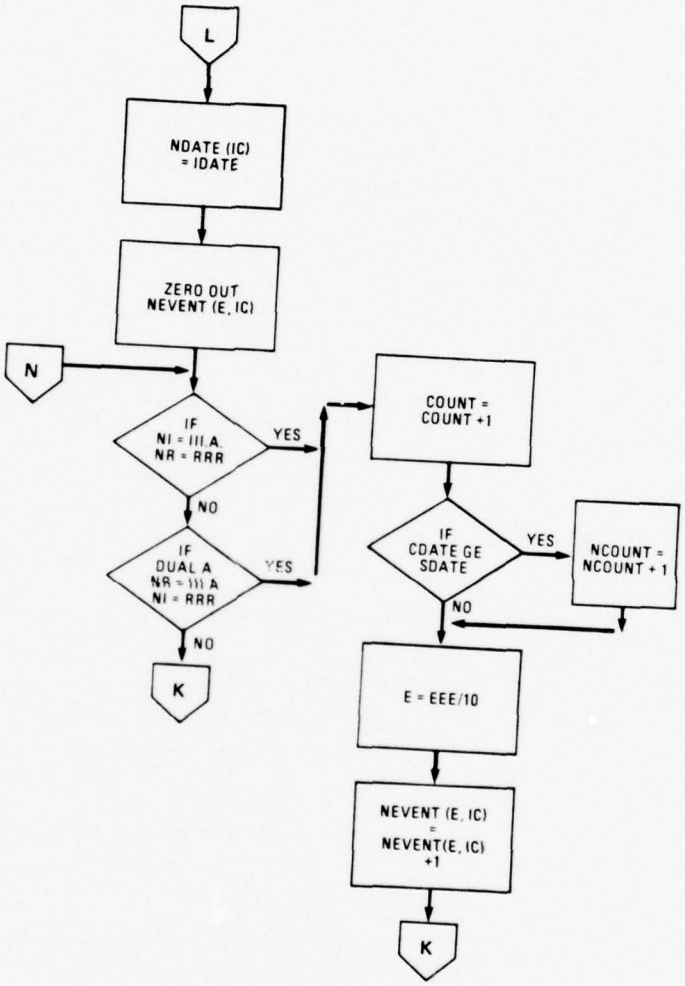
OVERLAY 1,0 (cont.)



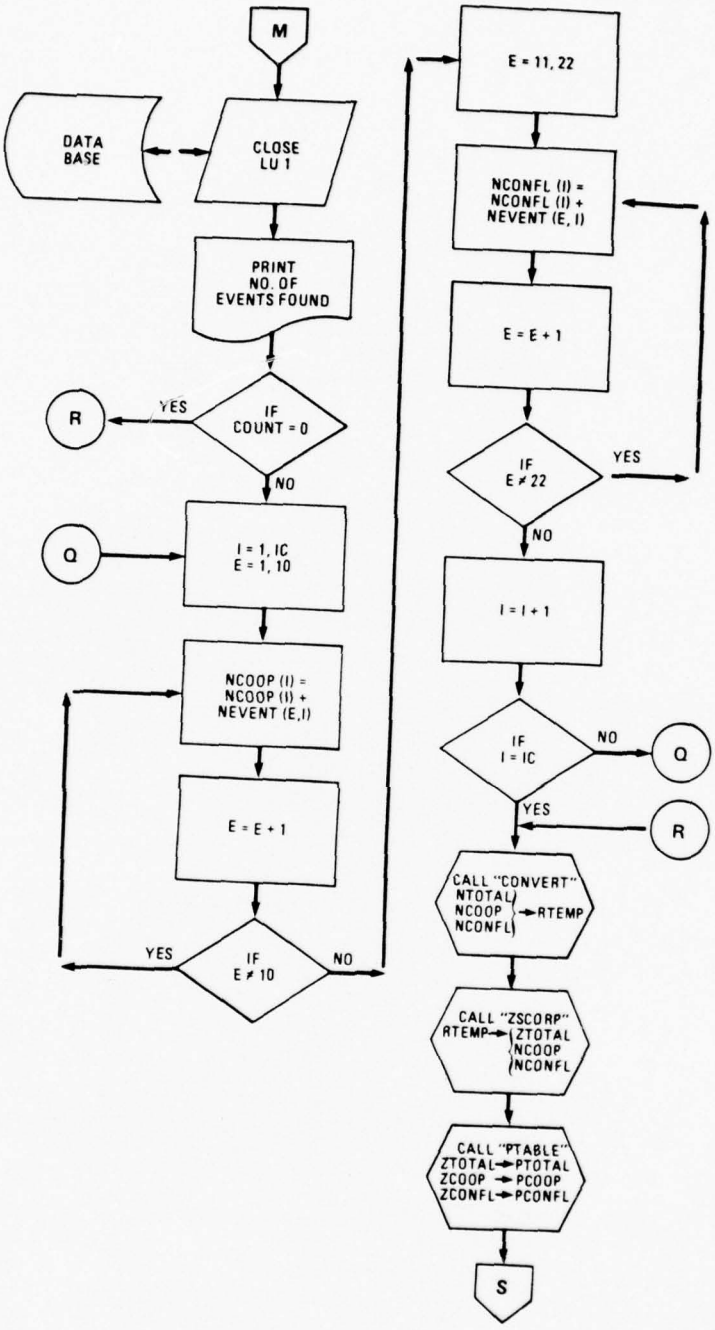
OVERLAY 1,0 (cont.)



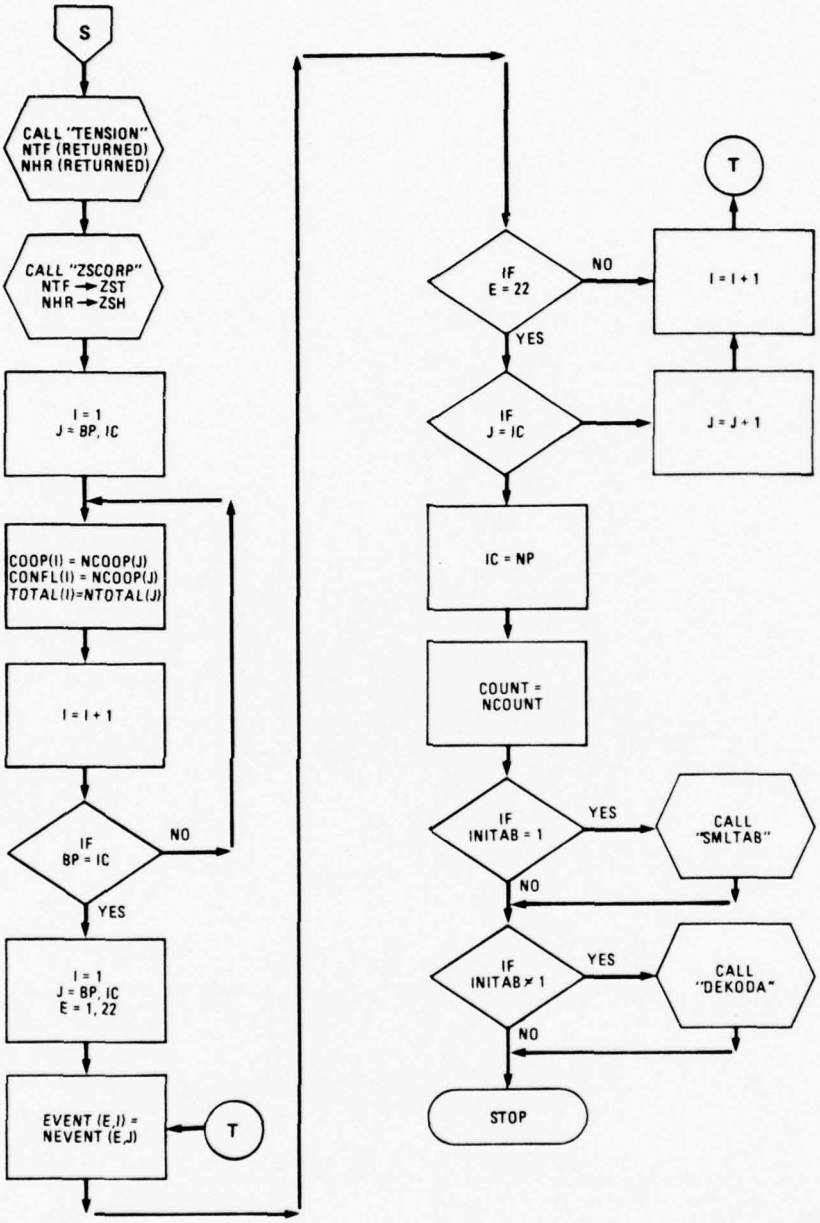
OVERLAY 2,0 (Primary) "SEARCH"



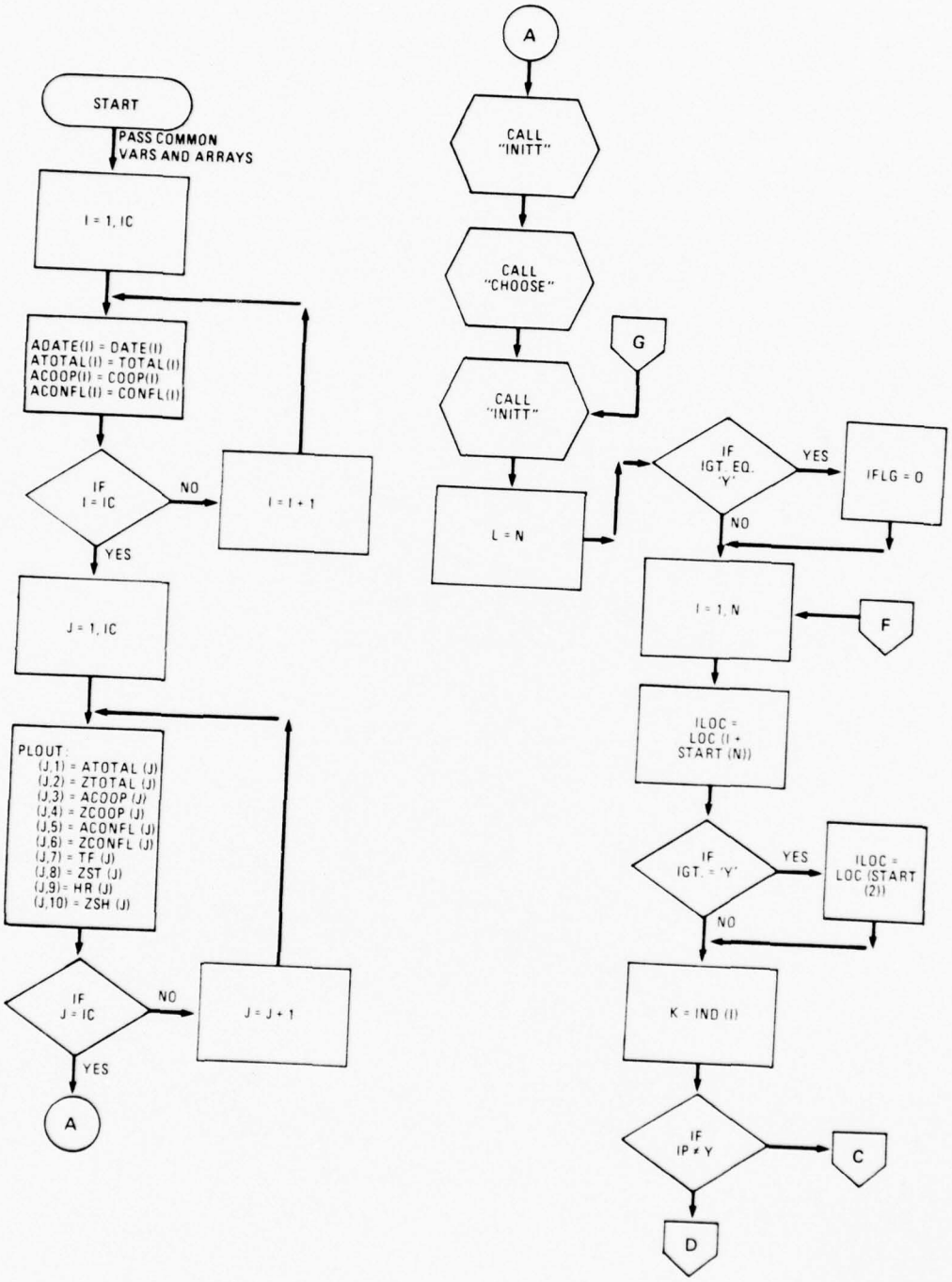
OVERLAY 2,0 (cont.)



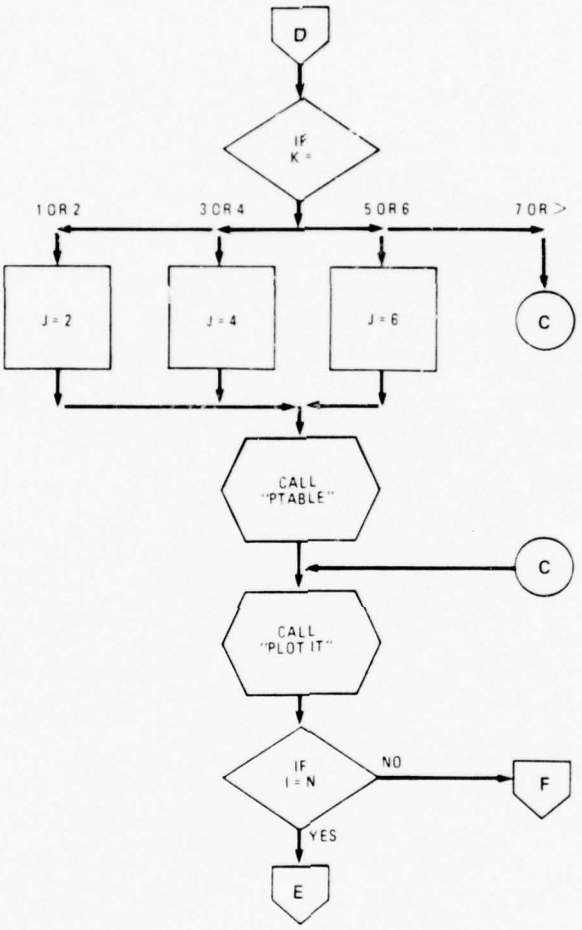
OVERLAY 2,0 (cont.)



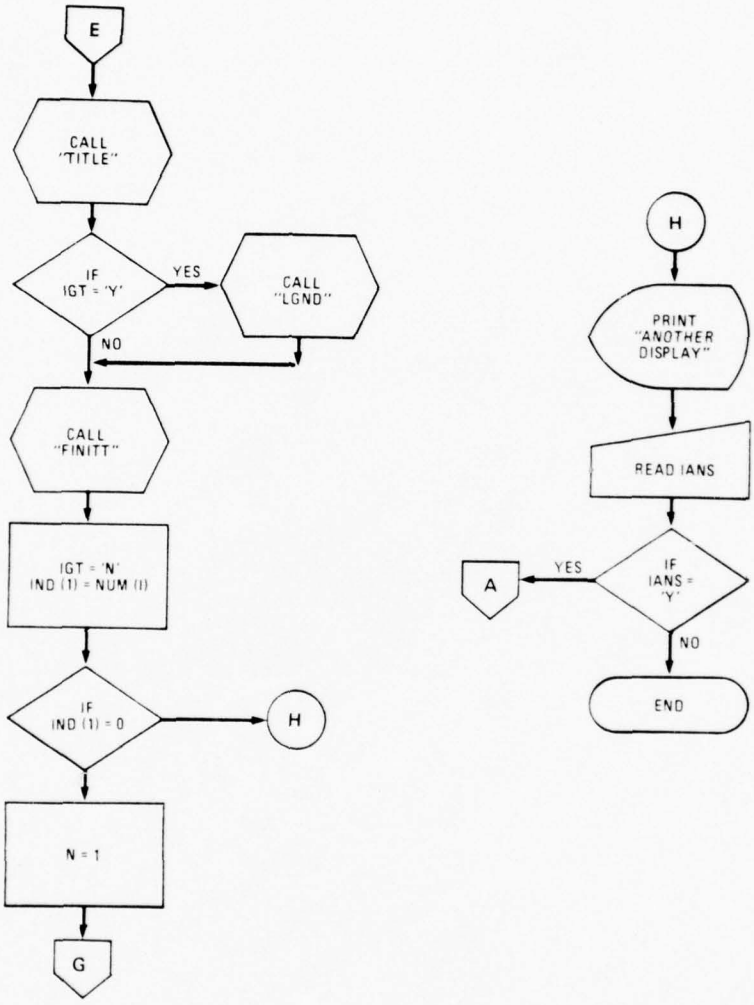
OVERLAY 2,0 (cont.)



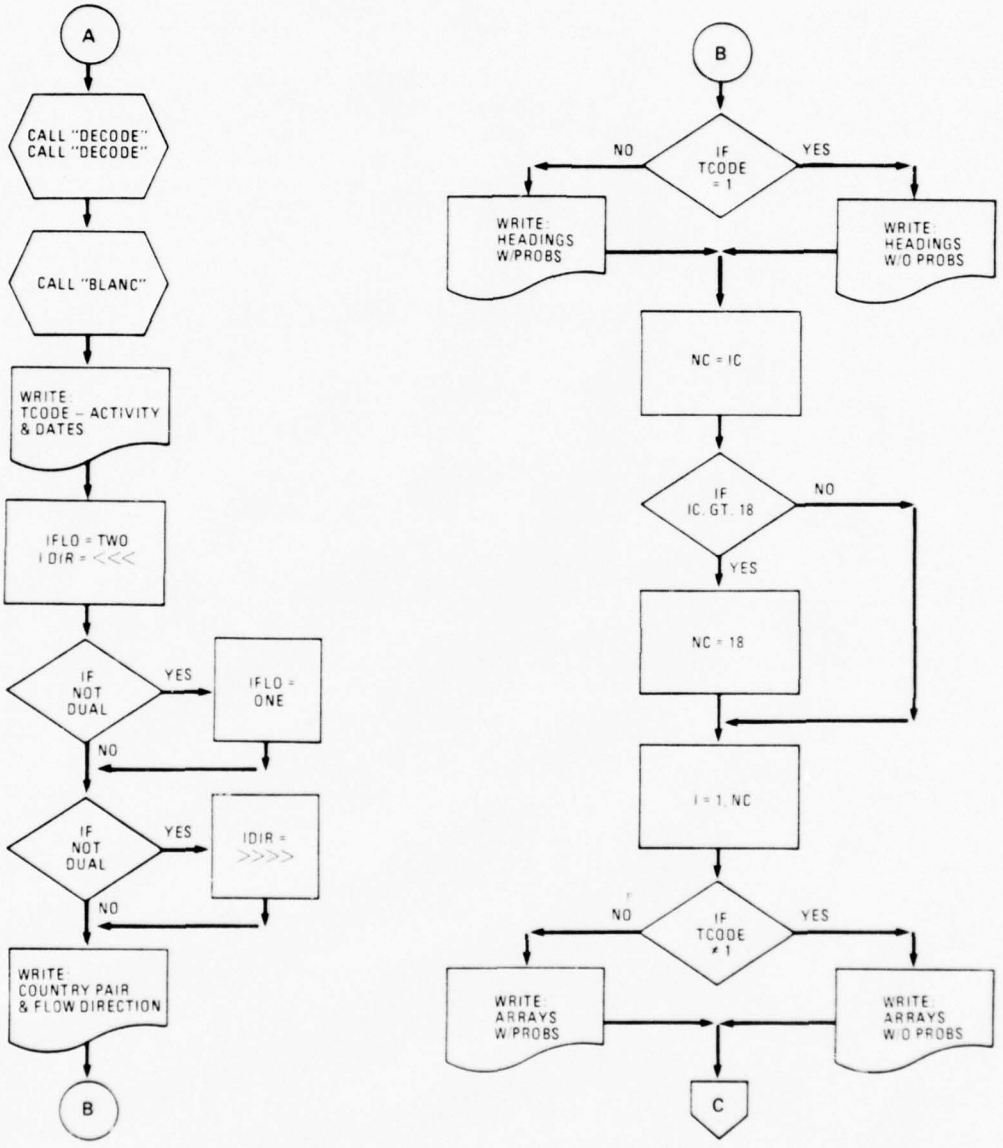
OVERLAY 4,0 (Primary) "PLOT"



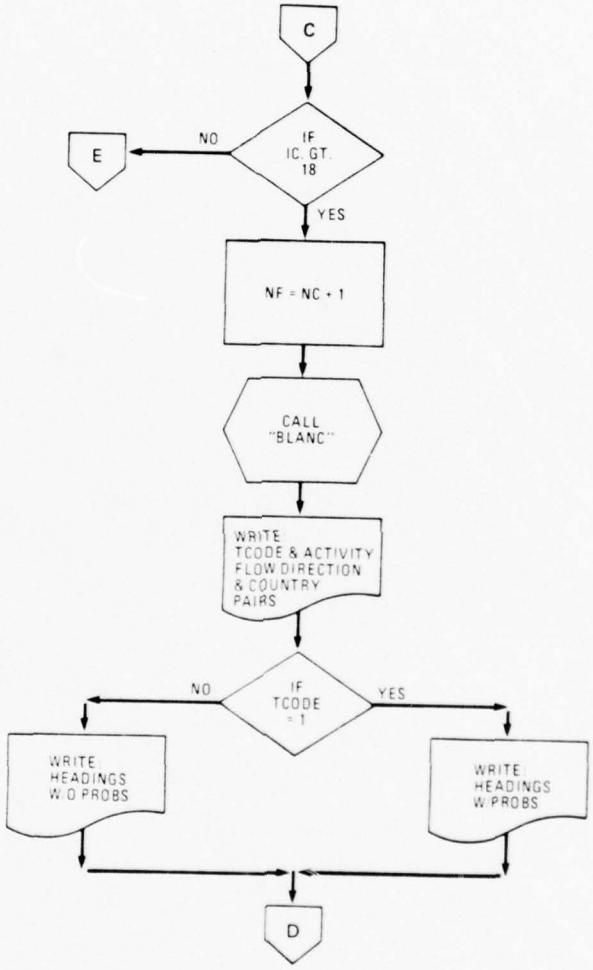
OVERLAY 4,0 (cont.)



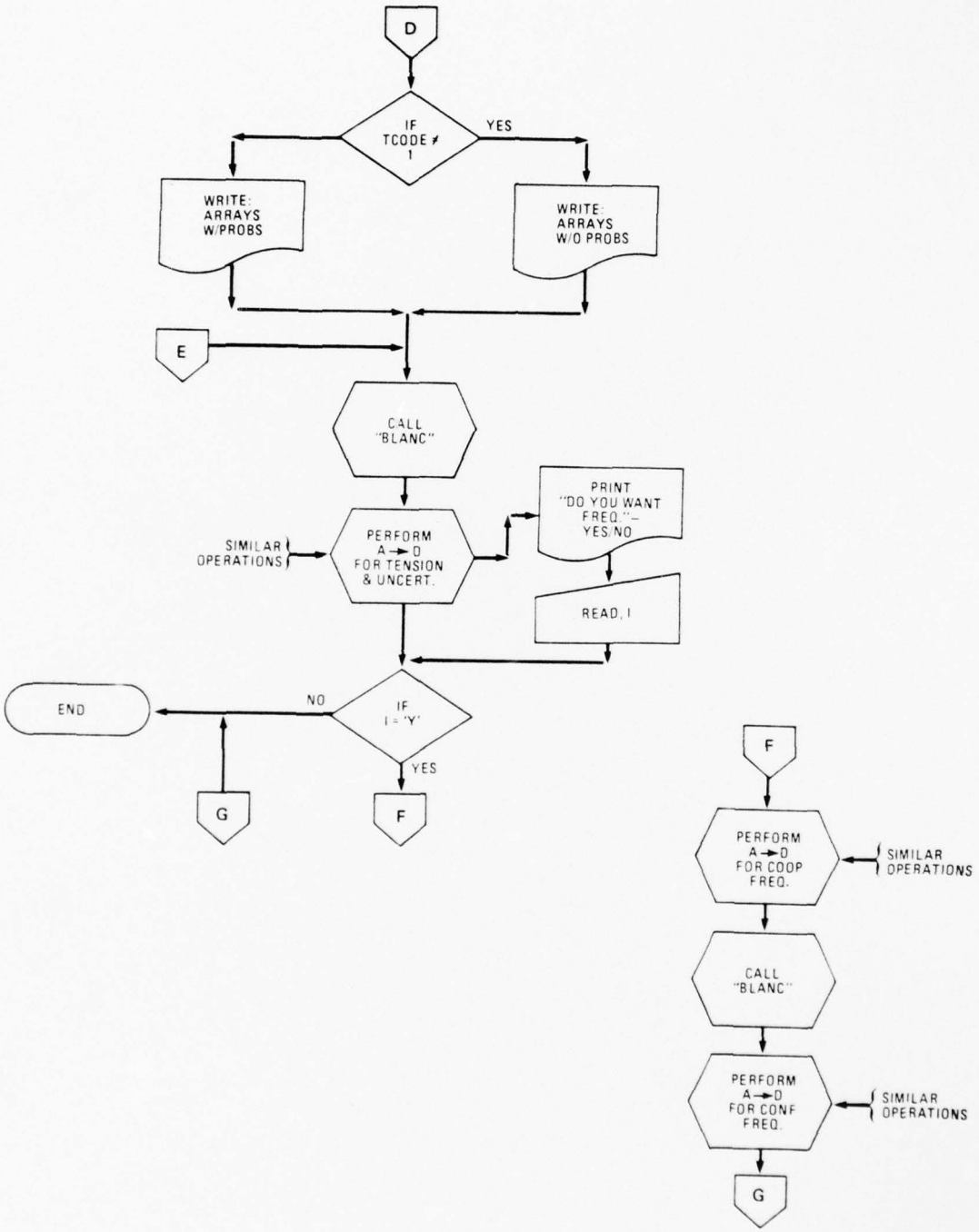
OVERLAY 4,0 (cont.)



OVERLAY 5,0 (Primary) "TABULAR"



OVERLAY 5,0 (cont.)



OVERLAY 5,0 (cont.)

7.0 DOCUMENTED FORTRAN SOURCE CODE

```
00100       OVERLAY(OVRLAY,0,0)
00110C
00120C      AUTHORS:  L.S.RANDALL (ORIGIONAL CODE)
00130C                J.J.ALLEN   (OVERLAY 4,0) ORIG.
00140C                J.F. WITTMEYER (MODIFICATIONS)
00150C
00160C      MAIN OVERLAY CALLS FOUR PRIMARIES
00170C        1. OVERLAY 1,0 - PARAMETER ENTRY + NATION CHECK
00180C        2. OVERLAY 2,0 - DB READ, ARRAY CALCULATIONS
00190C        3. OVERLAY 4,0 - ALL GRAPHICS
00200C        4. OVERLAY 5,0 - TABULAR O/P
00210C
00220C
00230C
00240C      SUBROUTINE CALLS IN MAIN OVERLAY
00250C    1) BLANC - UNIVERSAL ROUTINE TO CLEAR PAGE ON 4051
00260C    2) CREDITS - ROUTINE DISPLAYS PROGRAMMERS PURPOSE
00270C    3) INSTRNS - ROUTINE (SHOULD NOT BE IN MAIN) GIVES A QUICK
00280C                 DESCRIPTION OF HOW OVERLAY 4,0 WORKS...
00290C
00300C
00310       PROGRAM ANALYZ(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,
00320+       TAPE1,TAPE45=OUTPUT)
00330C
00340C    OBJECTIVE:  TO TABULATE THE FREQUENCY, TENSION, H-REL, AND
00350C    Z-SCORES OF EVENTS BETWEEN TWO COUNTRIES OVER A SPECIFIED
00360C    SPAN OF TIME BY DAY, MONTH, QUARTER, HALF-YEAR OR YEAR.
00370C
00380C    WEIS DATA ARE MAINTAINED IN A DIRECT ACCESS FILE (RUSC)
00390C    WITH RECORDS OF 18 CHARS EACH:
00400C       YYMMDDIIIEEERRRAAA
00410C              WHERE
00420C       YYMMDD - DATE
00430C
00440C       III    - INITIATOR
00450C       EEE    - EVENT
00460C       RRR    - RECIPIENT
00470C       AAA    - ARENA
00480C
00490C    A MONTH INDEX FILE (USCINDX) FOR THE WEIS DATA IS FORMATTED
00500C    AS A SEQUENTIAL FILE WITH RECORDS OF 10 CHARS EACH:
00510C       YYMM XXXXX
00520C              WHERE
00530C      YYMM - YEAR + MONTH
00540C       XXXXX - INDEX (RECORD NO) OF FIRST RECORD IN RUSC WHICH
00550C              INVOLVES THE GIVEN YEAR + MONTH.
00560C
00570C       COMMON LAYOUT
00580C    PARAM ---
00590C     CI = 'CHARACTER' INITIATOR
00600C     NI = ALPHA INITIATOR  (3CHRS)
00610C     ADI = 3 WD ALPHA INITIATOR
00620C     CR = 'CHARACTER' RECIPIENT 3 CHRS
00630C     NR = ALPHA RECIPIENT 3-CHRS
00640C     ADR = 3 WD ALPHA RECIPIENT
00650C     DUAL = FLOW DIRECTION  1=2-WAY, 0=ONE WAY
00660C     SDATE = I6 START DATE
00670C     FDATE = I6 FINAL DATE
00680C     TCODE = TIME INCREMENT (MONTH,QTR,YR)
00690C     SMODE = Z-SCORE MODE (PROGRESSIVE = 2)
00700C      R = ADDRESS OF BEGINNING SECTOR OF SDATE ON D.B.
00710C     ISPEED = BAUD RATE FOR PLOT-10 (1200)
00720C     INITAB = FLAG FOR 30 DAY PROBABILITES 1=YES,2=NO
00730C
00740       IMPLICIT INTEGER(A-Z)
00750C      DELTA=CDATE/100-IDATE/100
00760C
00770       COMMON /PARAM/ CI,NI,ADI(3),CR,NR,ADR(3),DUAL,
```

```
00780+                      SDATE,FDATE,TCODE,SMODE,R,ISPEED,INITAB
00790       COMMON /RESLT/ COUNT,IC,DATE(36),EVENT(22,36),
00800+                     COOP(36),CONFL(36),TOTAL(36),ATOTAL(36),
00810+                     ZTOTAL(36),ACOOP(36),ZCOOP(36),ACONFL(36),
00820+                     ZCONFL(36),TF(36),ZST(36),HR(36),ZSH(36),
00830+                     PTOTAL(36),PCOOP(36),PCONFL(36),RDATE(36)
00840C
00850C       COMMON LAYOUT
00860C    RESULT--
00870C     COUNT = NUMBER OF EVENTS FOUND
00880C     IC = NUMBER OF ARRAY ELEMENTS FILLED
00890C     DATE = I6 DATES FROM SDATE TO FDATE
00900C     EVENTS = 22X36 TOTAL EVENT CELLS
00910C     COOP,ACOOP = COOPERATIVE EVENTS ARRAY INTEGER + REAL
00920C     CONFL,ACONFL = CONFLICTIVE EVENTS ARRAY INTEGER + REAL
00930C     TOTAL,ATOTAL = TOTAL EVENTS ARRAY INTEGER + REAL
00940C     ZCOOP,ZCONFL,ZTOTAL = Z-SCORE DATA ARRAYS
00950C     PCOOP,PCONFL,PTOTAL = PROBABILITY ARRAYS
00960C     TF = TENSION RAW DATA ARRAY
00970C     ZST = TENSION Z-SCORE ARRAY
00980C     HR = HREL RAW DATA ARRAY
00990C     ZSH = HREL Z-SCORE ARRAY
01000C     RDATE = ARRAY CONTAINING (DD MMM) DATES
01010C
01020       COMMON/POINT/BP,LSEL,ISEL
01030C
01040C       COMMON LAYOUT
01050C    POINT--
01060C     BP = BEGINNING POINT EQUAL TO IC(1)
01070C     LSEL = FLOW SELECTION 1=ACTORS,2=FLOWS
01080C     ISEL = CASE NUMBER SELECTION
01090C
01100       COMMON /CHOICE/ IYOLD,IT,IP,KNTR,PROB(36),IGT,IFLG
01110C
01120C       COMMON LAYOUT
01130C    CHOICE--
01140C     IYOLD = RESET POINTER FOR SUBROUTINE USERS
01150C     IT = TYPE OF GRAPH G=GRAPHS,B=BAR CHARTS,S=SHADED BARS
01160C     IP = PROBABILITIES FLAG Y=YES,N=NO
01170C     KNTR = COUNTER FLAG FOR PROBABILITIES
01180C     PROB = ARRAY CONTAINING PROBABILITIES
01190C     IGT = MULTI-GRAPH FLAG  Y=YES,N=NO
01200C     IFLG = FLAG COUNTS # OF SINGLE AXIS PLOTS TO SIX
01210C
01220C
01230       CHARACTER*1 CXXX,REPLY,RESP
01240       CHARACTER*3 CI,CR
01250       CHARACTER ADI,ADR
01260       LOGICAL DUAL
01270        REAL ATOTAL,ZTOTAL,ACOOP,ZCOOP,ACONFL,ZCONFL,TF,ZST,HR,ZSH
01280        REAL PTOTAL,PCOOP,PCONFL
01290C
01300       DATA RESP/'Y'/
01310C
01320C
01330     50 CONTINUE
01340         CALL BLANC
01350       WRITE(6,1001)
01360   1001 FORMAT(3X,15(3H*),' CRISIS EARLY WARNING PROTOTYFE SYSTEM
01370+      15(3H*),/,/)
01380       CALL CREDITS
01390       READ 6001,I
01400C
01410       ISPEED = 120
01420     53 CALL OVERLAY('OVRLAY',1,0,0)
01430       CALL OVERLAY('OVRLAY',2,0,0)
01440     52 CALL BLANC
01450       LSEL=0
```

```
01460         ISEL=0
01470         WRITE(6,1002)
01480   1002 FORMAT(22X,3(3H*),' USER COMMANDS ',3(3H*),///,
01490+       25X,' CASE COMMANDS: ',//,
01500+       22X,' I = PLOT INSTRUCTIONS ',/,
01510+       22X,' G = GRAPHICAL OUTPUT ',/,
01520+       22X,' T = TABULAR OUTPUT ',/,
01530+       22X,' F = FLOW SELECTION ',/,
01540+       22X,' A = ACTOR SELECTION ',/)
01550         WRITE(6,1003)
01560   1003 FORMAT(//,25X,' OTHER COMMANDS: ',//,
01570+       22X,' C = CRISIS CASES ',/,
01580+        22X,' E = END EXECUTION ',//,
01590+        ////,22X,' SPECIFY USER COMMAND:  ',$)
01600    55 READ 6001,IREPLY
01610   6001 FORMAT(A1)
01620         IF(IREPLY.EQ.1HG) GOTO 56
01630         IF(IREPLY.EQ.1HI) GOTO 58
01640         IF(IREPLY.EQ.1HT) GOTO 57
01650         IF(IREPLY.EQ.1HE) GOTO 500
01660         IF(IREPLY.EQ.1HC) GOTO 53
01670         IF(IREPLY.EQ.1HA) GOTO 54
01680         IF(IREPLY.EQ.1HF) GOTO 59
01690   1005 FORMAT(/,/,2X,'INCORRECT COMMAND, PLEASE',
01700+       ' RE-ENTER (I,G,T,C,A,F OR E)',$)
01710         GOTO 55
01720     59 LSEL=2
01730         CALL OVERLAY('OVRLAY',1,0,0)
01740         CALL OVERLAY('OVRLAY',2,0,0)
01750         GOTO 52
01760     56 CALL OVERLAY('OVRLAY',4,0,0)
01770         GOTO 52
01780     57 CALL OVERLAY('OVRLAY',5,0,0)
01790         GOTO 52
01800     54 LSEL=1
01810         CALL OVERLAY('OVRLAY',1,0,0)
01820         CALL OVERLAY('OVRLAY',2,0,0)
01830         GOTO 52
01840     58 CALL INSTRNS
01850         GOTO 52
01860    500 CONTINUE
01870         WRITE(6,5003)
01880   5003 FORMAT(2X,*EXECUTION TERMINATED.*/////)
01890          END
01900C
01910C
01920C
01930           SUBROUTINE BLANC
01940C     OBJECTIVE: THIS ROUTINE WILL CLEAR THE PAGE ON ANY
01950C                TEKTRONIX TERMINAL. THIS IS DONE USING TWO CALLS
01960C                TO THE PLOT-10 SOFTWARE...
01970C
01980          CALL INITT(ISPEED)
01990          CALL FINITT(0,780)
02000          PRINT,/,/
02010           RETURN
02020           END
02030C
02040C
02050C
02060         SUBROUTINE CREDITS
02065C
02070C     OBJECTIVE:  TO INTRODUCE TO THE USER THE GENERAL INTENTION
02080C                 OF THE PROGRAM...ETC...
02085C
02090 100 FORMAT(13X,*THE CRISIS EARLY WARNING PROTOTYPE SYSTEM WAS *,
02100+ *DEVELOPED*,/,8X,*FOR THE ADVANCED RESEARCH PROJECTS AGENCY'S *,
02110+  *CYBERNETICS*,/,8X,*TECHNOLOGY OFFICE BY DECISIONS AND DESIGNS,
```

```
02120+  *INCORPORATED. *)
02130        WRITE(6,100)
02140 101 FORMAT(/,13X,*THE SYSTEM IS DESIGNED TO SIMULATE HOW *,
02150+ *QUANTITATIVE*,/,8X,*POLITICAL INDICATORS MIGHT HAVE CONTRIBUTED
02160+ *TO THE U.S.*,/,8X,*DEFENSE COMMUNITY'S ABILITY TO FORECAST *,
02170+ *A NUMBER OF IMPORTANT *,/,8X,*INTERNATIONAL CRISES.  IT IS *,
02180+ *ALSO DESIGNED TO ILLUSTRATE*,/,8X,*HOW THE GENERAL NATURE AND *,
02190+ *DIRECTION OF INTERNATIONAL AFFAIRS*,/,8X,*MAY BE MONITORED *,
02200+ *USING POLITICAL INDICATORS.  *,/,/,/,/)
02210        WRITE(6,101)
02220        RETURN
02230        END
02240C
02250C
02260C
02270        SUBROUTINE INSTRNS
02280C
02290C     OBJECTIVE : WHEN THE I OPTION IS CHOSEN THE USER GETS THIS
02300C                 INFORMATION DISPLAYED...IT BRIEFLY DESCRIBES THE
02310C                 OPERATION OF THE GRAPHICS OVERLAY....
02315C
02320 100 FORMAT(6X,*UPON INITIAL ENTRY INTO THE GRAPHICS SECTION, THE *
02330+ *POLITICAL*,/,6X,*INDICATOR MATRIX WILL BE DISPLAYED.  THIS *,
02340+ *MATRIX DESCRIBES *,/,6X,*EACH PLOT AVAILABLE TO THE USER AND *,
02350+ *IS NUMBERED ONE THROUGH *,/,6X,*TEN (TEN POSSIBLE PLOTS).*,/)
02360 200 FORMAT(6X,*IT IS THE USERS DECISION WHICH PLOTS ARE TO BE *,
02370+ *DISPLAYED, *,/,6X,*HOW MANY, WHAT PLOT TYPE (BAR OR GRAPH), AND
02380+ *THE DESIRED*,/,6X,*POSITION ON THE CRT.  THE FOLLOWING ARE VALID
02390+ *RESPONSES TO*,/,6X,*QUERIES WHICH IMMEDIATELY FOLLOW THE *,
02400+ *INDICATOR MATRIX: *,/)
02410 300 FORMAT(11X,*1)  DO YOU WANT MORE THAN ONE PLOT PER GRAPH (Y OR
02420 400 FORMAT(6X,*--THE 'Y' (YES) ANSWER WILL PROVIDE THE USER WITH *
02430+ *CHOICES*,/,8X,*AVAILABLE FOR MORE THAN ONE GRAPH PLOTTED ON A *,
02440+ *SINGLE*,/,8X,*PAIR OF AXIS.*,/)
02450 500 FORMAT(6X,*--THE 'N' (NO) ANSWER WILL LEAD THE USER THROUGH *,
02460+ *A GROUP OF*,/,8X,*QUESTIONS THAT WILL ALLOW HIM TO GENERATE FROM
02470+ *ONE TO SIX*,/,8X,*SIMULTANEOUS PLOTS, EACH HAVING ITS OWN AXIS.
02480 600 FORMAT(6X,*AFTER ANY GRAPH IS DISPLAYED THE SYSTEM AWAITS THE
02490+ *INPUT OF*,/,6X,*A 'RETURN' TO CONTINUE.*,/,/)
02500 605 FORMAT(31X,*- OR - *,/,/)
02510 700 FORMAT(6X,*THE USER MAY SPECIFY ANY OF THE FOLLOWING PLOT *,
02520+ *NAMES OR*,/,6X,*NUMBERS TO CREATE A BLOW-UP OF AN INDIVIDUAL *,
02530+ *GRAPH. *,/)
02540 800 FORMAT(10X,*1    OR    TOT          6    OR    CONZ*,/,
02550+ 10X,*2    OR    TOTZ         7    OR    TENZ*,/,
02560+ 10X,*3    OR    COO          8    OR    TENZ*,/,
02570+ 10X,*4    OR    COOZ         9    OR    UNC*,/,
02580+ 10X,*5    OR    CON         10    OR    UNCZ*,/)
02590 810 FORMAT(11X,*2)  ANOTHER DISPLAY (Y OR N) *,/)
02600 820 FORMAT(6X,*--THE 'Y' (YES) ANSWER WILL CAUSE THE INDICATOR *,
02610+ *MATRIX TO BE*,/,8X,*RE-DRAWN. *,/,/,
02620+ 6X,*--THE 'N' (NO) ANSWER WILL TERMINATE THIS OVERLAY AND *,/,
02630+ 8X,*RE-INITIATE THE 'SPECIFY USER COMMAND' SECTION.*,/,/)
02640 10 FORMAT(20X,3(3H*),' PLOTTER INSTRUCTIONS ',3(3H*),/,/)
02650 830 FORMAT(/,/,/,/)
02660   840 FORMAT(/,/)
02665C
02670        CALL BLANC
02680        WRITE(6,10)
02690        WRITE(6,100)
02700        WRITE(6,200)
02710        WRITE(6,300)
02720        WRITE(6,400)
02730        WRITE(6,500)
02740        WRITE(6,830)
02750        READ(5,6001)I
02760  6001 FORMAT(A1)
02770        CALL BLANC
```

```
02780         WRITE(6,840)
02790         WRITE(6,600)
02800         WRITE(6,605)
02810         WRITE(6,700)
02820         WRITE(6,800)
02830         WRITE(6,810)
02840         WRITE(6,820)
02850         WRITE(6,840)
02860         READ(5,6001)I
02870         RETURN
02880         END
02890C
02900C
02910C
02920          OVERLAY(OVRLAY,1,0)
02930C
02940C    OBJECTIVE:
02950C     TO EXECUTE THE SELECTION OF PARAMETERS. THIS OVERLAY DOES:
02960C      1) PRINTS CASES AND READS SELECTIONS
02970C      2)  PRINTS SELECTED CASE HEADINGS AND ASKS FOR COUNTRY PAIR
02980C      3) ASKS FOR FLOW DIRECTION
02990C      4) ASKS FOR TIME INCREMENT
03000C      5) ASKS FOR START DATE (SDATE) AND FINISH DATE (FDATE)
03010C      6) READS DATE INDEX FILE + CHECKS FOR VALID DATE
03020C      7) READS NATION FILE + CHECKS FOR VALID ID'S
03030C
03040         PROGRAM DIALOG
03050C
03060C    SUBROUTINES--
03070C     1) NCODE - RETRIEVES NUMERIC ID AND ALPHA DESCRIPTOR OF ACTORS
03080C     2) CASE  - PRINTS OUT CASE MENU AND SELECTION
03090C     3) SETUP - ASKS FOR COUNTRY PAIRS, FLOW DIRECTION + TIME INCRE
03100C     4) TSETUP - ASKS FOR SDATE AND FDATE
03110C     5) SSBC,CZEINV,INDWAR,USSOV,USPEO,SOVPEO - ALL PRINT CASE HEAD
03120C
03130C
03140         IMPLICIT INTEGER(A-Z)
03150C
03160         COMMON /PARAM/ CI,NI,ADI(3),CR,NR,ADR(3),DUAL,
03170+                       SDATE,FDATE,TCODE,SMODE,R,ISPEED,INITAB
03180         COMMON/POINT/BP,LSEL,ISEL
03190C
03200C    MAJOR PARAMETERS:
03210C       LSEL - SET IN MAIN 0.0
03220C                1 = IF ACTOR SELECTION
03230C                2 = FLOW ONLY SELECTION
03240C       ISEL - 1,2,3,4,5 OR 6 CASE SELECTION
03250C       DUAL - TRUE = 2-WAY FLOW
03260C              FALSE = 1 WAY FLOW
03270C       INAT,NI,CI = INITIATOR
03280C       JNAT,NR,CR = RECIPIENT
03290C       FLOW -  0 = <<<<<< REVERSE DIRECTION
03300C                1 = >>>>>> OK AS IS
03310C                2 = TWO WAY FLOW
03320C       SDATE = START(BEGINNING) DATE
03330C       FDATE = END (FINAL) DATE
03340C       SMODE = ALWAYS SET TO (2) PROGRESSIVE Z-SCORES
03350C       TCODE,MQY = TIME INCREMENT SELECTED
03360C      R - RECORD NUMBER OF SDATE IN DATA BASE
03370C
03380         CHARACTER*3 CI,CR,ALL,INAT,JNAT
03390         CHARACTER*1 CXXX,REPLY,RESP
03400         CHARACTER ADI,ADR
03410         CHARACTER*10 ANY(3)
03420         LOGICAL DUAL
03430C
03440         DATA RESP/'Y'/
03450         DATA ANY/'ANY COUNTR','Y',' '/
```

```
03460        DATA ALL/'ALL'/
03470        CALL BLANC
03480        IF(LSEL.EQ.2) GOTO 205
03490        IF(LSEL.NE.1) GOTO 201
03500        WRITE(6,2001)
03510   2001 FORMAT(/,/,2X,'PLEASE SELECT ONE CASE NUMBER :',↑)
03520    203 READ(5,)ISEL
03530        IF(ISEL.GE.1.A.ISEL.LE.6) GOTO 202
03540        WRITE(6,2002)
03550   2002 FORMAT(2X,$INCORRECT CASE NUMBER, PLEASE RE-ENTER $,↑)
03560        GOTO 203
03570    201 CONTINUE
03580        CALL CASE(ISEL)
03590    202 CALL BLANC
03600        GOTO(301,302,303,304,305,309),ISEL
03610    301 CALL SSBC
03620        GOTO 306
03630    302 CALL CZEINV
03640        GOTO 306
03650    303 CALL INDWAR
03660        GOTO 306
03670    304 CALL USSOV
03680        GOTO 306
03690    305 CALL USPEO
03700        GOTO 306
03710    309 CALL SOVPEO
03720    306 CONTINUE
03730    308 CONTINUE
03740        CALL SETUP(ISEL,INAT,JNAT,FLOW,MQY)
03750        DUAL=.FALSE.
03760        IF(FLOW.EQ.2)DUAL=.TRUE.
03770        CALL TSETUP(SDATE,FDATE)
03780        GOTO 204
03790   205 CALL SETUP(ISEL,INAT,JNAT,FLOW,MQY)
03800        DUAL=.FALSE.
03810        IF(FLOW.EQ.2)DUAL=.TRUE.
03820    204 IF(MQY.NE.1) GOTO 307
03830        WRITE(6,3001)
03840  3001 FORMAT(/,2X,$DO YOU DESIRE 30 DAY PROBABLISTIC $,
03850+       $FORCASTS (Y OR N) : $,↑)
03860        READ(5,3003)IREPLY
03870    307 INITAB=0
03880        IF(IREPLY.EQ.1HY) INITAB=1
03890    100 CONTINUE
03900        WRITE(6,3004)
03910   3004 FORMAT(20X,' . . . P R O C E S S I N G . . . ',)
03920   3003 FORMAT(A1)
03930        CI=INAT
03940C
03950        IF(CI.NE.ALL) GO TO 105
03960        NI=0
03970        DO 104 I=1,3
03980        ADI(I)=ANY(I)
03990    104 CONTINUE
04000        GO TO 110
04010    105 CONTINUE
04020C
04030        CALL NCODE(CI,NI,ADI)
04040C
04050        IF(NI.NE.0) GO TO 110
04060        WRITE(6,9001)INAT
04070   9001 FORMAT('   $** ILLEGAL ACTOR - ',A3,' **$ ',/)
04080        GOTO 308
04090    110 CONTINUE
04100C
04110   1003 FORMAT(' INITIATOR IS ',3A10/)
04120C
04130    120 CONTINUE
```

```
04140         CR=JNAT
04150C
04160         IF(CR.NE.ALL) GO TO 125
04170         NR=0
04180         DO 124 I=1,3
04190         ADR(I)=ANY(I)
04200    124  CONTINUE
04210         GO TO 130
04220    125  CONTINUE
04230C
04240         CALL NCODE(CR,NR,ADR)
04250C
04260         IF(NR.NE.0) GO TO 130
04270         WRITE(6,9001)JNAT
04280         GOTO 308
04290    130  CONTINUE
04300C
04310   1005  FORMAT(' RECIPIENT IS ',3A10/)
04320C
04330   1006  FORMAT(' CAN ROLES BE REVERSED',↑)
04340C
04350    160  CONTINUE
04360C
04370         CALL PFUR(3HRET,1,6HRWINDX,0)
04380         REWIND 1
04390C
04400         YM=SDATE/100
04410C
04420    165  CONTINUE
04430         READ(1,3002,END=166) YYMM,R
04440   3002  FORMAT(I4,I6)
04450         IF(YYMM.NE.YM) GO TO 165
04460         GO TO 167
04470C
04480    166  CONTINUE
04490         WRITE(6,9005)
04500   9005  FORMAT(/' *** START DATE DOES NOT APPEAR IN WEIS DATA.'/)
04510         GO TO 160
04520C
04530    167  CONTINUE
04540C
04550    170  CONTINUE
04560         TCODE=MQY
04570          IF(TCODE.GE.1.AND.TCODE.LE.3) GO TO 180
04580C
04590    180  CONTINUE
04600C
04610         IF(FDATE.GE.SDATE) GO TO 190
04620    190  CONTINUE
04630C
04640    191  CONTINUE
04650   1010  FORMAT(/' SELECT MODE FOR COMPUTING Z-SCORES ',
04660+              '(1=NORMAL,2=PROGRESSIVE):  ',↑)
04670   2004  FORMAT(I1)
04680          SMODE = 2
04690C
04700   9004  FORMAT(' *** ILLEGAL MODE--TRY AGAIN.'/)
04710C
04720         END
04730C
04740C
04750C
04760         SUBROUTINE NCODE(ACODE,NID,DESCR)
04770C
04780C    OBJECTIVE:  TO DETERMINE THE NUMERIC ID AND THE DESCRIPTOR
04790C    (FULL NAME) FOR THE COUNTRY SPECIFIED BY A GIVEN ALPHABETIC
04800C    CODE.
04810C
```

```
04820C    ACODE - THREE CHAR ALPHABETIC CODE.
04830C    NID   - THREE DIGIT NUMERIC ID (SET BY NCODE).  SET TO 0
04840C            IF ACODE IS NOT A VALID CODE.
04850C    DESCR - THIRTY CHAR DESCRIPTOR (SET BY NCODE).
04860C
04870C    COUNTRY ID DATA (NATION) ARE ASSUMED FORMATTED IN
04880C    RECORDS OF 38 CHARS EACH:
04890C       NNN AAA DDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
04900C                <------------ 30 ------------>
04910C           WHERE
04920C       NNN     - NUMERIC ID
04930C       AAA     - ALPHABETIC CODE
04940C       DD...D - DESCRIPTOR (FULL COUNTRY NAME)
04950C
04960         IMPLICIT INTEGER(A-Z)
04970C
04980         CHARACTER*1 S1,S2
04990         CHARACTER*3 AAA,ACODE
05000         INTEGER DDD(3),DESCR(3)
05010C
05020C
05030         CALL PFUR(3HRET,1,6HNATION,0)
05040C
05050         NID=0
05060C
05070    100 CONTINUE
05080         READ(1,1001,END=200) NNN,S1,AAA,S2,(DDD(I),I=1,3)
05090   1001 FORMAT(I3,A1,A3,A1,3A10)
05100         IF(AAA.EQ.ACODE) GO TO 150
05110         GO TO 100
05120C
05130    150 CONTINUE
05140         NID=NNN
05150         DO 160 I=1,3
05160         DESCR(I)=DDD(I)
05170    160 CONTINUE
05180C
05190    200 CONTINUE
05200          RETURN
05210          END
05220C
05230C
05240C
05250         SUBROUTINE CASE(ISEL)
05260C
05270C     OBJECTIVE:
05280C      TO DISPLAY 6 CASE MENU AND RETURN 'ISEL' VARIABLE. 'ISEL'
05290C      SHOULD BE AN INTEGER FROM 1 TO 6 DENOTING THE CASE NUMBER
05300C      SELECTED...
05310C
05320         WRITE(6,1001)
05330   1001 FORMAT(4X,*THE FOLLOWING INTERNATIONAL CRISIS *,
05340+       *CASES ARE NOW AVAILABLE *,/,4X,*ON THE DEMONSTRATION *,
05350+       *SYSTEM : *,/)
05360         WRITE(6,1002)
05370   1002 FORMAT(6X,*1.*,2X,*SINO - SOVIET BORDER CLASHES,*,
05380+       /,10X,*JANUARY , 1967 AND MARCH , 1969 *,/)
05390         WRITE(6,1003)
05400   1003 FORMAT(6X,*2.*,2X,*CZECHOSLOVAKIAN INVASION, *,
05410+       /,10X,*AUGUST , 1968 *,/)
05420         WRITE(6,1004)
05430   1004 FORMAT(6X,*3.*,2X,*INDO - PAKISTANI WAR, *.
05440+       /,10X,*NOVEMBER , 1971 *,/)
05450         WRITE(6,1005)
05460   1005 FORMAT(4X,*THE SYSTEM ALSO CONTAINS THE FOLLOWING OTHER CASE
05470         WRITE(6,1006)
05480   1006 FORMAT(6X,*4.*,2X,*U.S. - SOVIET UNION, *,
05490+       /,10X,*JANUARY , 1966 - DECEMBER , 1975 *,/)
```

```
05500         WRITE(6,1007)
05510   1007 FORMAT(6X,$5.$,2X,$U.S. - PEOPLES REPUBLIC OF CHINA, $,
05520+        /,10X,$JANUARY , 1966 - DECEMBER , 1975 $,/)
05530         WRITE(6,1010)
05540   1010 FORMAT(6X,$6.$,2X,$SOVIET UNION - PEOPLES REPUBLIC OF CHINA,
05550+              /,10X,$JANUARY , 1966 - DECEMBER , 1975 $,/)
05560    10 WRITE(6,1008)
05570   1008 FORMAT(/,4X,$PLEASE SELECT ONE CASE NUMBER :$,$)
05580         READ(5,)ISEL
05590         IF(ISEL.GE.1.A.ISEL.LE.6) GOTO 20
05600         WRITE(6,1009)
05610   1009 FORMAT(2X,$INCORRECT CASE NUMBER, PLEASE RE-ENTER $)
05620         GOTO 10
05630      20 CONTINUE
05640         RETURN
05650         END
05660C
05670C
05680C
05690         SUBROUTINE SETUP(SEL,IN,JN,FL,MQ)
05700C     OBJECTIVE:
05710C       TO GATHER THE FOLLOWING INPUT FROM THE USER:
05720C         IN = INITIATOR IN FORM 'XXX'
05730C         JN = RECIPIENT IN FORM 'YYY'
05740C         FL = ACTIVITY FLOW
05750C              0 = >>>
05760C              1 = <<<
05770C              2 = <->
05780C         MQ = TIME INCREMENT
05790C              1= MONTHLY
05800C              2 = QUARTERLY
05810C              3 = YEARLY
05820C (NOTE) CODE FOR DAILY AND 1/2 YRLY STILL INTACT
05830C
05840          IMPLICIT INTEGER (A-Z)
05850         COMMON /PARAM/ CI,MI,ADI(3),CR,MR,ADR(3),DUAL,
05860+                       SDATE,FDATE,TCODE,SMODE,R,ISPEED,INITAB
05870         COMMON /POINT/ BP,LSEL,ISEL
05880        CHARACTER$3 IN,JN,TM
05890         CHARACTER$3 CI,CR
05900         IF(LSEL.EQ.2) GOTO 15
05910         IF(SEL.EQ.4.0.SEL.EQ.5.0.SEL.EQ.6) GOTO 10
05920         WRITE(6,1001)
05930   1001 FORMAT(2X,$PLEASE SELECT TWO ACTORS :(E.G. XXX,YYY )$,$)
05940         READ(5,2001)IN,JN
05950   2001 FORMAT(A3,1X,A3)
05960      10 CONTINUE
05970         GOTO(13,13,13,11,12,14),SEL
05980      14 IN='USR'
05990         JN='CHN'
06000         GOTO 13
06010      11 IN='USA'
06020         JN='USR'
06030         GOTO 13
06040      12 IN='USA'
06050         JN='CHN'
06060      13 CONTINUE
06070         GOTO 16
06080      15 IN=CI
06090         JN=CR
06100      16 CONTINUE
06110         WRITE(6,1002)IN,JN,IN,JN,IN,JN
06120   1002 FORMAT(/,2X,$SPECIFY ACTIVITY FLOW : $,/,
06130+        6X,$0. ONE WAY ($,A3,$ >>> $,A3,$)$,/,
06140+        6X,$1. ONE WAY ($,A3,$ <<< $,A3,$)$,/,
06150+        6X,$2. TWO WAY ($,A3,$ <-> $,A3,$)$,$)
06160      20 CONTINUE
06170         READ(5,)FL
```

```
06180  2002 FORMAT(I1)
06190       IF(FL.GE.0.A.FL.LE.2) GOTO 30
06200       WRITE(6,1003)
06210  1003 FORMAT(2X,*INCORRECT FLOW NUMBER, PLEASE RE-ENTER *)
06220       GOTO 20
06230    30 IF(FL.NE.1) GOTO 31
06240       TN=IN
06250       IN=JN
06260       JN=TN
06270    31 CONTINUE
06280       IF(LSEL.NE.2) GOTO 32
06290       MQ=TCODE
06300       RETURN
06310    32 WRITE(6,1004)
06320  1004 FORMAT(/,2X,*SELECT TIME INCREMENT :*,/,
06330+      6X,*1. MONTHLY *,/,6X,*2. QUARTERLY *,/,6X,*3. YEARLY*,6X,/)
06340       READ(5,)MQ
06350       IF(MQ.GE.1.A.MQ.LE.3) GOTO 40
06360       WRITE(6,1005)
06370  1005 FORMAT(2X,*INCORRECT TIME INCREMENT, PLEASE RE-ENTER*)
06380       GOTO 30
06390    40 CONTINE
06400       RETURN
06410       END
06420C
06430C
06440C
06450       SUBROUTINE TSETUP(IDATE,JDATE)
06460C    OBJECTIVE:
06470C     TO READ IN PARAMETERS FM,FY,SM,SY, WHICH ARE THE 'FIRST MONTH'
06480C      'FIRST YEAR','SECOND MONTH','SECOND YEAR', RESPECTIVELY.
06490C     TO CHECK THESE MONTHS/YEARS FOR VALIDITY AND CONVERT THEM FROM
06500C     3-DIGIT ALPHA TWO DIGIT NUMERIC TO SIX DIGIT NUMERIC. I.E.
06510C          JUN 77 - MAY 78       TO    770601 - 780531
06520C
06530C    MAJOR PARAMETERS
06540C     IDATE - 'SDATE RETURNED IN FORM YYMMDD
06550C     JDATE - 'FDATE' RETURNED IN FORM YYMMDD
06560C
06570       INTEGER FM,FY,FD,SM,SY,SD,FMC,SMC
06580       DIMENSION MONTHS(12)
06590       DATA MONTHS /3HJAN,3HFEB,3HMAR,3HAPR,
06600+         3HMAY,3HJUN,3HJUL,3HAUG,
06610+         3HSEP,3HOCT,3HNOV,3HDEC/
06620       FD=01
06630       SD=31
06640    10 WRITE(6,1001)
06650  1001 FORMAT(/,2X,*SET TIME PARAMETERS (JAN66,DEC75) :*,/)
06660       READ(5,2001)FM,FY,SM,SY
06670  2001 FORMAT(A3,I2,1X,A3,I2)
06680       FMC=0
06690       DO 20 I=1,12
06700       IF(FM.EQ.MONTHS(I))FMC=I
06710    20 CONTINUE
06720       IF(FMC.NE.0) GOTO 30
06730       WRITE(6,1002)
06740  1002 FORMAT(2X,*FIRST MONTH INCORRECT, PLEASE RE-ENTER *)
06750       GOTO 10
06760    30 SMC=0
06770       DO 40 I=1,12
06780       IF(SM.EQ.MONTHS(I))SMC=I
06790    40 CONTINUE
06800       IF(SMC.NE.0) GOTO 50
06810       WRITE(6,1003)
06820  1003 FORMAT(2X,*SECOND MONTH INCORRECT, PLEASE RE-ENTER*)
06830       GOTO 10
06840    50 CONTINUE
06850       IF(FY.GE.66.A.FY.LE.75) GOTO 60
```

```
06860         WRITE(6,1004)
06870   1004  FORMAT(2X,*YEAR PARAMETER NOT WITHIN DATA BASE *,
06880+        *, PLEASE RE-ENTER *)
06890         GOTO 10
06900      60 CONTINUE
06910         IF(SY.GE.66.A.SY.LE.75) GOTO 70
06920         WRITE(6,1004)
06930         GOTO 10
06940      70 CONTINUE
06950         IDATE=(((FY*10000)+(FMC*100))+FD)
06960         JDATE=(((SY*10000)+(SMC*100))+SD)
06970         RETURN
06980         END
06990C
07000C
07010C
07020         SUBROUTINE SSBC
07030C
07040C    OBJECTIVE - PROVIDE HEADING IF ISEL = 1
07050C
07060         WRITE(6,1001)
07070   1001  FORMAT(2X,3(3H*),* SINO - SOVIET BORDER CLASHES , *,
07080+        *JAN , 1967 AND MARCH , 1969 *,3(3H*),/)
07090         WRITE(6,1002)
07100   1002  FORMAT(6X,*PRIMARY ACTORS : *,/,
07110+        10X,*PEOPLES REPUBLIC OF CHINA (CHN) *,/,
07120+        10X,*SOVIET UNION (USR) *,/,
07130+        6X,*OTHER ACTORS : *,/,
07140+        10X,*UNITED STATES (USA) *,/)
07150         RETURN
07160         END
07170C
07180C
07190C
07200         SUBROUTINE CZEINV
07210C
07220C    OBJECTIVE - PROVIDE HEADING IF ISEL = 2
07230C
07240         WRITE(6,1001)
07250   1001  FORMAT(2X,3(3H*),* CZECHOSLOVAKIAN INVASION , *,
07260+        *AUGUST , 1968 *,3(3H*),/)
07270         WRITE(6,1002)
07280   1002  FORMAT(6X,*PRIMARY ACTORS : *,/,
07290+        10X,*SOVIET UNION (USR) *,/,
07300+        10X,*CZECHOSLOVAKIA (CZE) *,/,
07310+        6X,*OTHER ACTORS :*,/,
07320+        10X,*UNITED STATES (USA)*,/,
07330+        10X,*PEOPLES REPUBLIC OF CHINA (CHN) *,/)
07340         RETURN
07350         END
07360C
07370C
07380C
07390         SUBROUTINE INDWAR
07400C
07410C    OBJECTIVE - PROVIDE HEADING IF ISEL = 3
07415C
07420         WRITE(6,1001)
07430   1001  FORMAT(2X,3(3H*),* INDO - PAKISTAN WAR ,*,
07440+        *NOVEMBER , 1971 *,3(3H*),/)
07450         WRITE(6,1002)
07460   1002  FORMAT(6X,*PRIMARY ACTORS :*,/,
07470+        10X,*INDIA (IND)*,/,
07480+        10X,*PAKISTAN (PAK)*,/,
07490+        6X,*OTHER ACTORS :*,/,
07500+        10X,*UNITED STATES (USA)*,/,
07510+        10X,*SOVIET UNION (USR)*,/,
07520+        10X,*PEOPLES REPUBLIC OF CHINA (CHN)*,/,
```

```
07530+       10X,*BANGLADESH (BGD)*,/)
07540        RETURN
07550        END
07560C
07570C
07580C
07590        SUBROUTINE USSOU
07600C
07610C    OBJECTIVE - PROVIDE HEADING IF ISEL = 4
07620C
07630        WRITE(6,1001)
07640    1001 FORMAT(2X,3(3H*),* U.S. - SOVIET UNION ,*,
07650+       *JAN , 1966 - DEC , 1975 *,3(3H*),/)
07660        WRITE(6,1002)
07670    1002 FORMAT(6X,*ACTORS : USA , USR *,/)
07680        RETURN
07690        END
07700C
07710C
07720C
07730        SUBROUTINE USPEO
07740C
07750C    OBJECTIVE - PROVIDE HEADING IF ISEL = 5
07760C
07770        WRITE(6,1001)
07780    1001 FORMAT(2X,3(3H*),* U.S. - PEOPLES REPUBLIC OF CHINA , *,
07790+       *JAN , 1966 - DEC , 1975 *,3(3H*),/)
07800        WRITE(6,1002)
07810    1002 FORMAT(6X,*ACTORS : USA , CHN *,/)
07820        RETURN
07830        END
07840C
07850C
07860C
07870        SUBROUTINE SOVPEO
07880C
07890C    OBJECTIVE - PROVIDE HEADING IF (SEL - 6
07900C
07910        WRITE(6,1001)
07920    1001 FORMAT(,3(3H*),* SOVIET UNION - PEOPLES REPUBLIC OF CHINA
07930+       *JAN , 1966 - DEC , 1975 *,3(3H*),/)
07940        WRITE(6,1002)
07950    1002 FORMAT(6X,*ACTORS : USR , CHN *,/)
07960        RETURN
07970        END
07980C
07990C
08000C
08010C
08020         OVERLAY(OVRLAY,2,0)
08030C
08040C    OBJECTIVE:
08050C     .TO READ THE DATA BASE AND EXTRACT ALL QUALIFIED DATA AND PLAC
08060C      INTO NEVENT ARRAY.
08070C      .TO DETERMINE NO. OF QUALIFIED EVENTS WITHIN TIME PERIOD, IN
08080C      INCREMENT AND TO COUNT THEM.
08090C     . TO CREATE ARRAYS AS FOLLOWS:
08100C           COOP - COOPERATIVE EVENTS
08110C           CONFL - CONFLICTUAL EVENTS
08120C           TOTAL - TOTAL COOPERATIVE + CONFLICTIVE
08130C           TF - TENSION LEVELS
08140C           HR - H-RELS
08150C     .TO CREATE A Z-SCORE ARRAY FOR EACH
08160C     .TO CREATE A PROBABILITY ARRAY FOR EACH
08170C     .TO PRODUCE A 30-DAY PROBABILITY TABLE IF REQUESTED IN OULY 1
08180C     .TO PRODUCE RDATE ARRAY OF READABLE DATES IN FORM MMM YY
08190C
08200C
```

```
08210C    SUBROUTINES--
08220C       ZSCORP - CALCULATE Z-SCORES
08230C       TENSION - CALCULATE TENSION
08240C       HREL - CALCULATE H-RELS
08250C       CONVRT - CONVERT INTEGER TO REAL ARRAYS
08260C       PTABLE - ASSIGN PROBABILITIES
08270C       NXTINC - NEXT TIME INCREMENT FUNCTION
08280C       SMLTAB - PRINT 30-DAY PROBABILITY TABLE
08290C       DEKODE - RETURN ALPHA YEAR + MONTH
08300C       DEKODA - RETURN ALPHA DATE ARRAY
08310C
08320C
08330         PROGRAM SEARCH
08340C
08350C    MAJOR PARAMETERS
08360C       MAX = MAXIMUM NUMBER OF PERIODS
08370C       MAXREC = LAST SECTOR OF DATA BASE
08380C       RUSD = FILENAME OF DATA BASE (LOGICAL UNIT 1)
08390C       JR = 1ST RECORD OF DATA BASE
08400C       R = RECORD CORRESPONDING TO SDATE IN DATA BASE
08410C       CDATE = CURRENT DATE AS DATA BASE IS BEING READ
08420C       NP = NUMBER OF POINTS AFTER CDATE = SDATE
08430C       BP = POINT WHEN NP=1
08440C       IC = TOTAL PERIODS COLLECTED
08450C       NDATE,NEVENT,NCONFL,NCOOP,NTOTAL ARE THE INITIAL
08460C           COLLECTION ARRAYS USED FOR CALCULATION OF PROGRESSSIVE
08470C           Z-SCORES
08480C       NCOUNT,COUNT = TOTAL + ACTUAL EVENT COUNTS
08490C
08500         IMPLICIT INTEGER(A-Z)
08510C
08520         COMMON /PARAM/ CI,NI,ADI(3),CR,NR,ADR(3),DUAL,
08530+                        SDATE,FDATE,TCODE,SMODE,R,ISPEED,INITAB
08540         COMMON /RESLT/ COUNT,IC,DATE(36),EVENT(22,36),
08550+                       COOP(36),CONFL(36),TOTAL(36),ATOTAL(36),
08560+                       ZTOTAL(36),ACOOP(36),ZCOOP(36),ACONFL(36),
08570+                       ZCONFL(36),TF(36),ZST(36),HR(36),ZSH(36),
08580+                       PTOTAL(36),PCOOP(36),PCONFL(36),RDATE(36)
08590         COMMON/POINT/BP,LSEL,ISEL
08600         DIMENSION NEVENT(22,120),NCOOP(120),NCONFL(120),NTOTAL(120)
08610C
08620         CHARACTER*3 CI,CR
08630         CHARACTER ADI,ADR
08640         LOGICAL DUAL
08650         REAL ZTOTAL,ZCOOP,ZCONFL
08660          REAL ATOTAL,ACOOP,ACONFL
08670         REAL ZST,ZSH,TF,HR
08680         REAL RTEMP(120),NTF(120),NHR(120),NDATE(120)
08690         REAL PTOTAL,PCOOP,PCONFL
08700C
08710         DATA NE/22/
08720C
08730         DATA MAX/36/
08740         DATA MAXREC/6552/
08750C
08760         CALL DFUR('GETR','RUSD','RUSD',0,0)
08770C
08780         CLOSE(1)
08790         OPEN(1,NAME='RUSD',READONLY=.TRUE.,TYPE='OLD')
08800C
08810         COUNT=0
08820         NCOUNT=0
08830         BP=0
08840         IC=0
08850         JR=1
08860         DATE(IC)=IDATE
08870         GO TO 201
08880C
```

```
08890    200 CONTINUE
08900        IF(JR.GE.R)R=R+1
08910        JR=JR+1
08920        IF(R.GT.MAXREC) GO TO 300
08930    201 CONTINUE
08940        READ(1,3001,REC=JR)CDATE,III,EEE,RRR,AAA
08950   3001 FORMAT(I6,4I3)
08960C
08970        IF(JR.LT.R)GOTO 202
08980        IF(CDATE.LT.SDATE) GO TO 200
08990        IF(CDATE.GT.FDATE) GO TO 300
09000C
09010    202 CONTINUE
09020        IF(.NOT.NXTINC(TCODE,IDATE,CDATE)) GO TO 210
09030        IF(IC.LT.MAX) GO TO 203
09040        GOTO 203
09050        WRITE(6,1012) MAX
09060   1012 FORMAT(' MAX NO OF INTERVALS (',I3,') EXCEEDED--',
09070+            'TABULATION INTERRUPTED.')
09080        GO TO 300
09090    203 CONTINUE
09100        IC=IC+1
09110        IF(CDATE.GE.SDATE)NP=NP+1
09120        IF(NP.EQ.1)BP=IC
09130        NDATE(IC)=IDATE
09140        DO 205 E=1,22
09150        NEVENT(E,IC)=0
09160    205 CONTINUE
09170    210 CONTINUE
09180C
09190        IF((NI.EQ.0.OR.III.EQ.NI).AND.(NR.EQ.0.OR.RRR.EQ.NR))
09200+            GO TO 220
09210        IF(DUAL.AND.
09220+           (NR.EQ.0.OR.III.EQ.NR).AND.(NI.EQ.0.OR.RRR.EQ.NI))
09230+             GO TO 220
09240        GO TO 200
09250C
09260    220 CONTINUE
09270        COUNT=COUNT+1
09280        IF(CDATE.GE.SDATE)NCOUNT=NCOUNT+1
09290        E=EEE/10
09300        NEVENT(E,IC)=NEVENT(E,IC)+1
09310        GO TO 200
09320C
09330    300 CONTINUE
09340        CLOSE(1,DISP='KEEP')
09350        WRITE(6,1011)NCOUNT
09360   1011 FORMAT(/,2X,'NUMBER OF EVENTS FOUND :  ',I5,/)
09370        READ 100,I
09380    100 FORMAT(A1)
09390        IF(COUNT.EQ.0) GO TO 320
09400C
09410        DO 310 I=1,IC
09420        NCOOP(I)=0
09430        DO 305 E=1,10
09440        NCOOP(I)=NCOOP(I)+NEVENT(E,I)
09450    305 CONTINUE
09460        NCONFL(I)=0
09470        DO 306 E=11,22
09480        NCONFL(I)=NCONFL(I)+NEVENT(E,I)
09490    306 CONTINUE
09500        NTOTAL(I)=NCOOP(I)+NCONFL(I)
09510    310 CONTINUE
09520C
09530    320 CONTINUE
09540C
09550C
09560C
```

```
09570         CALL CONVRT(IC,NTOTAL,RTEMP)
09580         CALL ZSCORP(IC,RTEMP,ZTOTAL)
09590         CALL PTABLE(ZTOTAL,NP,2,PTOTAL)
09600         CALL CONVRT(IC,NCOOP,RTEMP)
09610         CALL ZSCORP(IC,RTEMP,ZCOOP)
09620         CALL PTABLE(ZCOOP,NP,4,PCOOP)
09630         CALL CONVRT(IC,NCONFL,RTEMP)
09640         CALL ZSCORP(IC,RTEMP,ZCONFL)
09650         CALL PTABLE(ZCONFL,NP,6,PCONFL)
09660C
09670    313 CONTINUE
09680C
09690         CALL TENSON(IC,NCONFL,NTOTAL,NTF)
09700         CALL ZSCORP(IC,NTF,ZST)
09710         CALL HREL(IC,NE,NEVENT,NTOTAL,NHR)
09720         CALL ZSCORP(IC,NHR,ZSH)
09725C
09730         I=1
09740         DO 150 J=BP,IC
09750         TF(I)=NTF(J)
09760         HR(I)=NHR(J)
09770         DATE(I)=NDATE(J)
09780         I=I+1
09790    150 CONTINUE
09800         I=1
09810         DO 160 J=BP,IC
09820         COOP(I)=NCOOP(J)
09830         CONFL(I)=NCONFL(J)
09840         TOTAL(I)=NTOTAL(J)
09850         I=I+1
09860    160 CONTINUE
09870         I=1
09880         DO 170 J=BP,IC
09890          DO 180 E=1,22
09900          EVENT(E,I)=NEVENT(E,J)
09910     180 CONTINUE
09920          I=I+1
09930    170 CONTINUE
09940         IC=NP
09950         COUNT=NCOUNT
09960         IF(INITAB.EQ.1)CALL SMLTAB
09970         IF(INITAB.NE.1)CALL DEKODA(IC,DATE,RDATE)
09980         END
09990C
10000C
10010C
10020         SUBROUTINE ZSCORP(N,C,ZS)
10030C
10040C    OBJECTIVE:  TO CALCULATE PROGRESSIVE MEANS AND PROGRESSIVE
10050C    Z-SCORES FOR A COLLECTION OF ITEMS.  THE MEAN AND THE Z-SCORE
10060C    FOR THE ITH ITEM IS BASED UPON THE PRECEDING I-1 ITEMS ONLY.
10070C
10080C    N         - NO OF ITEMS IN COLLECTION (>=1).
10090C    C(N)      - COLLECTION OF ITEMS.
10100C    MEAN(N)   - MEANS (SET BY ZSCORP).
10110C    ZS(N)     - Z-SCORES (SET BY ZSCORP).
10120C        ZSR(120)  - TEMPORARY COLLECTION ARRAY
10130C
10140         IMPLICIT INTEGER(A-Z)
10150C
10160         COMMON/POINT/BP
10170         REAL C(N)
10180         REAL SUM,SQ,SD,DEV
10190         REAL SQRT
10200         REAL ZSR(120),ZS(36)
10210         REAL MEAN(120)
10220C
10230C
```

```
10240         SUM=0.
10250         SQ=0.
10260         MEAN(1)=0.
10270          ZSR(1)=0.
10280C
10290         IF(N.LE.1) GO TO 200
10300C
10310         DO 140 I=2,N
10320         K=I-1
10330         SUM=SUM+C(K)
10340         SQ=SQ+C(K)*C(K)
10350         MEAN(I)=SUM/K
10360         IF(K.GT.1) GO TO 100
10370         SD=0.
10380         GO TO 110
10390     100 CONTINUE
10400         SD=SQRT((K*SQ-SUM*SUM)/(K*(K-1)))
10410     110 CONTINUE
10420         IF(SD.NE.0.) GO TO 120
10430          ZSR(I)=0.
10440         GO TO 130
10450     120 CONTINUE
10460         ZSR(I)=(C(I)-MEAN(I))/SD
10470     130 CONTINUE
10480     140 CONTINUE
10490C
10500     200 CONTINUE
10510         I=1
10520         DO 150 J=BP,N
10530          ZS(I)=ZSR(J)
10540         I=I+1
10550     150 CONTINUE
10560         RETURN
10570C
10580         END
10590         SUBROUTINE TENSON(N,CONFL,TOTAL,TF)
10600C
10610C    OBJECTIVE:  TO CALCULATE THE TENSION FACTOR (PERCENT OF
10620C    CONFLICTIVE EVENTS) FOR EACH OF A COLLECTION OF TIME
10630C    INTERVALS.
10640C
10650C    N         - NO OF TIME INTERVALS (>=1).
10660C    CONFL(N)  - NO OF CONFLICTIVE EVENTS IN EACH TIME INTERVAL.
10670C    TATAL(N)  - TOTAL NO OF EVENTS IN EACH TIME INTERVAL.
10680C    TF(N)     - TENSION FACTOR FOR EACH TIME INTERVAL (SET BY
10690C                TENSON).
10700C
10710         IMPLICIT INTEGER(A-Z)
10720C
10730         INTEGER CONFL(N),TOTAL(N)
10740         REAL TF(N)
10750C
10760C
10770         DO 100 I=1,N
10780         TF(I)=0.
10790         IF(TOTAL(I).EQ.0.) GO TO 100
10800         TF(I)=100.*CONFL(I)/TOTAL(I)
10810     100 CONTINUE
10820C
10830         RETURN
10840C
10850         END
10852C
10853C
10854C
10860         SUBROUTINE HREL(N,NE,EVENT,TOTAL,HR)
10870C
10880C    OBJECTIVE:  TO CALCULATE H-REL OVER ALL EVENT CATEGORIES
```

```
10890C    FOR EACH OF A COLLECTION OF TIME INTERVALS.
10900C
10910C    N             - NO OF TIME INTERVALS (>=1).
10920C    NE            - NO OF EVENT CATEGORIES.
10930C    EVENT(NE,N)   - NO OF EVENTS IN EACH CATEGORY FOR EACH TIME
10940C                    INTERVAL.
10950C    TOTAL(N)      - TOTAL NO OF EVENTS IN EACH TIME INTERVAL.
10960C    HR(N)         - H-REL FOR EACH TIME INTERVAL (SET BY HREL).
10970C
10980         IMPLICIT INTEGER(A-Z)
10990C
11000         INTEGER EVENT(NE,N),TOTAL(N)
11010         REAL HR(N)
11020         REAL HMAX,HABS,R
11030         REAL ALOG,ALOG2
11040C
11050C    NOTE:  LOG2(X)=ALOG(X)/ALOG(2)
11060C
11070C
11080         ALOG2=ALOG(2.)
11090C
11100         HMAX=-ALOG(1./NE)/ALOG2
11110C
11120         DO 110 I=1,N
11130         HABS=0.
11140         DO 100 E=1,NE
11150         IF(EVENT(E,I).EQ.0) GO TO 100
11160         R=1.*EVENT(E,I)/TOTAL(I)
11170         HABS=HABS-R*ALOG(R)/ALOG2
11180    100  CONTINUE
11190         HR(I)=HABS/HMAX
11200    110  CONTINUE
11210C
11220         RETURN
11230C
11240         END
11245C
11246C
11247C
11250         SUBROUTINE CONVRT(N,IDATA,RDATA)
11260C
11270C    OBJECTIVE:  TO CONVERT AN ARRAY OF INTEGER VALUES INTO AN
11280C    ARRAY OF REAL VALUES.
11290C
11300C    N         - NO OF ITEMS.
11310C    IDATA(N)  - INTEGER ARRAY.
11320C    RDATA(N)  - REAL ARRAY (SET BY CONVRT).
11330C
11340         IMPLICIT INTEGER(A-Z)
11350C
11360         INTEGER IDATA(N)
11370         REAL RDATA(N)
11380C
11390C
11400         DO 100 I=1,N
11410         RDATA(I)=IDATA(I)
11420    100  CONTINUE
11430C
11440         RETURN
11450C
11460         END
11470C
11480C
11490C
11500          SUBROUTINE PTABLE(ZSCR,N,J,PROB)
11510C     OBJECTIVE : TO ASSIGN PROBABILITIES FROM Z-SCORES ACCORDING TO
11520C                 PROPER TABLE. TABLE 1 BEGINS AT PSCR(1). TABLE 2
11530C                 BEGINS AT PSCR(21). TABLE 3 BEGINS AT PSCR(41). EA
```

```
11540C                TABLE IS 20 ELEMENTS LONG. THE VALUE FOR 19 TO INF
11550C                ITY IS IN THE 19TH LOCATION AND THE VALUE FOR 0 IS
11560C                THE 20TH.
11570C
11580C     PARAMS:
11590C       ZSCR - Z-SCORE OF TOTAL,COOP, OR CONFL
11600C       N - NUMBER OF ELEMENTS IN ARRAY
11610C       J - TABLE TYPE
11620C           2 = TOTAL ACTIVITY
11630C           4 = COOPERATIVE ACTIVITY
11640C           6 = CONFLICTUAL ACTIVITY
11650C       PROB - PROBABILITY ARRAY RETURNED
11660C
11670C
11680          DIMENSION PROB(36)
11690          DIMENSION ZSCR(36),PSCR(60)
11700          DATA PSCR /.05,.1,.4,.5,.6,.6,.7,.7,.7,.7,
11710+          .8,.8,.8,.8,.8,.8,.8,.8,.9,.01,
11720+          .01,.01,.05,.1,.2,.2,.4,.4,.4,.4,
11730+          .6,.6,.6,.6,.6,.6,.6,.6,.7,.01,
11740+          .05,.1,.4,.5,.6,.6,.7,.7,.7,.7,
11750+          .8,.8,.8,.8,.8,.8,.8,.8,.9,.01/
11760C
11770C
11780          IF(J.EQ.2) K=0
11790          IF(J.EQ.4) K=20
11800          IF(J.EQ.6) K = 40
11810C
11820          DO 10 I=1,N
11830          IF(INT(ZSCR(I)+.5).GE.19) GOTO 20
11840          IF(INT(ZSCR(I)+.5).LE.0) GOTO 30
11850          IVAL = INT((ZSCR(I)+.5)+K)
11860          GOTO 40
11870       20 IVAL = 19 + K
11880          GOTO 40
11890       30 IVAL = 20 + K
11900       40 PROB(I) = PSCR(IVAL)
11910       10 CONTINUE
11920          RETURN
11930          END
11940C
11950C
11960C
11970          LOGICAL FUNCTION NXTINC(TCODE,IDATE,CDATE)
11980C
11990C     OBJECTIVE:  TO DETERMINE WHETHER OR NOT A GIVEN DATE SIGNALS
12000C     THE START OF THE NEXT TIME INCREMENT.  NXTINC IS SET .TRUE.
12010C     IF THE CURRENT DATE IS INTO THE NEXT TIME INCREMENT RELATIVE
12020C     TO AN INITIAL DATE; ELSE, NXTINC IS SET .FALSE.
12030C
12040C     TCODE - TIME INCREMENT CODE--
12050C             1:  MONTH
12060C             2:  QUARTER
12070C             3:  YEAR
12080C
12090C
12100C     IDATE - INITIAL DATE IN FORM YYYMMMDDD.  (SET TO CURRENT
12110C             DATE IF NXTINC IS SET TO .TRUE.; ELSE, UNCHANGED.)
12120C     CDATE - CURRENT DATE IN FORM YYMMDD.
12130C
12140          IMPLICIT INTEGER(A-Z)
12150C
12160C
12170          NXTINC=.FALSE.
12180C
12190           GO TO(200,300,500),TCODE
12200C
12210     100 CONTINUE
```

```
12220         IF(CDATE.LE.IDATE) GO TO 800
12230         GO TO 700
12240C
12250    200 CONTINUE
12260         DELTA=CDATE/100-IDATE/100
12270         IF(DELTA.LE.0) GO TO 800
12280         GO TO 700
12290C
12300    300 CONTINUE
12310         XDATE=CDATE
12320         DELTA=CDATE/10000-IDATE/10000
12330         IF(DELTA.LE.0) GO TO 310
12340         XDATE=CDATE-8800
12350    310 CONTINUE
12360         DELTA=XDATE/100-IDATE/100
12370         IF(DELTA.LE.2) GO TO 800
12380         GO TO 700
12390C
12400    400 CONTINUE
12410         XDATE=CDATE
12420         DELTA=CDATE/10000-IDATE/10000
12430         IF(DELTA.LE.0) GO TO 410
12440         XDATE=CDATE-8800
12450    410 CONTINUE
12460         DELTA=XDATE/100-IDATE/100
12470         IF(DELTA.LE.5) GO TO 800
12480         GO TO 700
12490C
12500    500 CONTINUE
12510         XDATE=CDATE
12520         DELTA=CDATE/10000-IDATE/10000
12530         IF(DELTA.LE.0) GO TO 510
12540         XDATE=CDATE-8800
12550    510 CONTINUE
12560         DELTA=XDATE/100-IDATE/100
12570         IF(DELTA.LE.11) GO TO 800
12580         GO TO 700
12590C
12600    700 CONTINUE
12610         NXTINC=.TRUE.
12620         IDATE=CDATE
12630C
12640    800 CONTINUE
12650         RETURN
12660C
12670         END
12680C
12690C
12700C
12710          SUBROUTINE SMLTAB
12720C
12730C    OBJECTIVE - TO PRINT OUT THE 30 DAY PROBABLISTIC TABLES USING
12740C                ARRAYS CONSTRUCTED AND PASSED THRU COMMON
12750C
12760C
12770C    MAJOR PARAMETERS:
12780C       DEKODE - SUBROUTINE TO GIVE ALPHA DATE
12790C       DEKODA - SUBROUTINE TO GIVE ALPHA DATE ARRAY
12800C       CI,CR - CHARACTER INITIATOR + RECIPEIENT (3-CHRS)
12810C       IDIR - FLOW DIRECTION FOR FORMATTING
12820C       BLANC - ROUTINE TO BLANC PAGE IN MAIN
12830C       INTRVL - TIME INCREMENT  ( 9 CHRS)
12840C       ISMONTH - START MONTH
12850C       ISYEAR - START YEAR
12860C       IFMONTH - FINISH MONTH
12870C       IFYEAR - FINISH YEAR
12880C       RDATE - ARRAY OF ALPHA DATES
12890C       PTOTAL - PROBABILITIES ON TOTAL EVENTS
```

```
12900C
12910        IMPLICIT INTEGER(A-Z)
12920         COMMON /PARAM/ CI,NI,ADI(3),CR,NR,ADR(3),DUAL,
12930+        SDATE,FDATE,TCODE,SMODE,R,ISPEED,INITAB
12940         COMMON /RESLT/ COUNT,IC,DATE(36),EVENT(22,36),
12950+        COOP(36),CONFL(36),TOTAL(36),ATOTAL(36),
12960+        ZTOTAL(36),ACOOP(36),ZCOOP(36),ACONFL(36),
12970+        ZCONFL(36),TF(36),ZST(36),HR(36),ZSH(36),
12980+        PTOTAL(36),PCOOP(36),PCONFL(36),RDATE(36)
12990         COMMON/POINT/BP
13000         CHARACTER*3 CI,CR
13010         CHARACTER ADI,ADR
13020        CHARACTER*9 INTRVL(3)
13030         LOGICAL DUAL
13040         REAL ZTOTAL,ZCOOP,ZCONFL
13050         REAL ZTOTAL,ACOOP,ACONFL
13060         REAL ZST,ZSH,TF,HR
13070         REAL PTOTAL,PCOOP,PCONFL
13080         DATA INTRVL /'  MONTHLY','QUARTERLY','    YEARLY'/
13090         IDIR=6H<<<>>>
13100         IF(.NOT.DUAL)IDIR=6H>>>>>>
13110         CALL BLANC
13120     1001 FORMAT(24X,3(3H*),1X,A3,2X,A6,2X,A3,1X,3(3H*),/)
13130         WRITE(6,1002)INTRVL(TCODE)
13140     1002 FORMAT(24X,A9,* PROBABILITY *)
13150         CALL DEKODE(SDATE,ISMONTH,ISYEAR)
13160         CALL DEKODE(FDATE,IFMONTH,IFYEAR)
13170         WRITE(6,1003)ISMONTH,ISYEAR,IFMONTH,IFYEAR
13180     1003 FORMAT(25X,A3,*, *,I4,* - *,A3,*, *,I4,/)
13190         WRITE(6,1001)CI,IDIR,CR
13200         WRITE(6,1004)
13210     1004 FORMAT(24X,*DATE*,9X,*PROBABILITY *,/)
13220         CALL DEKODA(IC,DATE,RDATE)
13230         NC=IC
13240         IF(IC.GT.18) 10,20
13250      10  NC=18
13260       20 DO 30 I=1,NC
13270         WRITE(6,1005)RDATE(I),PTOTAL(I)
13280     1005 FORMAT(24X,A6,10X,F6.2)
13290       30 CONTINUE
13300         READ 100,I
13310      100 FORMAT(A10)
13320         IF(IC.GT.18)11,21
13330       11 NF=NC+1
13340         CALL BLANC
13350         WRITE(6,1002)INTRVL(TCODE)
13360         WRITE(6,1003)ISMONTH,ISYEAR,IFMONTH,IFYEAR
13370         WRITE(6,1001)CI,IDIR,CR
13380         WRITE(6,1004)
13390         DO 31,I=NF,IC
13400         WRITE(6,1005)RDATE(I),PTOTAL(I)
13410       31 CONTINUE
13420         READ 100,I
13430       21 CONTINUE
13440         RETURN
13450         END
13460C
13470C
13480C
13490 SUBROUTINE DEKODE(ID,IMONTH,IYEAR)
13500C
13510C    OBJECTIVE - TO SPLIT I6 DATE INTO TWO PARTS MONTH + YEAR
13520C
13530C    PARAMETERS:
13540C       ID = I6 DATE
13550C       IMONTH = XXX (RETURNED)
13560C       IYEAR = 19XX (   "      )
13570C
```

```
13580 DIMENSION MONTHS(12)
13590 DATA MONTHS /3HJAN,3HFEB,3HMAR,3HAPR,3HMAY,3HJUN,
13600+              3HJUL,3HAUG,3HSEP,3HOCT,3HNOV,3HDEC/
13610 IY=ID/10000
13620 IM=(ID-10000*IY)/100
13630 IMONTH=MONTHS(IM)
13640 IYEAR=(IY+1900)
13650 RETURN
13660 END
13670C
13680C
13690C
13700 SUBROUTINE DEKODA(IC,IDATE,JDATE)
13710C
13720C    OBJECTIVE - TO BUILD AN ARRAY  FROM DATES IN I6 FORMAT TO
13730C                ALPHA FORMAT
13740C
13750C    PARAMETERS:
13760C       IC = NUMBER OF ELEMENTS IN ARRAY
13770C       IDATE = ARRAY THAT IS IC BIG, IN I6 FORMAT (YYMMDD)
13780C       JDATE = RETURNED ARRAY IN FORM (MMM YY)
13790C       ENCODE,DECODE - UCS SYSTEM UTILLITIES
13795C
13800 DIMENSION IDATE(IC),JDATE(IC),MONTHS(12)
13810 DATA MONTHS /3HJAN,3HFEB,3HMAR,3HAPR,3HMAY,3HJUN,
13820+    3HJUL,3HAUG,3HSEP,3HOCT,3HNOV,3HDEC/
13830 DO 10 I=1,IC
13840 IY=IDATE(I)/10000
13850 IM=(IDATE(I)-10000*IY)/100
13860 IMONTH=MONTHS(IM)
13870 ENCODE(6,1,ITEMP)IMONTH,IY
13880 DECODE(6,2,ITEMP)JDATE(I)
13890 2 FORMAT(A6)
13900 1 FORMAT(A3,1X,I2)
13910 10 CONTINUE
13920 RETURN
13930 END
13940C
13950C
13960C
13970C
13980       OVERLAY(OVRLAY,4,0)
13990C
14000C    OBJECTIVE:
14010C     TO PROVIDE GRAPHIC DISPLAY CAPABILITY TO THE PROGRAM.
14020C     THIS OVERLAY CALLS THE PLOT-10 PACKAGE USING BOTH THE
14030C     TCS AND AG-II ROUTINES.
14040C
14050C    SUBROUTINES--
14060C
14070C     PLOTIT - ALL PLOTS (BAR+GRAPH) ARE GENERATED BY A
14080C              CALL TO THIS SUBROUTINE
14090C     CHOOSE - ROUTINE DISPLAY MATRIX AND ASKS USER FOR
14100C              GRAPH TYPE, NUMBER, AND QUANTITY
14110C     USERS  - ROUTINE PLOTS PROBABILITIES ON EXISTING
14120C              GRAPH
14130C     PTABLE - ASSIGNS PROBABILITY VALUES FOR 'USERS' ROUTINE.
14140C              UNLIKE PTABLE IN OVERLAY 2 - PROBABILITIES
14150C              ARE ONLY ONE DECIMAL PLACE
14160C     CHCNVT - CONVERTS CHARACTER STRINGS TO ASCII ARRAYS
14170C     TITLE  - DISPLAYS TITLE OF GRAPH ON UPPER PORTION OF CRT.
14180C     LGND   - IF PLOT IS A MULTI-GRAPH ON ONE AXIS. THIS
14190C              ROUTINE WILL PLACE A LEGEND ABOVE GRAPH.
14200C
14210C    MAJOR PARAMETERS:
14215C
14220C     PLOUT  = 10X36 ARRAY CONTAINING THE 10 MAJOR OUTPUT
14230C              ARRAYS.  THEY ARE ORDERED 1-10 LIKE THE MATRIX
```

```
14240C     LABELS = 2X10 ARRAY CONTAINING LABELS FOR ABOVE DATA
14250C     START  = INDEX ARRAY USED TO CALCULATE 'LOC'
14260C     LOC    = INDEX SUBROUTINE PLOTIT.  GIVES INFORMATION
14270C              TO LOCATE POSITION OF GRAPH ON CRT.
14280C     FVCODE = 22 CATEGORIES OF EVENTS
14290C     IND    = ARRAY THAT CONTAINS VALUES OF PLOTS TO
14300C              BE DISPLAYED.
14310C     IANS   = VARIABLE  Y=YES, N=NO TO QUESTION 'ANOTHER
14320C                DISPLAY"
14330C     ADATE  = REAL DATE ARRAY
14340C      IGT     =  Y=YES(MULTI-GRAPH), N=NO
14350C      NUMB    =  FUNCTION TO REFILL IND FOR A BLOW-UP
14360C      IP      =  Y=YES(PROBABILITIES), N=NO
14370C
14380         PROGRAM PLOT
14390C
14400         IMPLICIT INTEGER(A-Z)
14410C
14420         COMMON /PARAM/ CI,NI,ADI(3),CR,NR,ADR(3),DUAL,
14430+                       SDATE,FDATE,TCODE,SMODE,R,ISPEED,INITAB
14440         COMMON /RESLT/ COUNT,IC,DATE(36),EVENT(22,36),
14450+                       COOP(36),CONFL(36),TOTAL(36),ATOTAL(36),
14460+                       ZTOTAL(36),ACOOP(36),ZCOOP(36),ACONFL(36),
14470+                       ZCONFL(36),TF(36),ZST(36),HR(36),ZSH(36),
14480+                       PTOTAL(36),PCOOP(36),PCONFL(36),RDATE(36)
14490          COMMON / CHOICE/ IYOLD,IT,IP,KNTR,PROB(36),IGT,IFLG
14500C
14510         CHARACTER*3 CI,CR
14520         CHARACTER ADI,ADR
14530         LOGICAL DUAL
14540         CHARACTER*4 FVCODE
14550         DIMENSION FVCODE(22)
14560         DIMENSION LABELS(2,10),PLOUT(36,10),START(6),LOC(13),IND(6)
14570C
14580C
14590         CHARACTER*9 INTRVL(3)
14600         CHARACTER*11 STYPE(2)
14610         REAL ZTOTAL,ZCOOP,ZCONFL,ZST,ZSH,TF,HR
14620         REAL PTOTAL,PCOOP,PCONFL
14630         REAL ADATE(36)
14640         REAL ATOTAL,ACOOP,ACONFL,PROB,PLOUT
14650C
14660         DATA NE/22/
14670          DATA INTRVL/'MONTH','QUARTER','YEAR'/
14680         DATA STYPE/'NORMAL','PROGRESSIVE'/
14690         DATA FVCODE/'YLD ','CMNT','CNSL','APPR','PRMS','GRNT',
14700+                    'REWD','AGRE','RQST','PROP','RJCT','ACUS',
14710+                    'PROT','DENY','DMND','WARN','THRT','DEMO',
14720+                    'RDUC','EXPL','SEIZ','FRCE'/
14730         DATA LABELS(1,1)/10HTOTAL ACTI/,LABELS(2,1)/10HUITY       /
14740         DATA LABELS(1,2)/10HTOT ACTIVI/,LABELS(2,2)/10HTY Z-SCORE/
14750         DATA LABELS(1,3)/10HCOOPERATIV/,LABELS(2,3)/10HE ACTIVITY/
14760         DATA LABELS(1,4)/10HCOO ACTIVI/,LABELS(2,4)/10HTY Z-SCORE/
14770         DATA LABELS(1,5)/10HCONFLICTUA/,LABELS(2,5)/10HL ACTIVITY/
14780         DATA LABELS(1,6)/10HCON ACTIVI/,LABELS(2,6)/10HTY Z-SCORE/
14790          DATA LABELS(1,7)/10H      TENS/,LABELS(2,7)/10HION        /
14800          DATA LABELS(1,8)/10H  TENSION /,LABELS(2,8)/10HZ-SCORE    /
14810          DATA LABELS(1,9)/10H    UNCERT/,LABELS(2,9)/10HAINTY       /
14820         DATA LABELS(1,10)/10HUNCERTAINT/,LABELS(2,10)/10HY Z-SCORE /
14830         DATA START/0,1,3,3,7,7/
14840         DATA LOC/13,11,12,7,8,9,10,1,2,3,4,5,6/
14850C
14860C
14870         DO 314 I=1,IC
14880         ADATE(I)=DATE(I)
14890         ATOTAL(I)=TOTAL(I)
14900         ACOOP(I)=COOP(I)
14910     314 ACONFL(I)=CONFL(I)
```

```
14920          DO 320 J=1,IC
14930          PLOUT(J,1)=ATOTAL(J)
14940          PLOUT(J,2)=ZTOTAL(J)
14950          PLOUT(J,3)=ACOOP(J)
14960          PLOUT(J,4)=ZCOOP(J)
14970          PLOUT(J,5)=ACONFL(J)
14980          PLOUT(J,6)=ZCONFL(J)
14990          PLOUT(J,7)=TF(J)
15000          PLOUT(J,8)=ZST(J)
15010          PLOUT(J,9)=HR(J)
15020          PLOUT(J,10)=ZSH(J)
15030      320 CONTINUE
15040C
15050C     THIS IS THE SECTION THAT CALLS THE PLOT SUBROUTINE
15060C        TWO SUBROUTINES ARE INVOLVED SO FAR:
15070C            CHOOSE (IANS,IND)
15080C                  WHERE
15090C              IANS = NUMBER OF PLOTS DESIRED
15100C              IND  = THE INDIVIDUAL PLOT NUMBERS
15110C
15120C            PLOTIT(ILOC,IC,ADATE,PLOUT,LABELS)
15130C
15140C
15150       20 CALL INITT(ISPEED)
15160          CALL CHOOSE(N,IND)
15170           L=N
15180       25 CALL INITT(ISPEED)
15190           IF(IGT.EQ.1HY)IFLG=0
15200          DO 10 I=1,N
15210          ILOC=LOC(I+START(N))
15220           IF(IGT.EQ.1HY) ILOC=LOC(START(2))
15230           K=IND(I)
15240           IF(IP.NE.1HY) GOTO 12
15250           GOTO(31,31,32,32,33,33,12,12,12,12),K
15260       31 J = 2
15270           GO TO 11
15280       32 J = 4
15290           GO TO 11
15300       33 J = 6
15310       11 CALL PTABLE(PLOUT(1,J),IC,J)
15320       12 CALL PLOTIT(ILOC,IC,ADATE(1),PLOUT(1,K),LABELS(1,K),IND)
15330       10 CONTINUE
15340           CALL TITLE
15350           IF(IGT.EQ.1HY) CALL LGND(L,IND)
15360           CALL FINITT(0,60)
15370           IGT=1HN
15380          READ 100,I
15390      100 FORMAT(A10)
15400           IND(1)=NUMB(I)
15410           IF(IND(1).EQ.0)GOTO 40
15420           N=1
15430           GOTO 25
15440       40 PRINT,* ANOTHER DISPLAY (Y OR N)*,
15450          READ,IANS
15460          IF(IANS.EQ.1HY) GOTO 20
15470         RETURN
15480          END
15490C
15500C
15510C
15520           SUBROUTINE PLOTIT(INDEX,NP,XDATA,YDATA,TEXT,INUM)
15530C
15540C     OBJECTIVE:
15550C      THIS ROUTINE IS THE SINGLE MOST IMPORTANT PART OF
15560C      THE PROGRAM.  ALL PLOTS ARE GENERATED FROM HERE.
15570C      MOST OF THE CALLS ARE TO PLOT-10(AG-II)
15580C
15590C     MAJOR PARAMETERS:
```

```
15600C     INDEX = POSITION OF PLOT ON CRT 1-13
15610C     NP    = NUMBER OF X POINTS ON EACH PLOT
15620C     XDATA = X POINTS
15630C     YDATA = Y POINTS
15640C     TEXT  = TITLE TEXT (FROM LABELS)
15650C     INUM  = NUMBER OF GRAPHS BEING PLOTTED ON ONE PAGE
15660C
15670C
15680C
15690C
15700C
15710C
15720          COMMON /PARAM/ CI,NI,ADI(3),CR,NR,ADR(3),DUAL,
15730+                        SDATE,FDATE,TCODE,SMODE,R,ISPEED,INITAB
15740          COMMON / CHOICE/ IYOLD,IT,IP,KNTR,PROB(36),IGT,IFLG
15750C
15760C
15770          INTEGER INUM(6)
15780          INTEGER IASCII(20),TEXT(2),ILINE(2)
15790          INTEGER TCODE,PERIOD(3),PERWDTH(3)
15800          INTEGER POS(13),XTIT(13),YTIT(13)
15810          REAL PROB
15820          REAL X(13),Y(37),YDATA(NP)
15830          INTEGER WID(13)
15840C
15850C
15860          DATA ILINE /5454,1212/
15870          DATA PERIOD/12,4,1/
15880          DATA PERWDTH/3,1,4/
15890          DATA(X(I),I=1,5)/-2.,0.,12.,1966.,0./
15900          DATA POS/3HUL6,3HLL6,3HUC6,3HLC6,3HUR6,3HLR6,
15910+          3HUL4,3HLL4,3HUR4,3HLR4,3HUPH,3HLOH,3HSTD/
15920          DATA XTIT/238,238,563,563,888,888,300,300,
15930+          800,800,500,500,500/
15940          DATA YTIT/725.350,725,350,725,350,725,350,
15950+          725,350,725,350,725/
15960         DATA WID/15,15,15,15,15,15,20,20,20,20,40,40,40/
15970C
15980C
15990          X(2)=FLOAT(NP)
16000           X(3)=FLOAT(PERIOD(TCODE))
16010           X(4)=FLOAT(INT((XDATA/10000)+1900))
16020           X(5)=FLOAT(INT((XDATA-((X(4)-1900)*10000))/100))
16030          Y(1)=FLOAT(NP)
16035C
16036C
16040C     MAJOR VARIABLES AND DATA VARIABLES:
16050C      IASCII   = RETURNED ASCII CHRS FROM CHCNVT ROUTINE
16060C      ILINE    = ALTERNATE LINES USED FOR 2 AND 3 PLOTS
16070C                 PER GRAPH, ILINE(1)=DASHES, ILINE(2)=DOTS
16080C      XTIT     = X POSITION OF TITLE DEPENDING ON POS
16090C      YTIT     = Y POSITION OF TITLE DEPENDING ON POS
16100C      WID      = INDEXED VALUES FOR WIDTH OF BAR-CHARTS
16110C      JLAB     = INDEXED VALUES OF X LABEL
16120C      IFLG     = FLAG DENOTING ITERATION ON MULTI-GRAPH
16130C      TCODE    = TIME INCREMENT
16140C      KNTR     = FLAG USED FOR PROBABILITY IF = 1 PLOT PROBS
16150C
16160C
16170C
16180C
16190          IWIDTH=WID(INDEX)/INT((NP+11.)/12.)
16200          DO 10 I=1,NP
16210      10 Y(1+I) = YDATA(I)
16220          CALL CHCNVT(20,TEXT,IASCII)
16230          JLAB=TCODE+5
16240          CALL BINITT
16250          IF(IT.EQ.1HB) CALL VBARST(0,IWIDTH,0)
```

```
16260          IF(IT.EQ.1HS) CALL VBARST(8,IWIDTH,IWIDTH)
16270          IF(IGT.EQ.1HY.A.INUM(1).EQ.7) GOTO 11
16280          CALL PLACE(POS(INDEX))
16290         GOTO 12
16300        11 CALL SLIMX(150,900)
16310        12 CONTINUE
16320          CALL YLOC(0)
16330          CALL XLAB(1)
16340           CALL XWDTH(PERWDTH(TCODE))
16350           CALL YTYPE(1)
16360           IF(IFLG.GT.0.A.IGT.EQ.1HY.A.INUM(1).EQ.7)GOTO 23
16370           IF(IFLG.GT.0.A.IGT.EQ.1HY)20,21
16380        20 CONTINUE
16390          CALL XFRM(2)
16400          CALL XTYPE(JLAB)
16410          CALL XDEN(5)
16420          CALL CHECK(X,Y)
16430          CALL LLINE(ILINE(IFLG))
16440           CALL CPLOT(X,Y)
16450           GOTO 22
16460         21 CONTINUE
16470          CALL XFRM(2)
16480          CALL XTYPE(JLAB)
16490          CALL XDEN(5)
16500          CALL CHECK(X,Y)
16510           CALL DSPLAY(X,Y)
16520        22 CONTINUE
16530          IF(IGT.EQ.1HY)IFLG=IFLG+1
16540           IF(IP.NE.1HY) GOTO 30
16550           IYOLD=0
16560           KNTR = 1
16570           CALL LLINE(8888)
16580           CALL SYMBL(-1)
16590          CALL CPLOT(X,Y)
16600          GOTO 30
16610        23 CONTINUE
16620          CALL YFRM(2)
16630             CALL YLOCRT(0)
16640          CALL XLAB(0)
16650          CALL XFRM(0)
16660          CALL DLIMY(0.,0.)
16670          CALL CHECK(X,Y)
16680          CALL LLINE(ILINE(IFLG))
16690          CALL DSPLAY(X,Y)
16700         30 IF(IFLG.EQ.1.O.INDEX.LT.13) CALL FRAME
16710          IF(IGT.EQ.1HN.A.INDEX.EQ.13) CALL FRAME
16720          CALL JUSTER(20,IASCII,0,32,LEN,IOFF)
16730          IF(IGT.EQ.1HY)GOTO 31
16740          CALL MOTATE(XTIT(INDEX)+IOFF,YTIT(INDEX),LEN,IASCII)
16750         31 CONTINUE
16760          RETURN
16770          END
16780C
16790C
16800C
16810          SUBROUTINE CHOOSE(IANS,INDU)
16820C
16830C
16840          COMMON /PARAM/ CI,NI,ADI(3),CR,NR,ADR(3),DUAL,
16850+                        SDATE,FDATE,TCODE,SMODE,R,ISPEED,INITAB
16860          COMMON / CHOICE/ IYOLD,IT,IP,KNTR,PROB(36),IGT,IFLG
16870         DIMENSION INDU(6)
16880         DIMENSION Z(7),RD(8),A(8),TOT(5),C(4),CON(8),T(7),U(6),L(5),
16890         INTEGER Z,RD,TOT,A,C,CON,T,U,L,N
16900        INTEGER TIT(20)
16910          INTEGER SDATE,FDATE,TCODE,SMODE
16920          REAL PROB
16930C
```

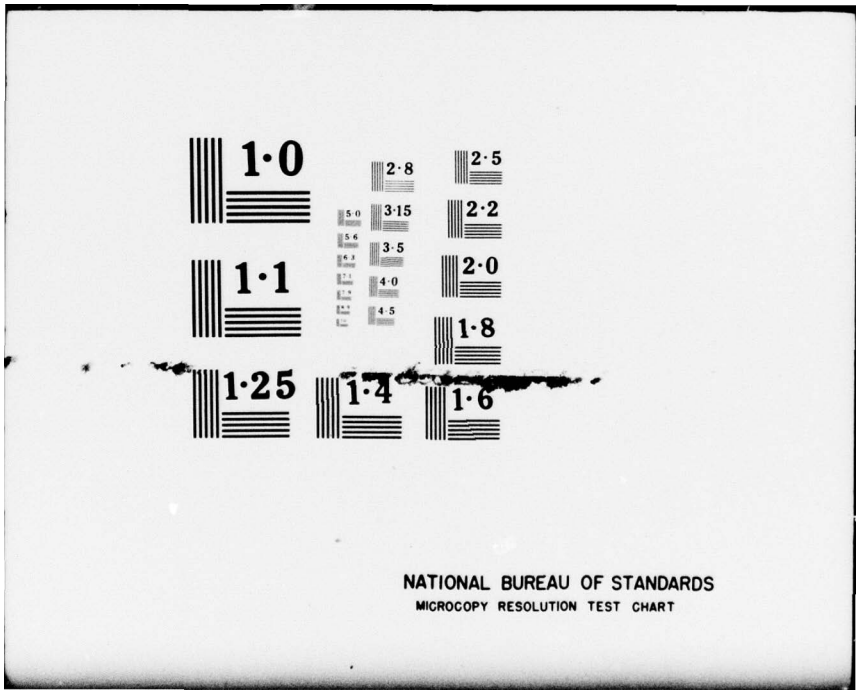
```
16940C     LITERAL DATA FOR LABELS
16950C
16960        DATA Z/90,45,83,67,79,82,69/
16970        DATA RD/82,65,87,32,68,65,84,65/
16980        DATA TOT/84,79,84,65,76/
16990        DATA A/65,67,84,73,86,73,84,89/
17000        DATA C/67,79,79,80/
17010        DATA CON/67,79,78,70,76,73,67,84/
17020        DATA T/84,69,78,83,73,79,78/
17030        DATA U/85,78,67,69,82,84/
17040        DATA L/76,69,86,69,76/
17050        DATA N/49,50,51,52,53,54,55,56,57,49,48/
17060       DATA TIT/80,79,76,73,84,73,67,65,76,32,
17070+              73,78,68,73,67,65,84,79,82,83/
17080C
17090C     OBJECTIVE - TO PROVIDE A SUBROUTINE THAT WILL DRAW A MATRIX.
17100C       ON THE GRAPH THE USER WILL BE GIVEN A CHOICE OF HOW MANY
17110C      GRAPHS HE WANTS TO SEE AND WHICH ONES...
17120C
17130C     THE SUBROUTINE WILL PASS BACK TO OVERLAY 4, THE VALUES
17140C
17150C
17160C           N = THE NUMBER OF GRAPHS DESIRED
17170C               THIS NUMBER RANGES FROM 1-6.
17180C        INDV = THE NUMBERS OF THE GRAPHS THEMSELVES
17190C               EQUALS THE VALUE OF N..
17195C
17200C     PLOT-19 (TEKTRONICS) SOFTWARE IS BEING USER IN THIS ROUTINE
17210C       THE THREE MAIN CALLS USED ARE:
17220C              MOVABS- MOVE TO ABSOULTE SCREEN COORD (X,Y)
17230C                DRWABS- DRAW A LINE FROM CURRENT COORD. TO (X,Y)
17240C              NOTATE- AT COORD(X,Y) PRINT N-CHARS, OF IARRAY
17250C
17260C     MAJOR PARAMETERS:
17270C      IGT    = SET TO 'Y' IF USER WANTS MULTI-GRAPH
17280C      IT     = SET TO 'G'=GRAPHS, 'B'=BAR CHARTS, 'S'=SHADED BARS
17290C      IANS   = NUMBER OF CHARTS/GRAPHS DESIRED.  THESE
17300C               WILL BE PLOTTED EACH WITH THEIR OWN AXIS.
17310C      INDV   = THE GRAPH NUMBERS AS SHOWN IN THE MATRIX
17320C               THAT THE USER WANTS PLOTTED (FROM ONE TO SIX).
17330C               THIS VARIABLE IS USED WHEN IGT IS 'Y' OR 'N'.
17340C      IP     = IF 'Y' PROBABILITIES ARE PRINTED
17350C      IREP   = MULTI-GRAPH SELECTION (1, 2, 3, OR 4)
17360C
17370C
17380C     INITIALIZE PLOT-10
17390C
17400C
17410C     MOVE TO LOWER LEFT CORNER OF RECTANGLE
17420        CALL MOVABS(200,300)
17430C     DRAW BOTTOM LINE
17440        CALL DRWABS(950,300)
17450C     DRAW RIGHT SIDE
17460        CALL DRWABS(950,600)
17470C     DRAW TOP
17480C
17490        CALL DRWABS(200,600)
17500C     DRAW LEFT SIDE
17510        CALL DRWABS(200,300)
17520C     MOVE TO THE LEFT SIDE AND DRAW A HORIZ. LINE THRU RECTANGLE
17530        CALL MOVABS(70,450)
17540        CALL DRWABS(950,450)
17550C
17560C     NOW MOVE TO THE TOP AND WITH 4 OTHER MOVES AND DRAWS
17570C     DRAW 4 VERT. LINES THRU THE RECTANGLE..THIS WILL CREATE 10
17580C     LITTLE BOXES. THAT ARE CLOSED..
17590C
17600        CALL MOVABS(350,700)
```

```
17610         CALL DRWABS(350,300)
17620         CALL MOVABS(500,700)
17630         CALL DRWABS(500,300)
17640         CALL MOVABS(650,700)
17650         CALL DRWABS(650,300)
17660         CALL MOVABS(800,700)
17670         CALL DRWABS(800,300)
17680         CALL MOVABS(950,600)
17690         CALL DRWABS(950,700)
17700         CALL DRWABS(70,700)
17710         CALL DRWABS(70,300)
17720         CALL DRWABS(200,300)
17730         CALL MOVABS(70,700)
17740         CALL DRWABS(200,600)
17750C
17760C     NOW LABEL ALL THE BOXES
17770C
17780         CALL NOTATE(85,375,7,Z)
17790         CALL NOTATE(85,525,8,RD)
17800         CALL NOTATE(210,650,5,TOT)
17810         CALL NOTATE(210,610,8,A)
17820         CALL NOTATE(360,650,4,C)
17830         CALL NOTATE(360,610,8,A)
17840         CALL NOTATE(510,650,8,CON)
17850         CALL NOTATE(510,610,8,A)
17860         CALL NOTATE(660,650,7,T)
17870         CALL NOTATE(660,610,5,L)
17880         CALL NOTATE(810,650,6,U)
17890         CALL NOTATE(810,610,5,L)
17900C
17910C
17920C     PUT THE NUMBERS 1 THRU 10 IN THE APPROPRIATE BOXES
17930C     ALSO,RING THE BELL JUST TO GIVE THIS CODE SOME
17940C     REAL CLASS
17950C
17960         CALL NOTATE(275,525,1,N(1))
17970         CALL BELL
17980         CALL NOTATE(275,375,1,N(2))
17990         CALL BELL
18000         CALL NOTATE(425,525,1,N(3))
18010         CALL BELL
18020         CALL NOTATE(425,375,1,N(4))
18030         CALL BELL
18040         CALL NOTATE(575,525,1,N(5))
18050         CALL BELL
18060         CALL NOTATE(575,375,1,N(6))
18070         CALL BELL
18080         CALL NOTATE(725,525,1,N(7))
18090         CALL BELL
18100         CALL NOTATE(725,375,1,N(8))
18110         CALL BELL
18120         CALL NOTATE(875,525,1,N(9))
18130         CALL BELL
18140         CALL NOTATE(875,375,2,N(10))
18150         CALL BELL
18160       CALL NOTATE(400,740,20,TIT)
18170C
18180C     CALL FINITT BOTH EXITS AND POSITIONS THE CURSER BELOW THE BOX
18190         CALL FINITT(0,275)
18200         PRINT,*     DO YOU WANT MORE THAN ONE PLOT PER GRAPH (Y OR N
18210         READ,IGT
18220         IF(IGT.EQ.1HY)GOTO 150
18230      10 PRINT,*     DO YOU WANT GRAPHS OR BAR-CHARTS (G OR B) *,
18240         READ,IT
18250         IPRT=10HGRAPHS
18260         IF(IT.EQ.1HB.O.IT.EQ.1HS)IPRT=10HCHARTS
18270         PRINT 101,IPRT
18280   101   FORMAT(4X,*HOW MANY *,A6,* DO YOU WANT (1-6) *,?)
```

AD-A039 663 DECISIONS AND DESIGNS INC MCLEAN VA F/G 5/2
SOFTWARE DESIGN FOR AN INTERACTIVE CRISIS EARLY WARNING PROTOTY--ETC(U)
DEC 76 J F WITTMEYER N00014-76-C-0712
UNCLASSIFIED DT/TR76-20 NL

2 OF 2
AD
A039663

END
DATE
FILMED
6-77


1·0
1·1
1·25
1·4
1·6
1·8
2·0
2·2
2·5
2·8
3·15
3·5
4·0
4·5
5·0
5·6
6·3
NATIONAL BUREAU OF STANDARDS
MICROCOPY RESOLUTION TEST CHART

```
18290         READ,IANS
18300         IF(IANS.LE.0.O.IANS.GT.6)20,30
18310      20 PRINT,*   MINIMUM/MAXIMUM GRAPH COUNT INCORRECT, PLEASE RE-T
18320         GOTO 10
18330      30 PRINT,* *
18340           INO=10H NUMBERS
18350        IF(IANS.EQ.1)INO=10H NUMBER
18360           IF(IANS.EQ.1)INDU(2)=0
18370           PRINT 100,IANS,IPRT,INO
18380     100   FORMAT(4X,*PLEASE TYPE IN THE*,I2,1X,A5,A8,* YOU *,/,
18390+       *     WOULD LIKE DISPLAYED IN THE FORM (1,2, ... 6) *,↑)
18400           IP=1HN
18410         READ,(INDU(I),I=1,IANS)
18420         IF(IANS.GT.2)GOTO 40
18430         IF(INDU(1).GT.6.0.INDU(2).GT.6) GOTO 40
18440           IF(TCODE.NE.1)GOTO 40
18450         PRINT,/,*     DO YOU WANT PROBABILITIES (Y OR N) *,
18460         READ,IP
18470       40 CONTINUE
18480         GOTO 800
18490      150 PRINT 155
18500      155 FORMAT(4X,*YOU HAVE THESE CHOICES: *,/,10X,*1 = GRAPHS 1,3
18510+       /,10X,*2 = GRAPHS 2,4 + 6 *,
18520+         /,10X,*3 = GRAPHS 7 + 9 *,
18530+         /,10X,*4 = GRAPHS 8 + 10 *,/,4X,*PLEASE PICK ONE *,↑)
18540     203 READ,IREP
18550    1001 FORMAT(4X,*INCORRECT RESPONSE, PLEASE RE-ENTER *,↑)
18560         IF(IREP.LE.4.A.IREP.GE.1)GOTO 202
18570         WRITE(6,1001)
18580         GOTO 203
18590     202 CONTINUE
18600           IT=1HG
18610           J=0
18620         GOTO(200,400,600,700),IREP
18630      200 IANS=3
18640          IP=1HN
18650          DO 201 I=1,5,2
18660          J=J+1
18670          INDU(J)=I
18680      201 CONTINUE
18690          GOTO 800
18700      300  IANS=3
18710          IP=1HY
18720          DO 301 I=1,5,2
18730          J=J+1
18740          INDU(J)=I
18750      301 CONTINUE
18760          GOTO 800
18770      400 IANS=3
18780          IP=1HN
18790          DO 401 I=2,6,2
18800          J=J+1
18810          INDU(J)=I
18820      401 CONTINUE
18830          GOTO 800
18840      500 IANS=3
18850          IP=1HY
18860          DO 501 I=2,6,2
18870          J=J+1
18880          INDU(J)=I
18890      501 CONTINUE
18900          GOTO 800
18910      600 IANS=2
18920          IP=1HN
18930          DO 601 I=7,9,2
18940          J=J+1
18950          INDU(J)=I
18960      601 CONTINUE
```

```
18970          GOTO 800
18980      700 IANS=2
18990          IP=1HN
19000          DO 701 I=8,10,2
19010          J=J+1
19020          INDV(J)=I
19030      701 CONTINUE
19040      800 CONTINUE
19050          RETURN
19060          END
19070C
19080C
19090C
19100        FUNCTION NUMB(I)
19110C
19120C     OBJECTIVES:
19130C      THIS FUNCTION WILL RETURN AN INTEGER FROM ONE TO TEN.
19140C      THIS IS DONE WHEN THE USER DECIDES TO BLOW-UP ONE GRAPH
19150C      AFTER HAVING PLOTTED ANYTHING PRIOR.  THE USER HAS THE
19160C      OPTION OF TYPING EITHER THE NUMBER OF THE PLOT OR AN
19170C      ALPHA CODE REPRESENTING IT.  THE POSSIBLE VALID ANSWERS
19180C      ARE:
19190C            1  OR  TOT              6  OR  CONZ
19200C            2  OR  TOTZ             7  OR  TEN
19210C            3  OR  COO              8  OR  TENZ
19220C            4  OR  COOZ             9  OR  UNC
19230C            5  OR  CON             10  OR  UNCZ
19240C
19250        DIMENSION IGOOD(10),ICH(10)
19260        DATA IGOOD/2H1 ,2H2 ,2H3 ,2H4 ,2H5 ,2H6 ,2H7 ,
19270+        2H8 ,2H9 ,2H10/
19280         DATA ICH/3HTOT,4HTOTZ,3HCOO,4HCOOZ,3HCON,4HCONZ,
19290+               3HTEN,4HTENZ,3HUNC,4HUNCZ/
19300        NUMB=0
19310        DO 10 K=1,10
19320        IF(I.EQ.IGOOD(K))NUMB=K
19330        IF(I.EQ.ICH(K))NUMB=K
19340     10 CONTINUE
19350        RETURN
19360         END
19370C
19380C
19390C
19400        SUBROUTINE USERS(X,Y,ISYM)
19410C
19420C
19430C     THIS ROUTINE WAS STOLEN FROM PAGE 191 OF THE AG II MANUAL
19440C
19450C      WILL LABEL POINTS WITH PROBABILITIES
19460C
19470C
19480          COMMON / CHOICE/ IYOLD,IT,IP,KNTR,PROB(36),IGT,IFLG
19490          REAL PROB
19500         INTEGER IA(15)
19510          DATA ISIZ/10/
19520C     CONVERT TO SCREEN COORDINATES
19530          CALL WINCOT(X,Y,IX,IY)
19540C     THIS CODE IS A FIRST ENTRY BRANCH
19550C     STORE SLOPE OF LINE -DOWN OR UP- FOR LABEL LOC
19560          ISIG=IY-IYOLD
19570C     BRANCH IF POINT IS OUTSIDE OF WO=INDOW
19580          IF(GENFLG(1))GOTO 100
19590C     GET PARAMETERS FROM AXIS COMMON TO DEFINE LENGTH OF LABEL
19600          NBASE=IBASEY(0)
19610          NDEC=1
19620           IWIDTH=2
19630           P = PROB(KNTR)
19640C     CONVERT Y VALUE TO ALPHA STRING
```

```
19650         CALL FFORM(P,IWIDTH,NDEC,IA,32)
19660C      RIGHT JUSTIFY LABEL
19670         CALL JUSTER(IWIDTH,IA,1,32,LEN,IOFF)
19680          IF(IT.EQ.1HB.O.IT.EQ.1HS) GOTO 99
19690C      DRAW SMALL ARROW ABOVE OR BELOW LINE
19700         CALL MOVABS(IX-10,IY+ISIGN(2,ISIG))
19710         CALL DRWABS(IX,IY)
19720         CALL DRWABS(IX-2,IY+ISIGN(10,ISIG))
19730         CALL MOVABS(IX,IY)
19740         CALL DRWABS(IX-ISIZ,IY+ISIGN(ISIZ,ISIG))
19750         CALL DRWABS(IX-ISIZ-5,IY+ISIGN(ISIZ,ISIG))
19760C      WRITE OUT LABEL
19770         IYPOS=IY+ISIGN(ISIZ,ISIG)-8
19780         CALL NOTATE(IX-ISIZ-7+IOFF,IYPOS,LEN,IA(IWIDTH-LEN+1))
19790C      PREPARE FOR NEXT SLOPE CALCULATION
19800         GOTO 100
19810      99 CALL NOTATE(IX-10,IY+5,LEN,IA(IWIDTH-LEN+1))
19820     100 IYOLD=IY
19830          KNTR = KNTR + 1
19840         RETURN
19850         END
19860C
19870C
19880C
19890          SUBROUTINE PTABLE(ZSCR,N,J)
19900C
19910C      OBJECTIVE : TO ASSIGN PROBABILITIES FROM Z-SCORES ACCORDING TO
19920C                  PROPER TABLE. TABLE 1 BEGINS AT PSCR(1). TABLE 2
19930C                  BEGINS AT PSCR(21). TABLE 3 BEGINS AT PSCR(41). EA
19940C                  TABLE IS 20 ELEMENTS LONG. THE VALUE FOR 19 TO INF
19950C                  ITY IS IN THE 19TH LOCATION AND THE VALUE FOR 0 IS
19960C                  THE 20TH.
19970C
19980C      PARAMS:
19990C        ZSCR - Z-SCORE OF TOTAL,COOP, OR CONFL
20000C        N - NUMBER OF ELEMENTS IN ARRAY
20010C        J - TABLE TYPE
20020C            2 = TOTAL ACTIVITY
20030C            4 = COOPERATIVE ACTIVITY
20040C            6 = CONFLICTUAL ACTIVITY
20050C        PROB - PROBABILITY ARRAY RETURNED
20060C
20070C
20080          COMMON / CHOICE/ IYOLD,IT,IP,KNTR,PROB(36),IGT,IFLG
20090          DIMENSION ZSCR(36),PSCR(60)
20100          DATA PSCR /.1,.1,.4,.5,.6,.6,.7,.7,.7,.7,
20110+          .8,.8,.8,.8,.8,.8,.8,.8,.9,.1,
20120+          .1,.1,.1,.1,.2,.2,.4,.4,.4,.4,
20130+          .6,.6,.6,.6,.6,.6,.6,.6,.7,.1,
20140+          .1,.1,.4,.5,.6,.6,.7,.7,.7,.7,
20150+          .8,.8,.8,.8,.8,.8,.8,.8,.9,.1/
20160C
20170C
20180          IF(J.EQ.2) K=0
20190          IF(J.EQ.4) K=20
20200          IF(J.EQ.6) K = 40
20210C
20220          DO 10 I=1,N
20230          IF(INT(ZSCR(I)+.5).GE.19) GOTO 20
20240          IF(INT(ZSCR(I)+.5).LE.0) GOTO 30
20250          IVAL = INT((ZSCR(I)+.5)+K)
20260          GOTO 40
20270       20 IVAL = 19 + K
20280          GOTO 40
20290       30 IVAL = 20 + K
20300       40 PROB(I) = PSCR(IVAL)
20310       10 CONTINUE
20320          RETURN
```

```
20330         END
20340C
20350C
20360C
20370         SUBROUTINE CHCNVT(N,TEXT,IASCII)
20380C
20390C     OBJECT: THIS ROUTINE WILL TAKE A TEXT CHARACTER STRING OF
20400C     OF N LENGTH AND CONVERT IT TO USASCII CODE FUNCTIONS FOR
20410C     PLOT-10 (AG-II) PLOTTER OUTPUT...
20420C
20430C      PARAMS:
20440C     N-NUMBER OF CHARS TO CONVERT
20450C     TEXT- TEXT ARRAY
20460C     IASCII- OUTPUT ARRAY IN UASCII CODE
20470C
20480         DIMENSION TEXT(N),IASCII(N),XTEXT(72),USASCII(63)
20490         INTEGER IASCII,USASCII,IVAL,FORM
20500         INTEGER XTEXT
20510C
20520C    DATA FOR 1-63 DECIMAL 6-BIT BYTE CHARACTER SET
20530C
20540C    DATA FOR A-Z
20550         DATA USASCII /65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,8
20560+                      81,82,83,84,85,86,87,88,89,90,
20570C    DATA FOR 0-9
20580+                      48,49,50,51,52,53,54,55,56,57,
20590C    DATA FOR +-*/()$= ,.'[]:'
20600+                      43,45,42,47,40,41,36,61,32,44,46,34,91,93,58,39
20610C    DATA FOR X-ON,CR,LF,↑,X-OFF,<,>,UNUSED,?,UNUSED,¡
20620+                      32,13,10,94,32,60,62,32,63,32,59/
20630C
20640C    CREATE VARIABLE FORMAT TO BE USED BY DECODE
20650C
20660         ENCODE(6,101,FORM)N
20670     101 FORMAT(1H(,I2,3HR1))
20680C
20690C    BREAK UP TEXT STRING TO ONE R1 CHARACTER PER WORD
20700C
20710         DECODE(N,FORM,TEXT)XTEXT
20720C
20730C    TABLE LOOK-UP
20740C
20750         DO 10 I=1,N
20760         IVAL=XTEXT(I)
20770         IASCII(I)=USASCII(IVAL)
20780C
20790      10 CONTINUE
20800         RETURN
20810         END
20820C
20830C
20840C
20850         SUBROUTINE TITLE
20860C
20870C    OBJECTIVE:
20880C     TO DISPLAY THE TWO EXPANDED COUNTRY NAMES (30 CHARACTERS)
20890C     IN THE TOP CENTER OF THE CRT AFTER ANY TYPE OF PLOT
20900C     THE FIRST NAME WILL BE THE INITIATOR.  THE SECOND
20910C     THE RECIPIENT.  IN BETWEEN THESE WILL BE AN ARROW
20920C      DRAWN WITH PLOT-10 TCS CALLS.  THE ARROW WILL HAVE
20930C      EITHER A DOUBLE POINT DENOTING 2-WAY FLOW OR ONE
20940C     POINT DENOTING A SINGLE FLOW.  IN THE CASE OF A
20950C      SINGLE FLOW FROM RECIPIENT TO INITIATOR THE CI +
20960C     CR, NI AND NR, ADI AND ADR WILL BE FLIP-FLOPPED.
20970C
20980         COMMON /PARAM/ CI,NI,ADI(3),CR,NR,ADR(3),DUAL,
20990+                       SDATE,FDATE,TCODE,SMODE,R,ISPEED,INITAB
21000         INTEGER ADR,ADI,INITR(30),RECIP(30)
```

```
21010         LOGICAL DUAL
21020         CALL CHCNVT(30,ADI,INITR)
21030         CALL CHCNVT(30,ADR,RECIP)
21040        DO 20 I=1,30
21050        IF(INITR(I).EQ.32.A.INITR(I+1).EQ.32)GOTO 21
21060        20 CONTINUE
21070        21 ILEN=I
21080        DO 30 J=1,30
21090        IF(RECIP(J).EQ.32.A.RECIP(J+1).EQ.32)GOTO 31
21100        30 CONTINUE
21110        31 JLEN=J
21120        CALL JUSTER(ILEN,INITR,+1,32,LEN,IOFF)
21130        CALL NOTATE(410+IOFF,765,LEN,INITR)
21140        IX=450
21150        IY=775
21160C
21170C    DRAW ARROW
21180C
21190           CALL MOVABS(IX,IY)
21200           CALL DRWREL(100,0)
21210           CALL DRWREL(-20,5)
21220           CALL MOVREL(0,-10)
21230           CALL DRWREL(20,5)
21240           IF(.NOT.DUAL) GOTO 10
21250           CALL MOVREL(-100,0)
21260           CALL DRWREL(20,5)
21270           CALL MOVREL(0,-10)
21280           CALL DRWREL(-20,5)
21290C
21300           10 CALL JUSTER(JLEN,RECIP,-1,32,LEN1,IOFF1)
21310           CALL NOTATE(600,765,LEN1,RECIP)
21320           RETURN
21330           END
21340C
21350C
21360C
21370            SUBROUTINE LGND(IANS,IND)
21380C
21390C    OBJECTIVE:
21400C     IF IGT = 'Y' THEN THE USER HAS CHOSEN A MULTI-GRAPH
21410C     GRAPH, AND JUST BELOW THE TITLE.  THE LEGEND WILL
21420C     CONSIST OF THE ASSOCIATED LINE DRAWN ON THE GRAPH
21430C     FOLLOWED BY THE 3 OR 4 LETTER CODE NAMING IT.  SINCE
21440C     THE OPTIONS ALLOW ONLY 2 OR 3 CURVES ON A MULTI-GRAPH
21450C     THE LEGEND HAS ONLY TWO OR THREE LINES FOLLOWED BY THE
21460C      CURVE NAME.4
21465C
21470C    PARAMETERS:
21480C     IANS = NUMBER OF CURVES (2 OR 3)
21490C     IND  = THE MATRIX NUMBERS OF THESE CURVES (1-10)
21500C
21510           DIMENSION IND(6),ILINE(2),ITITL(10),IPRT(6)
21520           DIMENSION IFST(4),ISND(4),ITRD(4)
21530           DATA ILINE /5454,1212/
21540           DATA ITITL /4HTOT ,4HTOTZ,4HCOO ,4HCOOZ,4HCON ,4HCONZ,
21550+          4HTEN ,4HTENZ,4HUNC ,4HUNCZ/
21560           CALL MOVABS(250,720)
21570           CALL DRWABS(350,720)
21580           IF(IANS.EQ.2) GOTO 10
21590           CALL MOVABS(450,720)
21600           CALL DSHABS(550,720,ILINE(1))
21610           N=2
21620           GOTO 30
21630        10 N=1
21640        30 CONTINUE
21650           CALL MOVABS(650,720)
21660           CALL DSHABS(750,720,ILINE(N))
21670           CALL CHCNVT(4,ITITL(IND(1)),IFST)
```

```
21680          CALL CHCNUT(4,ITITL(IND(2)),ISND)
21690          IF(IANS.EQ.3) CALL CHCNUT(4,ITITL(IND(3)),ITRD)
21700          CALL NOTATE(370,720,4,IFST)
21710          IF(IANS.EQ.2) GOTO 60
21720          CALL NOTATE(570,720,4,ISND)
21730          N=3
21740          GOTO 70
21750       60 N=2
21760       70 CONTINUE
21770          IF(N.EQ.2) CALL NOTATE(770,720,4,ISND)
21780          IF(N.EQ.3) CALL NOTATE(770,720,4,ITRD)
21790          RETURN
21800          END
21810C
21820C
21830C
21840C
21850          OVERLAY(OVRLAY,5,0)
21860C
21870          PROGRAM TABULAR
21880C
21890C    OBJECTIVE:
21900C     THIS OVERLAY WILL PRINT THE ARRAYS GENERATED IN
21910C     OVERLAY 2,0 IN TABULAR FORM
21915C
21920C    SUBROUTINES:
21930C     DEKODE - ROUTINE RETURNS THE ALPHA MONTH AND YEAR
21940C              FROM AN INTEGER I6 INPUT
21950C
21960          IMPLICIT INTEGER(A-Z)
21970C
21980          COMMON /PARAM/ CI,NI,ADI(3),CR,NR,ADR(3),DUAL,
21990+                        SDATE,FDATE,TCODE,SMODE,R,ISPEED,INITAB
22000          COMMON /RESLT/ COUNT,IC,DATE(36),EVENT(22,36),
22010+                        COOP(36),CONFL(36),TOTAL(36),ATOTAL(36),
22020+                        ZTOTAL(36),ACOOP(36),ZCOOP(36),ACONFL(36),
22030+                        ZCONFL(36),TF(36),ZST(36),HR(36),ZSH(36),
22040+                        PTOTAL(36),PCOOP(36),PCONFL(36),RDATE(36)
22050C
22060          CHARACTER*3 CI,CR
22070          CHARACTER ADI,ADR
22080          LOGICAL DUAL
22090          CHARACTER*4 FVCODE
22100          DIMENSION FVCODE(22)
22110           CHARACTER*9 INTRVL(3)
22120          CHARACTER*11 STYPE(2)
22130          REAL ZTOTAL,ZCOOP,ZCONFL,ZST,ZHR,TF,HR
22140          REAL PTOTAL,PCOOP,PCONFL
22150C
22160          DATA NE/22/
22170          DATA INTRVL /'  MONTHLY','QUARTERLY','   YEARLY'/
22180          DATA STYPE/'NORMAL','PROGRESSIVE'/
22190          DATA FVCODE/'YLD ','CMNT','CNSL','APPR','PRMS','GRNT',
22200+                     'REWD','AGRE','RQST','PROP','RJCT','ACUS',
22210+                     'PROT','DENY','DMND','WARN','THRT','DEMO',
22220+                     'RDUC','EXPL','SEIZ','FRCE'/
22230C
22240          CALL DEKODE(SDATE,ISM,ISY)
22250          CALL DEKODE(FDATE,IFM,IFY)
22260C
22270    4006 FORMAT(22X,A9,1X,'COOPERATIVE ACTIVITY')
22280    4009 FORMAT(22X,A9,1X,'CONFLICTUAL ACTIVITY')
22290    4016 FORMAT(27X,A3,',',I4,' - ',A3,',',I4,/)
22300    4005 FORMAT(11X,'TOTAL ACTIVITY',6X,'COOPERATIVE ACTIVITY',
22310+       2X,'CONFLICTUAL ACTIVITY',/,2X,'DATE',3X,
22320+       3('NUMBER  Z-SCORE PROB  ')/)
22330    6005 FORMAT(11X,'TOTAL ACTIVITY',6X.'COOPERATIVE ACTIVITY',
22340+       2X,'CONFLICTUAL ACTIVITY',/,2X,'DATE',3X,
```

```
22350+      2X,'NUMBER',4X,'Z-SCORE',3X,2('NUMBER',7X,'Z-SCORE',2X)/)
22360       CALL BLANC
22370       WRITE(6,4001)INTRVL(TCODE),ISM,ISY,IFM,IFY
22380  4001 FORMAT(27X,A9,1X,'ACTIVITY',/,26X,
22390+      A3,'  ',I4,' - ',A3,'  ',I4,/)
22400       IFLO = 3HTWO
22410       IDIR=10H<<<<<<<<<<
22420       IF(.NOT.DUAL)IFLO=3HONE
22430       IF(.NOT.DUAL)IDIR=10H>>>>>>>>>>
22440       WRITE(6,4002)CI,IDIR,IFLO,CR
22450  4002 FORMAT(17X,'***',1X,A3,1X,A6,1X,A3,
22460+      '-WAY FLOW >>>>>> ',A3,' ***',/)
22470       IF(TCODE.EQ.1)WRITE(6,4005)
22480       IF(TCODE.NE.1)WRITE(6,6005)
22490       NC=IC
22500       IF(IC.GT.18)500,501
22510   500 NC=18
22520   501 DO 350 I=1,NC
22530       IF(TCODE.NE.1) GOTO 600
22540       WRITE(6,5001) RDATE(I),TOTAL(I),ZTOTAL(I),PTOTAL(I),
22550+                    COOP(I),ZCOOP(I),PCOOP(I),
22560+                    CONFL(I),ZCONFL(I),PCONFL(I)
22570  5001 FORMAT(1X,A6,'    ',3(I4,2X,F7.2,1X,F5.2,3X))
22580       GOTO 350
22590   600 WRITE(6,6001) RDATE(I),TOTAL(I),ZTOTAL(I),COOP(I),ZCOOP(I),
22600+                    CONFL(I),ZCONFL(I)
22610   350 CONTINUE
22620    100 FORMAT(A10)
22630        READ 100,I
22640       IF(IC.GT.18) 502,503
22650   502 NF=NC+1
22660       CALL BLANC
22670       WRITE(6,4001)INTRVL(TCODE),ISM,ISY,IFM,IFY
22680       WRITE(6,4002)CI,IDIR,IFLO,CR
22690       IF(TCODE.EQ.1)WRITE(6,4005)
22700       IF(TCODE.NE.1)WRITE(6,6005)
22710       DO 355 I = NF,IC
22720       IF(TCODE.NE.1)GOTO 601
22730       WRITE(6,5001) RDATE(I),TOTAL(I),ZTOTAL(I),PTOTAL(I),
22740+                    COOP(I),ZCOOP(I),PCOOP(I),
22750+                    CONFL(I),ZCONFL(I),PCONFL(I)
22760       GOTO 355
22770   601 WRITE(6,6001) RDATE(I),TOTAL(I),ZTOTAL(I),COOP(I),ZCOOP(I),
22780+                    CONFL(I),ZCONFL(I)
22790  6001 FORMAT(1X,A6,'    ',I4,7X,F7.2,3X,
22800+      2(I4,8X,F7.2,3X))
22810   355 CONTINUE
22820       READ 100,I
22830   503 CALL BLANC
22840C
22850       WRITE(6,4011)INTRVL(TCODE),ISM,ISY,IFM,IFY
22860  4011 FORMAT(20X,A9,1X,'TENSION AND UNCERTAINTY',/,
22870+      26X,A3,'  ',I4,' - ',A3,'  ',I4,/)
22880       WRITE(6,4002)CI,IDIR,IFLO,CR
22890       WRITE(6,5002)
22900  5002 FORMAT(17X,'  DATE   TENSION Z-SCORE   H-REL Z-SCORE'/)
22910       NC=IC
22920       IF(IC.GT.18) 510,511
22930   510 NC=18
22940   511 DO 380 I=1,NC
22950       WRITE(6,5006) RDATE(I),TF(I),ZST(I),HR(I),ZSH(I)
22960  5006 FORMAT(18X,A6,'    ',F5.1,2X,F7.2,3X,F5.3,1X,F7.2)
22970   380 CONTINUE
22980       READ 100,I
22990       IF(IC.GT.18) 512,513
23000   512 NF=NC+1
23010       CALL BLANC
23020       WRITE(6,4011)INTRVL(TCODE),ISM,ISY,IFM,IFY
```

```
23030         WRITE(6,4002)CI,IDIR,IFLO,CR
23040         WRITE(6,5002)
23050         DO 385 I = NF,IC
23060         WRITE(6,5006)RDATE(I),TF(I),ZST(I),HR(I),ZSH(I)
23070    385  CONTINUE
23080         READ 100,I
23090    513  CALL BLANC
23100         PRINT,$ DO YOU WANT EVENT FREQUENCIES (Y OR N)$,
23110         READ 100,I
23120         IF(I.EQ.1HY)359,390
23130    359  CALL BLANC
23140         WRITE(6,4006)INTRVL(TCODE)
23150         WRITE(6,4016)ISM,ISY,IFM,IFY
23160         WRITE(6,4002)CI,IDIR,IFLO,CR
23170         WRITE(6,4007)(FVCODE(E),E=1,10)
23180   4007  FORMAT(10X,'DATE',4X,12(A4,1X))
23190         WRITE(6,4008)
23200   4008  FORMAT(' ')
23210         NC=IC
23220         IF(IC.GT.18) 520,521
23230    520  NC=18
23240    521  DO 360 I=1,NC
23250         WRITE(6,5005)RDATE(I),(EVENT(E,I),E=1,10)
23260   5005  FORMAT(9X,A6,'  ',12(I4,1X))
23270    360  CONTINUE
23280         READ 100,I
23290         IF(IC.GT.18)522,523
23300    522  NF=NC+1
23310         CALL BLANC
23320         WRITE(6,4006)INTRVL(TCODE)
23330         WRITE(6,4016)ISM,ISY,IFM,IFY
23340         WRITE(6,4002)CI,IDIR,IFLO,CR
23350         WRITE(6,4007)(FVCODE(E),E=1,10)
23360         WRITE(6,4008)
23370         DO 365 I=NF,IC
23380         WRITE(6,5005) RDATE(I),(EVENT(E,I),E=1,10)
23390    365  CONTINUE
23400         READ 100,I
23410    523  CALL BLANC
23420         WRITE(6,4009)INTRVL(TCODE)
23430         WRITE(6,4016)ISM,ISY,IFM,IFY
23440         WRITE(6,4002)CI,IDIR,IFLO,CR
23450         WRITE(6,6007)(FVCODE(E),E=11,22)
23460   6007  FORMAT(6X,'DATE',4X,12(A4,1X))
23470   6009  FORMAT(5X,A6,'  ',12(I4,1X))
23480         WRITE(6,4008)
23490         NC=IC
23500         IF(IC.GT.18)530,531
23510    530  NC=18
23520    531  DO 370 I=1,NC
23530         WRITE(6,6009)RDATE(I),(EVENT(E,I),E=11,22)
23540    370  CONTINUE
23550         READ 100,I
23560         IF(IC.GT.18)532,390
23570    532  NF=NC+1
23580         CALL BLANC
23590         WRITE(6,4009)INTRVL(TCODE)
23600         WRITE(6,4016)ISM,ISY,IFM,IFY
23610         WRITE(6,4002)CI,IDIR,IFLO,CR
23620         WRITE(6,6007)(FVCODE(E),E=11,22)
23630         WRITE(6,4008)
23640         DO 375 I=NF,IC
23650         WRITE(6,6009)RDATE(I),(EVENT(E,I),E=11,22)
23660    375  CONTINUE
23670         READ 100,I
23680     390  CONTINUE
23690          END
23700C
```

```
23710C
23720C
23725C
23726C
23730          SUBROUTINE DEKODE(ID,IMONTH,IYEAR)
23735C
23736C          SUBROUTINE DESCRIBED IN OVERLAY 2,0
23737C
23740          DIMENSION MONTHS(12)
23750          DATA MONTHS /3HJAN,3HFEB,3HMAR,3HAPR,3HMAY,3HJUN,
23760+                       3HJUL,3HAUG,3HSEP,3HOCT,3HNOV,3HDEC/
23770          IY=ID/10000
23780          IM=(ID-10000*IY)/100
23790          IMONTH=MONTHS(IM)
23800          IYEAR=(IY+1900)
23810          RETURN
23820          END
23830C
23840C
23850C
23860C
```

APPENDIX A

NATION FILE

NATION FILE

```
700 AFG AFGHANISTAN
339 ALB ALBANIA
615 ALG ALGERIA
232 AND ANDORRA
561 ANG ANGOLA
160 ARG ARGENTINA
900 AUL AUSTRALIA
305 AUS AUSTRIA
695 BAH BAHRAIN
053 BAR BARBADOS
211 BEL BELGIUM
266 EBE BERLIN/EAST
267 WBE BERLIN/WEST
760 BHU BHUTAN
765 BGD BANGLADESH
145 BOL BOLIVIA
571 BOT BOTSWANA
140 BRA BRAZIL
355 BUL BULGARIA
775 BUR BURMA
516 BUI BURUNDI
811 CAM CAMBODIA
471 CAO CAMEROUN
020 CAN CANADA
431 CPV CAPE VERDE
482 CEN CENTRAL AFRICAN REP
780 CEY CEYLON
483 CHA CHAD
155 CHL CHILE
710 CHN CHINA, PEOPLES REP
713 CHT CHINA, REP OF
100 COL COLUMBIA
495 COM COMORO ISLANDS
484 CON CONGO (BRAZZAVILLE)
490 COP ZAIRE (CONGO-KINSHASA)
094 COS COSTA RICA
040 CUB CUBA
352 CYP CYPRUS/GREEK
353 CYT CYPRUS/TURKISH
315 CZE CZECHOSLOVAKIA
434 DAH DAHOMEY
390 DEN DENMARK
042 DOM DOMINICAN REP
130 ECU ECUADOR
092 ELS EL SALVADOR
440 GUE EQUITORIAL GUINEA
531 ELF ERITREAN LIBERATION FRONT
530 ETH ETHIOPIA
375 FIN FINLAND
220 FRN FRANCE
900 FIJ FIJI
555 FRE FRELIMO-MOZAMBIQUE LIB. ORG.
481 GAB GABON
420 GAM GAMBIA
265 GME GERMANY/DEM REP
255 GMW GERMANY/FED REP
452 GHA GHANA
350 GRC GREECE
090 GUA GUATEMALA
438 GUI GUINEA
441 GBI GUINEA-BISSAU
110 GUY GUYANA
```

041 HAI HAITI
091 HON HONDURAS
310 HUN HUNGARY
720 HOK HONG KONG
395 ICE ICELAND
750 IND INDIA
850 INS INDONESIA
630 IRN IRAN
645 IRQ IRAQ
205 IRE IRELAND
666 ISR ISRAEL
325 ITA ITALY
437 IVO IVORY COAST
051 JAM JAMAICA
740 JAP JAPAN
663 JOR JORDAN
501 KEN KENYA
731 KON KOREA/NORTH
732 KOS KOREA/SOUTH
641 KUR KURDISTAN
690 KUW KUWAIT
812 LAO LAOS
660 LEB LEBANON
570 LES LESOTHO
450 LBR LIBERIA
620 LBY LIBYA
223 LIC LIECHTENSTEIN
212 LUX LUXEMBURG
721 MAC MACAO
580 MAG MALAGASY
553 MAW MALAWI
820 MAL MALAYSIA
782 MAD MALDIVE
432 MLI MALI
338 MLT MALTA
590 MAR MAURITIUS
435 MAU MAURITANIA
070 MEX MEXICO
221 MOC MONACO
712 MON MONGOLIA
600 MOR MOROCCO
698 MOM MUSCAT AND OMAN
562 NMB NAMIBIA
921 NAU NAURU
921 NAU NAURU
790 NEP NEPAL
210 NTH NETHERLANDS
920 NEW NEW ZEALAND
093 NIC NICARAGUA
436 NIR NIGER
475 NIG NIGERIA
385 NOR NORWAY
770 PAK PAKISTAN
095 PAN PANAMA
905 PAP PAPUA NEW GUINEA
150 PAR PARAGUAY
135 PER PERU
840 PHI PHILIPPINES
290 POL POLAND
235 POR PORTUGAL
485 STV PRINCIPEE AND SAO TOMEE
696 QAT QATAR
552 RHO RHODESIA
360 RUM RUMANIA
517 RWA RWANDA
331 SAN SAN MARINO
670 SAU SAUDI ARABIA
433 SEN SENEGAL

451 SIE SIERRA LEONE
830 SIN SINGAPORE
520 SOM SOMALIA
560 SAF SOUTH AFRICA
681 SYE SOUTH YEMEN
230 SPN SPAIN
625 SUD SUDAN
105 SUR SURINAM
572 SWA SWAZILAND
380 SWD SWEDEN
225 SWZ SWITZERLAND
652 SYR SYRIA
510 TAZ TANZANIA
800 TAI THAILAND
861 TIB TIBET
461 TOG TOGO
052 TRI TRINIDAD-TOBAGO
616 TUN TUNISIA
640 TUR TURKEY
500 UGA UGANDA
365 USR USSR
675 UAE UNITED ARAB EMIRATES
651 UAR UAR (EGYPT)
200 UNK UNITED KINGDOM
002 USA USA
439 UPP UPPER VOLTA
165 URU URUGUAY
328 VAT VATICAN
101 VEN VENEZUELA
817 VTS VIETNAM/SOUTH
816 VTN VIETNAM/NORTH
990 WSM WESTERN SAMOA
678 YEM YEMEN
681 SYE YEMAN/SOUTH
345 YUG YUGOSLAVIA
551 ZAM ZAMBIA
554 ZIM ZIMBABWE-RHODESIA LIB. ORG.
198 AFP ALLIANCE FOR PROGRESS
699 ARL ARAB LEAGUE
476 BIA BIAFRA
397 EEC EEC
398 EFT EFTA
396 NAT NATO
199 OAS OAS
599 OAU OAU
697 PLO ARAB COMMANDO GROUPS
813 LAP PATHET LAO
992 SEA SEATO
818 VCG VIETCONG AND NLF
394 WAR WARSAW PACT
399 UNO ANY INTL ORG (UN, ETC)
998 MLG ANY MULTILATERAL GRP
999 NSC NOT STATED, UNIDENTIFIED TRGT

APPENDIX B

RWINDX FILE

RWINDX FILE

6601	1
6602	63
6603	116
6604	163
6605	190
6606	247
6607	286
6608	365
6609	422
6610	480
6611	540
6612	596
6701	639
6702	682
6703	763
6704	796
6705	839
6706	877
6707	952
6708	979
6709	1024
6710	1066
6711	1091
6712	1110
6801	1122
6802	1157
6803	1188
6804	1213
6805	1253
6806	1322
6807	1360
6808	1455
6809	1535
6810	1618
6811	1665
6812	1692
6901	1725
6902	1774
6903	1826
6904	1936
6905	1999
6906	2061
6907	2081
6908	2125
6909	2167
6910	2208
6911	2275
6912	2307
7001	2338
7002	2367
7003	2415
7004	2447
7005	2483
7006	2528
7007	2558
7008	2597
7009	2621
7010	2650
7011	2739
7012	2812
7101	2840
7102	2879

7103	2942
7104	2994
7105	3103
7106	3168
7107	3202
7108	3226
7109	3257
7110	3297
7111	3362
7112	3492
7201	3851
7202	3951
7203	4071
7204	4146
7205	4206
7206	4268
7207	4314
7208	4356
7209	4405
7210	4473
7211	4539
7212	4601
7301	4666
7302	4701
7303	4775
7304	4858
7305	4918
7306	4970
7307	5067
7308	5125
7309	5183
7310	5319
7311	5468
7312	5573
7401	5668
7402	5725
7403	5795
7404	5852
7405	5913
7406	5959
7407	5999
7408	6050
7409	6088
7410	6107
7411	6156
7412	6195
7501	6204
7502	6228
7503	6262
7504	6274
7505	6287
7506	6307
7507	6323
7508	6344
7509	6371
7510	6408
7511	6445
7512	6492
-0	6552

REFERENCES

Andriole, S.J. Progress Report on the Development of a Crisis Early Warning Prototype System. Technical Report 76-19. McLean, VA: Decisions and Designs, Incorporated, December, 1976.

Azar, E.E., and Ben-Dak, J.D., eds. Theory and Practice of Events Research. New York: Gordon and Breach, 1975.

Burgess, P.M., and Lawton, R.W. Indicators of International Behavior: An Assessment of Events Data Research. Sage International Studies Series, Vol. 1. Beverly Hills, CA, 1972.

Decisions and Designs, Incorporated. The Development of a Crisis Early Warning Prototype System. Technical Proposal. McLean, VA, December, 1975.

_____. The Development of a Crisis Early Warning Prototype System. Technical Proposal. McLean, VA, July, 1976.

Kegley, C.W.; Raymond, G.A.; Rood, R.M.; and Skinner, R.A., eds. International Events and the Comparative Analysis of Foreign Policy. Columbia, SC: University of South Carolina Press, 1975.

McClelland, C.A. "The Beginning, Duration, and Abatement of International Crises: Comparisons in Two Conflict Arenas." In C. F. Hermann, ed., International Crises: Insights from Behavioral Research, pp. 83-108. New York: Free Press, 1972.

_____. Theory and the International System. New York: Macmillan, 1966.

_____; Fitzsimmons, B; Hoggord, G; Martin, W.; and Young, R.A. World Event/Interaction Survey (WEIS) Handbook and Codebook. WEIS Technical Report. Los Angeles, CA: University of Southern California, January, 1969.

_____; Tomlinson, R.G.; Sherwin, R.G.; Hill, G.A.; Calhoun, H.L.; Fenn, P.H.; and Martin, J.D. The Management and Analysis of International Event Data: A Computerized System for Monitoring and Projecting Event Flows. WEIS Technical Report. Los Angeles, CA: University of Southern California, September, 1971.

Rubin, T.J.; Peasley, D.; and Hopkins, L. Dissemination and Evaluation of Quantitative International Affairs Indicators. Technical Report. Santa Barbara, CA: CACI, Incorporated, July, 1972.

Spector, B.I.; Brownell, J.R.; Hayes, M.D.; Keynon, G.A.; and Moore, J.A. Quantitative Indicators for Defense Analysis. Final Technical Report. Arlington, VA: CACI, Incorporated, June, 1975.

Young, R.A.; Moore, J.A.; and Moore, V. The Utilization of ARPA-Supported Research for National Security Planning. Technical Report. Arlington, VA: CACI, Incorporated, October, 1972.

Zinnes, D.A. Contemporary Research in International Politics. New York: Free Press, 1976.

DISTRIBUTION LIST

Contractual

Addressee	DODAAD Code	Number of Copies: Unclassified/Unlimited	Number of Copies: Unclassified/Limited and Classified
Dr. John Negay Scientific Officer Office of Naval Research Code 452 800 North Quincy Street Arlington, VA 22217	N00014	1	1
Mr. William Gaudette Administrative Contracting Officer DCASMA Baltimore Office 300 East Joppa Road Towson, MD 21204	S2101A	1	1
Director, Naval Research Laboratory Attn: Code 2627 Washington, DC 20375	N00014	6	1
Office of Naval Research International Programs Code 1021P Department of the Navy Arlington, VA 22217	N00014	6	0
Defense Documentation Center Attention: DDC-TC Bldg. 5, Cameron Station Alexandria, VA 22314	S47031	12	2
Office of Naval Research Branch Office 495 Summer Street Boston, MA 02210	N62879	1	1

Non-Contractual

Department of Defense

Dr. Robert A. Young, Director
DARPA/CTO
1400 Wilson Boulevard
Arlington, VA 22209

Dr. Stephen J. Andriole, Program Manager
DARPA/CTO
1400 Wilson Boulevard
Arlington, VA 22209

Dr. Craig Fields, Program Manager
DARPA/CTO
1400 Wilson Boulevard
Arlington, VA 22209

Dr. Franklin F. Kuo, Asst. Director
ADP/Teleprocessing
OSD (ODTACCS)
The Pentagon, Room 3D234
Washington, DC 20301

CDR C. W. Easterbrook
Warning & Surveillance Branch
WWMCCS Plans & Requirements Division
The Pentagon, Room 1C832
Washington, DC 20301

Mr. N. Birch
Assistant Director (Advanced Technology)
OSD/DTACCS
The Pentagon, Room 3E163
Washington, DC 20301

Mr. Wallace D. Henderson
Assistant Director of Defense Intelligence I&W and Intelligence-Related Activities
The Pentagon, Room 3C200
Washington, DC 20301

Mr. James Kelly
I&W Support
Operations Support Division
Defense Intelligence Agency
The Pentagon, Room 1C924A
Washington, DC 20301

Mr. Morton E. Goulder
Deputy Assistant Secretary (Intelligence & Warning Systems)
OSD/I
The Pentagon, Room 3E282
Washington, DC 20301

Special Assistant to the Principal Deputy Assistant Secretary of Defense (International Security Affairs)
The Pentagon, Room 4E825
Washington, DC 20301

Assistant Director (Net Technical Assessment)
Office of the Deputy Director of Defense Research and Engineering (Test and Evaluation)
The Pentagon, Room 3C125
Washington, DC 20301

Department of the Army

Commander, U.S. Army Intelligence
Threats Analysis Detachment
Arlington Hall Station
4000 Arlington Boulevard
Arlington, VA 22212

Director, Army Library
Army Studies (ASDIRS)
The Pentagon, Room 1A534
Washington, DC 20310

Director, Systems Review and Analysis Office
Office of the Deputy Chief of Staff for Research, Development and Acquisition
Department of the Army
The Pentagon, Room 3E426
Washington, DC 20310

Department of the Navy

Office of Naval Research
Information Systems Program (Code 437)
800 North Quincy Street
Arlington, VA 22217

Other Government Agencies

Dr. Robert M. Behr, Asst. Director
Military and Economic Affairs Bureau
US Arms Control and Disarmament Agency
Washington, DC 20451

Dr. Joseph Markowitz
Analytic Methodology
Room 632, Ames Building
CIA
Washington, DC 20505
Outer Envelope: CIA
Washington, DC 20505

Research and Development Institutions

Dr. Barry Blechman
The Brookings Institution
1775 Massachusetts Avenue, NW
Washington, DC 20036

Dr. Bradford Dismukes, Director
Plans and Policies Program
Institute of Naval Studies
Center for Naval Analyses
1401 Wilson Boulevard
Arlington, VA 22209

Dr. James Foster
RAND Corporation
1700 Main Street
Santa Monica, CA 90406

Dr. Richard Hayes
CACI, Incorporated - Federal
1815 North Fort Myer Drive
Arlington, VA 22209

Dr. Leo Hazlewood
CACI, Incorporated - Federal
1815 North Fort Myer Drive
Arlington, VA 22209

Dr. Craig Sherbrooke
Mathtech, Incorporated
4630 Montgomery Avenue
Bethesda, MD 20014

Dr. Robert Slator
Mathtech, Incorporated
4630 Montgomery Avenue
Bethesda, MD 20014

Colleges and Universities

Dr. Harold Guetzkow
Fulcher Professor of Decision-Making
in International Affairs
Scott Hall
Northwestern University
Evanston, IL 60201

Dr. Gerald W. Hopple
Department of Government and Politics
University of Maryland
College Park, MD 20742

Dr. Joseph B. Kadane
Department of Statistics
Carnegie-Mellon University
Pittsburgh, PA 15213

Dr. Charles McClelland
School of International Relations
University of Southern California
Los Angeles, CA 90007

Dr. Wayne R. Martin
School of Social and Behavioral Sciences
California State College, Dominguez Hills
Dominguez Hills, CA 90246

Dr. Paul J. Werbos
Department of Government and Politics
University of Maryland
College Park, MD 20742

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: TR 76-20	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): SOFTWARE DESIGN FOR AN INTERACTIVE CRISIS EARLY WARNING PROTOTYPE SYSTEM	5. TYPE OF REPORT & PERIOD COVERED: Technical Report. January 1-September 30, 1975.
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): James F. Wittmeyer	8. CONTRACT OR GRANT NUMBER(s): ONR N00014-76-C-0712
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Decisions and Designs, Incorporated, Suite 600, 8400 Westpark Drive, McLean, VA 22101	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS: Defense Advanced Research Projects Agency, 1400 Wilson Boulevard, Arlington, VA 22209	12. REPORT DATE: December 1976
	13. NUMBER OF PAGES: 118
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): Office of Naval Research, 800 North Quincy Street, Arlington, VA 22217	15. SECURITY CLASS. (of this report): Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

International Crisis	Country-by-Country Scanning
Early Warning System	Prototype Early Warning System
Quantitative Indicators	Super FORTRAN Source Coded
Multi-Method Forecasting	Interactive Time Sharing Program
Probabilities	Graphics Display

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Statement of Problem and Proposed Solution. This report describes in detail one component of the development of an integrated crisis warning system, the design and use of interactive computer software necessary for the development of such a system. The component was developed for the Defense Advanced Research Projects Agency's Cybernetics Technology Office during the period from January 1 to September 30, 1976. Interactive software is

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

an essential component of the development, testing, and evaluation of a user-oriented integrated system and of prototype crisis warning systems.

Design, Test, and Demonstration of the First Stage Crisis Early Warning Prototype System. The prototype system, as designed, tested, and developed, was comprised of:

(1) five quantitative international political indicators;
(2) a forecasting capability which includes an international crisis object, a short-range retrospective goal, and an associative method; and
(3) a sophisticated computer base endowed with efficient information storage, retrieval, processing, and display capabilities.

The system has been tested for its ability or inability to forecast crises retrospectively. The demonstration system for the prototype is comprised of computer hardware and software necessary for the design, installation, and operation of interactive user-oriented systems.

System Description. The prototype system described in this report is the foundation upon which future systems will be built in parallel with the on-going development and application of quantitative indicators for early warning. The system is an overlayed FORTRAN program consisting of 1 main overlay, PROGRAM ANALYZ; 4 primary overlays, PROGRAMS DIALOG, SEARCH, PLOT, AND TABULAR; 26 subroutines; and 2 functions. The system is written in Super FORTRAN (a United Computing Systems, Incorporated compiler), and requires a time-sharing driver resident on a host mainframe. The program is relatively inefficient without the use of a Tektronix graphics terminal.

Sample Program Output. A sample of the output from the crisis early warning prototype system is notated to describe the area in the program from which the output is generated.

System Design. A graph of the crisis early warning prototype shows the relationship each overlay and its associated subroutines has to the main overlay as well as to each other.

System Flow Charts. The main overlay and each of the primaries are flow charted. Calls to the subroutines are shown as predefined processes and not charted individually.

Documented FORTRAN Source Code. To allow access and ease of modification to any FORTRAN programmer, each routine is internally documented. Appendix A and B contain, respectively, the NATION file and the RWINDX file, the index for the events file by month.