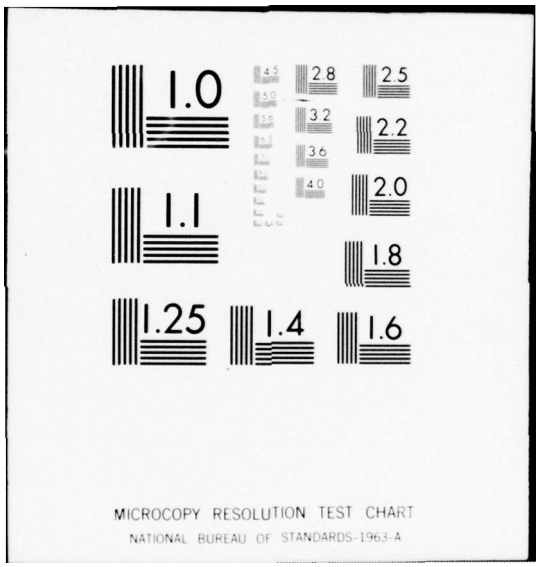
AD-A051 249 JOINT CONVENTIONAL AMMUNITION PROGRAM COORDINATING GR--ETC F/G 19/1
ANALYSTS' MANUAL FOR THE ITEM ACQUISITION/PRODUCTION TRADE-OFF --ETC(U)
DEC 77 G H MARTIN

UNCLASSIFIED JCAP-DM-T709 NL

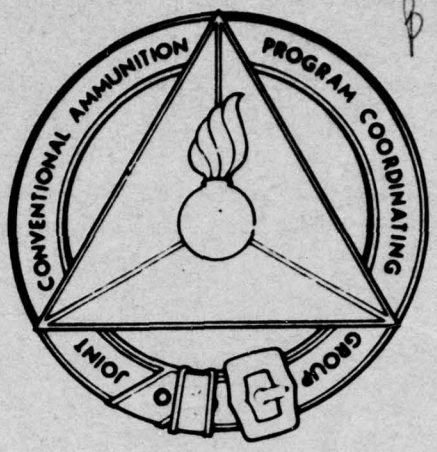
1 OF 1
AD A051249

END
DATE FILMED
4-78
DDC


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

(12)
B.S

AD
JCAP-DM-T709



ANALYSTS' MANUAL FOR THE ITEM ACQUISITION/PRODUCTION TRADE-OFF MODEL

COLD BASE VERSION

DECEMBER 1977

JOINT CONVENTIONAL
AMMUNITION PROGRAM COORDINATING GROUP
DECISION MODELS DIRECTORATE
ROCK ISLAND, ILLINOIS 61299

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER (11) JCAP-DM-T-709	2. GOVT ACCESSION NO. 3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) (6) Analysts' Manual for the Item Acquisition/Production Trade-Off Model. Cold Base Version.	5. TYPE OF REPORT & PERIOD COVERED. (9) Final Report, [redacted]
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) (10) George H. Martin	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Decision Models Directorate Joint Conventional Ammunition Program Coordinating Group, Rock Island, Illinois 61299	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS (11)
11. CONTROLLING OFFICE NAME AND ADDRESS Decision Models Directorate Joint Conventional Ammunition Program Coordinating Group, Rock Island, Illinois 61299	12. [redacted] Dec[redacted]77
	13. NUMBER OF PAGES 88 (15) 87 Pp.
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	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)

Procurement	Acquisition/Production Trade-Off	Ammunition Management
Stockpiling	Joint Conventional Ammunition Program	
Inventory	Computer Programming	Logistics
Production	Economics Analysis	Operations Research

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

The JCAP Item Acquisition/Production Trade-Off Model is a computerized decision model written in the FORTRAN, MPSX, and COBOL computer languages. The model is designed to develop an optimum cost-readiness relationship for an end item considering all available trade-off options that might meet requirements specified by logistics guidance. The model uses integer programming to identify specific optimum cost-readiness points either by maximizing readiness for a given cost or by minimizing cost for a given readiness.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE 410361

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Block 20 (Contd)

This volume contains:

(1) General background information to assist in understanding the environment for which the model is intended;

(2) A mathematical description of the model;

(3) A description of how the model is exercised; and

(4) An appendix with computer source listings.

JOINT CONVENTIONAL AMMUNITION PROGRAM
COORDINATING GROUP

REPLY TO
ATTENTION OF: Rock Island Arsenal, IL 61201

ANALYSTS' MANUAL
FOR THE
ITEM ACQUISITION PRODUCTION/TRADE-OFF MODEL

FOREWORD

In the Department of Defense environment, there is a need for the capability of evaluating the cost effectiveness of inventory acquisition and production trade-offs in the procurement process. This cost effectiveness is based on the optimum procurement plan for achieving a specified readiness. This plan is achieved by the utilization of component stockpiling and industrial preparedness measures (IPM's) and modernization investments.

To determine these optimum plans, based on joint specification by the Military Services, the Item Acquisition/Production Trade-Off Model was designed under the auspices of the Joint Conventional Ammunition Program Coordinating Group. This model has been successfully demonstrated by the Military Services.

This Analysts' Manual and a companion document, The Users' Manual, comprise an export package which will permit the Military Services to install and use this Item Acquisition/Production Trade-Off Model.

The Analysts' Manual consists of an explanation of the Item Acquisition/Production Trade-Off Model concept, along with appropriate uses of the model. It also describes in detail the input variables and how they are entered and arranged. Included also are descriptions of the model output and sample formats with descriptions of data input and output.

Configuration management of the model is retained by the Joint Conventional Ammunition Program Decision Models Directorate. Proposals for modification of the model and inquiries with respect to the model application and operation should be addressed to the Director, Joint Conventional Ammunition Program Decision Models Directorate, Rock Island Arsenal, IL 61299. Telephone inquiries should be addressed to the Chief, Item Acquisition and Materiel Planning Division, Decision Models Directorate, AUTOVON 793-5980.

JOINT CONVENTIONAL AMMUNITION PROGRAM
COORDINATING GROUP

Office of the Executive Director
Rock Island Arsenal, IL. 61201

ANALYSTS' MANUAL

FOR THE

ITEM ACQUISITION/PRODUCTION TRADE-OFF MODEL

This Analysts' Manual and a separately published Users' Manual provide detailed instructions and information for the Item Acquisition/Production Trade-Off (IA/PT) Model. The IA/PT Model was designed, developed, and demonstrated by the Joint Conventional Ammunition Program Decision Models Directorate in response to requirements established by the Military Services. The model has been accepted for their use as described herein.

Although the Item Acquisition/Production Trade-Off Model was designed to assist managers in the ammunition production base area, it is applicable to any commodity when the effects of inventory acquisition and production trade-off must be evaluated by decision makers.

EDWARD J. JORDAN
Executive Director

ABSTRACT

The JCAP Item Acquisition/Production Trade-Off Model is a computerized decision model written in the FORTRAN, MPSX, and COBOL computer languages. The model is designed to develop an optimum cost-readiness relationship for an end item considering all available trade-off options that might meet requirements specified by logistics guidance. The model uses integer programming to identify specific optimum cost-readiness points either by maximizing readiness for a given cost or by minimizing cost for a given readiness.

This volume contains:

(1) General background information to assist in understanding the environment for which the model is intended.

(2) A mathematical description of the model.

(3) A description of how the model is exercised.

(4) An appendix with computer source listings.

ACKNOWLEDGEMENTS

The contributions of the following individuals to the development, modification, application, and documentation efforts on this model are gratefully acknowledged.

DIRECTION AND INTEGRATION

Bernard C. Witherspoon, Director, JCAP Decision Models Directorate

Larry A. Guerrero, Chief, Item Acquisition & Materiel Planning Division, JCAP Decision Models Directorate

REPORT AUTHOR

George H. Martin, Operations Research Analyst, JCAP Decision Models Directorate

REPORT REVIEW AND EDITING ASSISTANCE

Richard Kurtz, Systems Analysis Directorate, HQ ARRCOM
Elizabeth Schwegler, Operations Research Analyst, JCAP Decision Models Directorate
Norman Hoesly, Operations Research Analyst, JCAP Decision Models Directorate
Thomas Short, Operations Research Analyst, JCAP Decision Models Directorate

MODEL DEVELOPMENT AND MODIFICATION

George H. Martin, Operations Research Analyst, JCAP Decision Models Directorate
Elizabeth Schwegler, Operations Research Analyst, JCAP Decision Models Directorate

SECRETARIAL SUPPORT

Lorraine Sabol, Secretary to the Chief, Item Acquisition and Materiel Planning Division, JCAP Decision Models Directorate
Elizabeth J. Foslien, Secretary to the Chief, Production and Mobilization Planning Division, JCAP Decision Models Directorate

SUMMARY

In May 1972 the Joint Logistics Commanders (JLC) established the Joint Conventional Ammunition Production Coordinating Group (JCAP/CG) and gave it the authority to coordinate and take action on all conventional ammunition production base activities and programs delegated by the respective commanders. The JCAP/CG basic charter was expanded in October 1974 to include conventional ammunition logistics programs and activities; and the name of JCAP/CG was changed to Joint Conventional Ammunition Program Coordinating Group. Under the sponsorship of the JCAP/CG, the Joint Conventional Ammunition Program Operating Group (JCAP/OG) has the responsibility for administering the Coordinated Management System for the DoD Conventional Ammunition Logistics Activities and Programs.

As directed by the JCAP/OG, the JCAP Decision Models Directorate (JCAP-DM) designs, develops, tests and provides guidance for implementation of all decision models, both economic and non-economic, required in the joint management of conventional ammunition logistics activities and programs.

The Joint Panel Report which led to the formation of JCAP states the motivation for development of the Item Acquisition/Production Trade-Off (IA/PT) Model: "An economic model(s) is needed that enables the determination of the most cost effective manner to program the ammunition production base so as to minimize the amount of inventory required while maximizing the responsiveness of the production base to meet wartime needs."

There are two ways of reducing inventory or inventory costs through use of the IA/PT model. Speed of production response to mobilization demands may be improved; or ammunition components may be stored instead of storing end items only. The IA/PT Model compares costs of increasing component and end item inventories with production response alternatives and identifies the least-cost alternative for improving readiness for management use.

The following table lists some of the end items studied during model development. The potential savings are found by comparing the IA/PT Model solution to buying end items only.

STUDY	ITEM	POTENTIAL COST AVOIDANCE ($ in millions)
JCAP MOBILIZATION LEADTIME STUDY APRIL 1975	CBU58B Bomb (Air Force)	67.7*
	MK82 Bomb (Navy)	77.5
	105mm HE, M1 (Army)	298.0
ARMCOM COMPONENT STOCKPILE STUDY APRIL 1975	81mm, M374A2 (Army)	212.0
	155mm, M107 (Army)	201.0
JCAP MOBILIZATION LEADTIME STUDY, SUPPLEMENTAL DECEMBER 1975	5"/54 Full Charge (Navy)	21.2
	5"/54 Projectile FCC (VT) (Navy)	20.7
	76mm Cartridge HE-IR (Navy)	19.3
JCAP CBU 58B STUDY MARCH 1976	CBU 58B Bomb (Air Force)	42.9
JCAP AMMUNITION READINESS STUDY FOR THE AIR FORCE FY79 BUDGET ESTIMATE JULY 1977	MK84 Bomb (Air Force)	205.0
	CBU MK20 Rockeye (Air Force)	4.8
	30mm API (Air Force)	6.7
	30mm HEI (Air Force)	54.9
	20mm HEI M56 (Air Force)	105.6
	TOTAL	$1,269.6

*Total does not include first CBU 58B study

This IA/PT Model Analysts' Manual contains basic model concepts, a mathematical description of the model, a program listing, and information needed to make computer runs. A companion IA/PT Model User's Manual describes input data and input data formats used in the model. For further information about the IA/PT Model and its application, the point of contact is Mr. George Martin (JCAP-DM) AUTOVON 793-5980, Commercial (309) 794-5980.

DISTRIBUTION LIST

Department of the Navy

Office of Assistant Secretary of the Navy (I&L)
ATTN: Dir, Ordnance Programs
Washington, DC 20360 (1)

Chief of Naval Operations
ATTN: OP411 (1)
OP411C (1)
Washington, DC 20350

Commandant
US Marine Corps
ATTN: LMG
Washington, DC 20380 (1)

Chief of Naval Materiel Command
ATTN: MAT-02 (1)
MAT-0242D (1)
Washington, DC 20362

Commander
Naval Sea Systems Command
ATTN: 04J (1)
Washington, DC 20362

Commander
Naval Air Systems Command
ATTN: AIR-05 (1)
AIR-532 (1)
Washington, DC 20361

Commander
Naval Supply Systems Command
NAVSSC Code 042
Rm 706 Crystal Mall 3
Washington, DC 20376 (1)

Commander
Naval Facilities Engineering Command
NAVFAC Code 202A
Washington, DC 20390 (1)

Commander
Naval Weapons Station Earle
Colts Neck, NJ 07722 (1)

Department of the Army

Office of the Assistant Secretary of the Army
(Research, Development & Acquisition)
Washington, DC 20310 (1)

Office DCS Research, Development & Acquisition
ATTN: DAMA-CSM (1)
DAMA-CSM-P (1)
DAMA-CS2-A (1)
Washington, DC 20310

Commander
US Army Materiel Development & Readiness Command
ATTN: DRCPA (1)
DRCCP (1)
DRCPP (1)
DRCSA-JS (1)
DRCLS (1)
Alexandria, VA 22333

Commander
US Army Armament Materiel Readiness Command
ATTN: DRSAR-DA (1)
DRSAR-PP (1)
DRSAR-PC (1)
DRSAR-PD (1)
DRSAR-IR (1)
DRSAR-MA (1)
DRSAR-QA (1)
DRSAR-AS (1)
DRSAR-TM (1)
DRSAR-SA (1)
DRSAR-IS (1)
DRSAR-MS (1)
DRSAR-OP (1)
DRSAR-IL (1)
DRSAR-CP (1)
DRSAR-PA (1)
DRSAR-MM (1)
DRSAR-LE (1)
Rock Island, IL 61299

DISTRIBUTION LIST (Cont'd)

Department of the Army (Cont'd)

Industrial Base Engineering Activity
ATTN: DRXIB-IP
Rock Island, IL 61299 (1)

Commander
Harry Diamond Laboratory
Washington, DC 20783 (1)

HQ
JUSMAG-K (Box 11)
ATTN: MKLO-S
APO San Francisco 96302 (1)

Commander
US Army Logistics Center
ATTN: ATCL-CG
Ft. Lee, VA 23801 (1)

Commander
US Army Armament Research & Development Command
Dover, NJ 07801 (1)

Commander
US Army Research Office
P. O. Box 12211 (1)
Research Triangle Park, NC 27709

Project Manager
Munitions Production Base Modernization & Expansion
ATTN: DRCPM-PBM
Dover, NJ 07801 (3)

Project Manager
Selected Ammunition
ATTN: DRCPM-SA
Dover, NJ 07801 (1)

Director
Army Materiel Systems Analysis Agency
ATTN: DRXSY-DI (1)
Aberdeen Proving Grounds, MD 21005

Department of the Air Force

Office of Assistant Secretary of the Air Force (I&L)
ATTN: SAF/ILP
Rm 4D862 Pentagon
Washington, DC 20330 (1)

Deputy Chief of Staff System & Logistics
ATTN: AF/LG (1)
AF/LGYW (1)
AF/LGYWC (1)
Rm 4E262 Pentagon
Washington, DC 20330

Deputy Chief of Staff R&D
ATTN: AF/RDQ (1)
AF/RDP (1)
Washington, DC 20330

Commander
Air Force Systems Command
ATTN: SDZA (1)
JTCG/MD (1)
Andrews AFB, MD 20331

Commander
Air Force Logistics Command
ATTN: MMWM (1)
XR (1)
XRX (1)
XRP (1)
LO (1)
LOWM (1)
Wright-Patterson AFB, OH 45433

Commander
Ogden Air Logistics Center
ATTN: MMJ
Hill AFB, UT 84406 (1)

Commander
Armament Development & Test Center
ATTN: SDC
Eglin AFB, FL 32542 (1)

DISTRIBUTION LIST (Cont'd)

Office, Secretary of Defense

Office of Director of Defense
Research & Engineering
ATTN: Standardization and Support,
Rm 2A318 Pentagon (1)
Ammunition Programs, Rm 2B284
Pentagon (1)
Washington, DC 20301

DoD Information Distribution

Defense Documentation Center
Cameron Station
Alexandria, VA 22314 (12)

Service Schools

Industrial College of the Armed
Forces
Ft. Leslie J. McNair
Washington, DC 20319 (1)

National War College
Ft. Leslie J. McNair
Washington, DC 20319 (1)

Armed Forces Staff College
Norfolk, VA 23511 (1)

Naval Postgraduate School
Monterey, CA 93940 (1)

US Air Force Air University
Maxwell AFB, AL 36112 (1)

Defense System Management College
Ft. Belvoir, VA 22060 (1)

Superintendent
US Military Academy
ATTN: COL A. F. Grum
West Point, NY 10996 (1)

Service Schools (Cont'd)

Commander
US Army Logistics Management
Center
ATTN: SCAD
Ft. Lee, VA 23801 (1)

Commander
US Army Logistics Management
Center
Ft. Lee, VA 23801 (1)

JCAP Coordinating Group

Executive Director, JCAP/CG
Rock Island, IL 61299 (10)

JCAP Decision Models Directorate
Rock Island, IL 61299 (10)

Defense Logistics Agency

Commander
Defense Fuel Supply Center
Cameron Station
Alexandria, VA 22314 (1)

Director of Procurement
Defense Logistics Agency
Cameron Station
Alexandria, VA 22314 (1)

TABLE OF CONTENTS

FIGURES

TABLES

CHAPTER 1

INTRODUCTION

1-1. GENERAL

This Analysts' Manual is one of two manuals for the JCAP Item Acquisition/Production Trade-Off Model (IA/PT). It contains sufficient information to permit the reader to operate the model. The companion Users' Manual describes how to prepare the input and interpret the output.

1-2. BACKGROUND

In planning to meet mobilization requirements, it is necessary to consider the mix of inventory and production response. The IA/PT model was developed to evaluate the trade-offs involved. The following discussion provides a basis for understanding how the model functions.

a. Post D-day Concept

(1) The post D-day (D-day is the day hostilities are declared) concept is a planning method used for the evaluation of the capability of meeting combat requirements from inventory and production. It provides a tool for ammunition planning that is independent of the duration of the conflict, but which assures the defense planner a reasonable degree of readiness to meet wartime consumption. The classic inventory/production response trade-off is based on a known demand for the product under consideration. Knowing this demand, the planner establishes an appropriate stock level (inventory policy) and production response to meet the demand. Demand for an ammunition item is supported from inventory procured before D-day and from production after D-day. Combat ammunition uses requirements rates which eventually level off at a constant value. Since the production rate for an item will not be programmed to decrease during a mobilization period, it is only necessary that the IA/PT address satisfaction of the requirements up to the time that the production rate meets or exceeds the requirements (this time is called "P-day") or some earlier specified time.

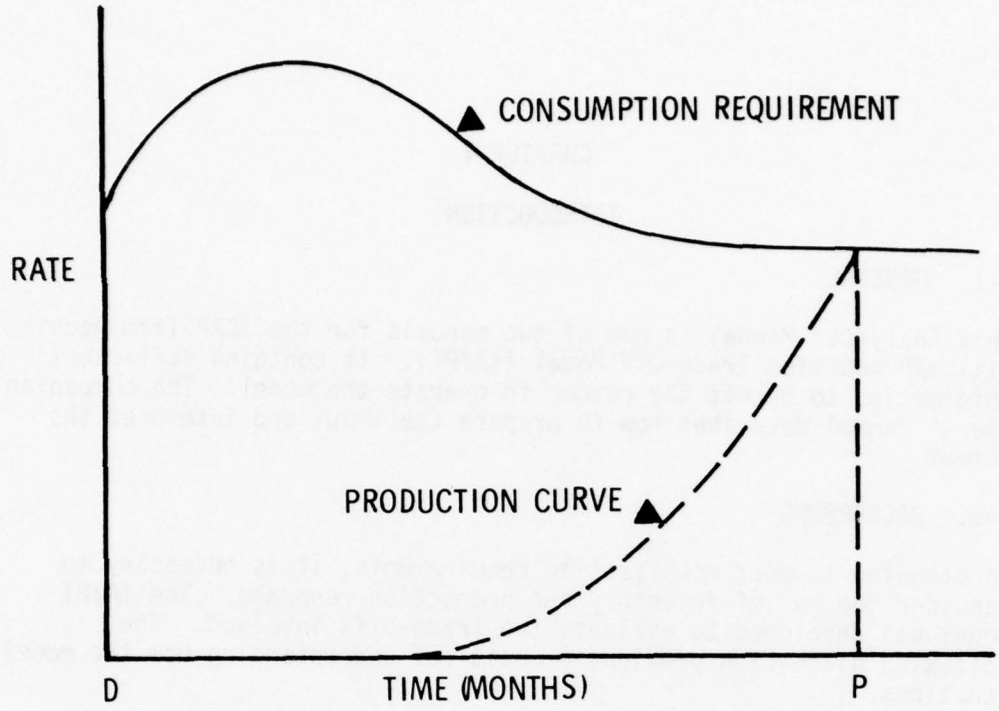


FIGURE 1. WARTIME PRODUCTION AND CONSUMPTION RATES VERSUS TIME

(2) Figure 1 illustrates projected ammunition consumption and production for an item. The vertical axis represents a rate (rounds/month) and the horizontal axis represents time (months). Therefore, an area on the chart represents a quantity of ammunition. The area under the production curve for a specified time period is the quantity of ammunition produced (the "production offset" for the period). The difference between what ought to be available for conflict (cumulative requirements) and what can be produced during conflict (production offset) must be stockpiled if requirements are to be met.

b. Inventory/Production Response Trade-Off

(1) Production rate decisions are sensitive to maintenance and layaway policies, modernization programs, item and component inventory, and production base activity. When the goal is to minimize cost, the planner should select the combination of item and component inventory and production response expenditures that incurs the least cost and still meets the desired level of readiness. If the funding level changes for these expenditures, the mobilization production response rate changes and the inventory required to provide a specified readiness changes.

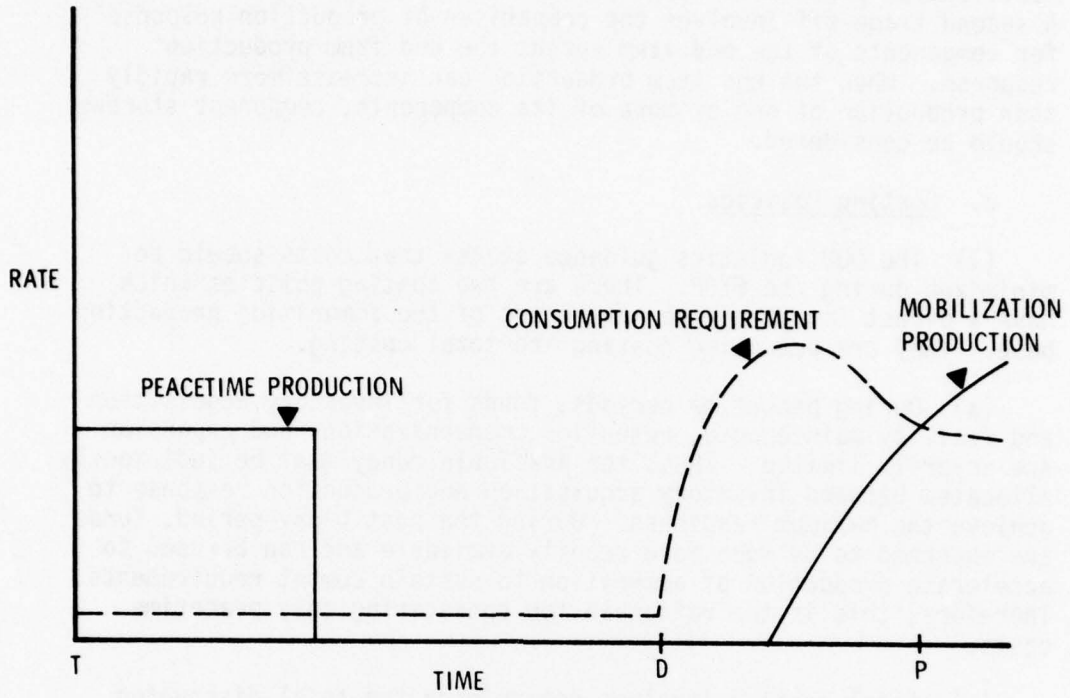


FIGURE 2. PRODUCTION RATE AND CONSUMPTION VERSUS TIME

(2) Figure 2 portrays major events and activities involved in the trade-off process. The end of the planning period and the beginning of the Five Year Defense Program (FYDP) is designated as T-day. Assets-on-hand (inventory) as of T-day are considered to have been acquired with sunk funds (funds that have been committed) and are not available for trade-off and so are not a part of the current decision process. (Assets on hand contribute to current readiness but can't be converted to funds to improve production response). The production curve (solid line) can be moved as a result of decisions made by the planner.

(3) To exactly meet requirements, initial assets plus production must equal consumption plus quantity in transit (not shown in figures). Because mobilization production cannot start immediately on D-day, a large part of the combat requirements must be met by inventory. This is illustrated in Figure 2. The inventory accumulated in peacetime must be adequate to support consumption until mobilization production can increase to support consumption needs (materiel in transit must

be included). One of the trade-offs to be considered is improving mobilization production response so that inventory may be decreased. A second trade-off involves the comparison of production response for components of the end item versus the end item production response. When the end item production can increase more rapidly than production of one or more of its components, component storage should be considered.

c. Costing Policies

(1) The DOD logistics guidance states that costs should be minimized during the FYDP. There are two costing policies which have a direct impact on the management of the ammunition production base. They are peacetime costing and total costing.

(a) During peacetime periods, funds for inventory acquisition and facility maintenance, retention, modernization, and expansion are severely limited. Thus, the available money must be judiciously allocated between inventory acquisition and production response to achieve the maximum readiness. During the post D-day period, funds are expected to be much more readily available and can be used to accelerate production of ammunition to sustain combat requirements. Therefore, this is the rationale for considering only peacetime costs.

(b) Total costing involves determining the total discounted cost of inventory acquisition and facilities modernization and expansion for a specific time frame.

(2) Costing policies are significant in the management of the ammunition production base, but they have little impact on the modeling techniques used. The IA/PT model is normally operated with peacetime costing but can be modified for use with total costing.

d. Period Considered in IA/PT Analysis

Any planning period may be used with the IA/PT model. The period following D-day for which requirements and production are included in the analysis depends on the scenario desired for the study. Since the post-mobilization period is variable, the model can be used to perform short to long term planning. For the long term situation, a 24-month period is normally used. There is no need to consider post-mobilization periods extending past the point that production exceeds requirements. The use of the FYDP period in describing data inputs reflects the intended use of the IA/PT model in FYDP planning, but the user should select the period appropriate to the planning situation.

e. Model Application

(1) The IA/PT model makes trade-offs for a single end item at a time; it selects a plan specifying a combination of plant response, end item inventory, industrial preparedness measures (IPM's), and component inventory. The model provides either the least cost plan to meet a specified level of item readiness, or a maximum readiness plan for a specified budget.

(2) Ammunition readiness is a measure of the availability of an item relative to the requirements for that item. The IA/PT model uses a readiness measure, the readiness ratio (RR), to indicate the capability to support requirements following mobilization. The readiness ratio for an item defines a planned asset allocation until no assets remain (N-Day); subsequent requirements are supported only by production. An equation for readiness ratio may be written as follows, for each item:

$$RR = \frac{Assets_T + \sum_{1}^{5} (Proc + Prod\text{-}Losses) + \sum_{D}^{N} Production}{\sum_{D}^{N} Requirements}$$

Where

$Assets_T$	Represents those assets for which funds will have been committed by T-day
$\sum_{1}^{5}$ Proc + Prod	Total Peacetime Procurement and Production unfunded at T-day
$\sum_{1}^{5}$ Losses	Total of training and other peacetime ammunition uses
$\sum_{D}^{N}$ Production	The production offset through N-day
$\sum_{D}^{N}$ Requirements	Represents cumulative requirements through N-day (including materiel in transit)

(3) The use of the RR defined above is intended to apply only in the 0.0 to 1.0 interval. A current RR is calculated to project readiness obtained by using T-day assets and projected mobilization production response figures in the above equation. At the two extremes: if the "basic load", "ships stores" or similar stockage cannot be met, then no assets will be available for trade-off (N-day is the same as D-day) and current RR = 0.0; if assets equal or exceed cumulative requirements to P-day then the current RR is 1.0.

(4) The period over which requirements are considered is variable. This will be determined by the scenario or specific study constraints but in any case, extending time past P-day will have no effect on results from the model.

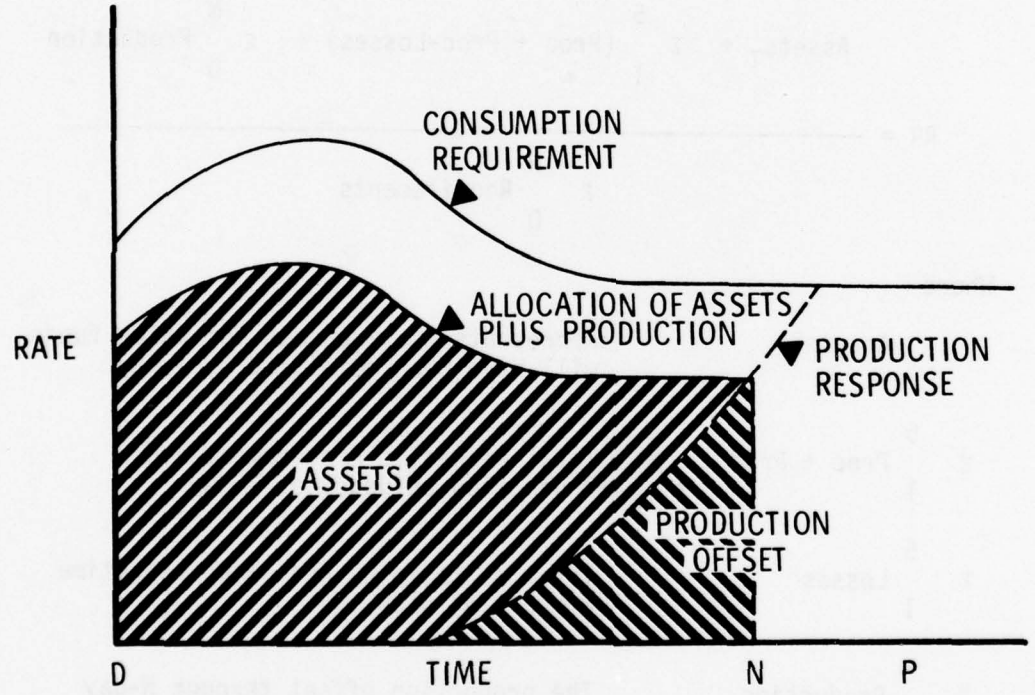


FIGURE 3. DETERMINATION OF N-DAY

(5) Figure 3 shows how N-day is normally determined. Assets are allocated so that the sum of assets plus production is a constant fraction of requirements and there are no more assets when the production rate equals this allocation rate. (Cross-hatching is used to show that assets fill the area above the production response.) N-day is the day inventory is exhausted, when allocation is made by the above rule. If requirements are met without using up all of the assets, N-day and P-day are the same and RR = 1.0. For RR less than 1.0, the production rate continues to increase after N-day and from N-day to P-day the ratio of the amount of ammunition available to the amount required is larger than the readiness ratio. After N-day, requirements are satisfied only from post D-day production. If the period selected for the analysis is too short to permit the production rate to equal the allocation rate at N-day, assets are to be used up in the period and N-day is the end of the period.

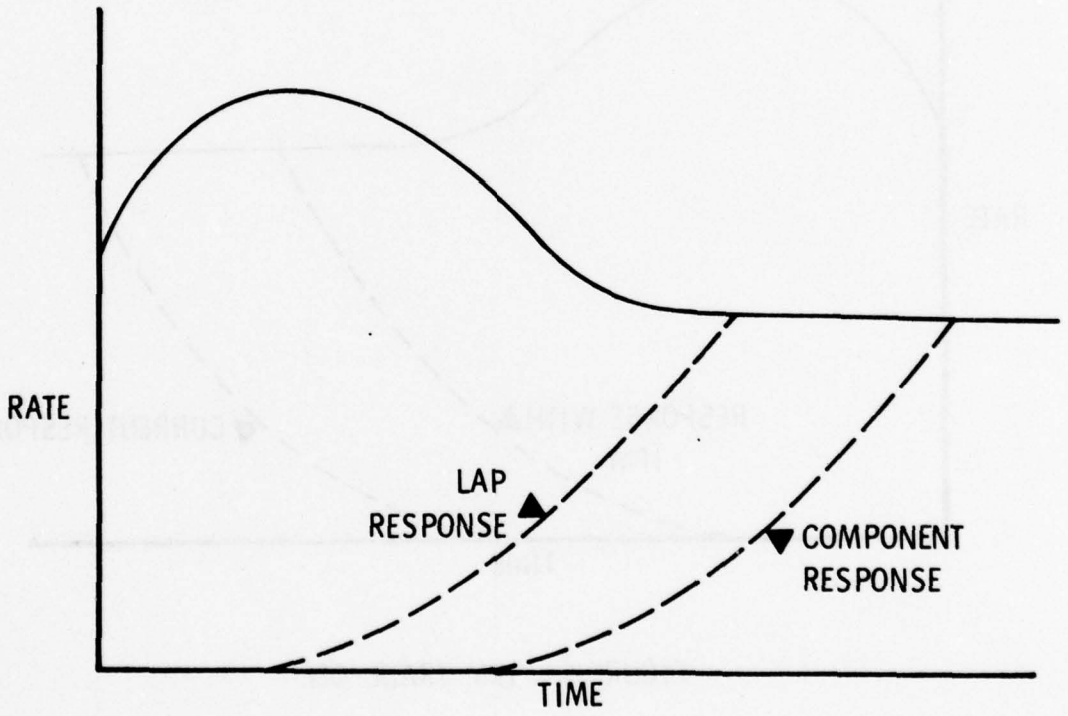


FIGURE 4. COMPONENT TRADE-OFF

(6) The selection of component buy or end item buy depends on the relationship of the component production to the end item production (Load, Assembly, and Pack or "LAP") response. If, as shown in Figure 4, a component production rate occurs later than the end item LAP rate, requirements represented by the area between the two curves can be satisfied by stockpiling components. These stockpiled components would then be available to supply maximum LAP capabilities. For one end item, more than one component may need to be stockpiled. Lower costs associated with component storage often make component stockpiling cheaper than end item stockpiling. The model will select the lowest cost alternative.

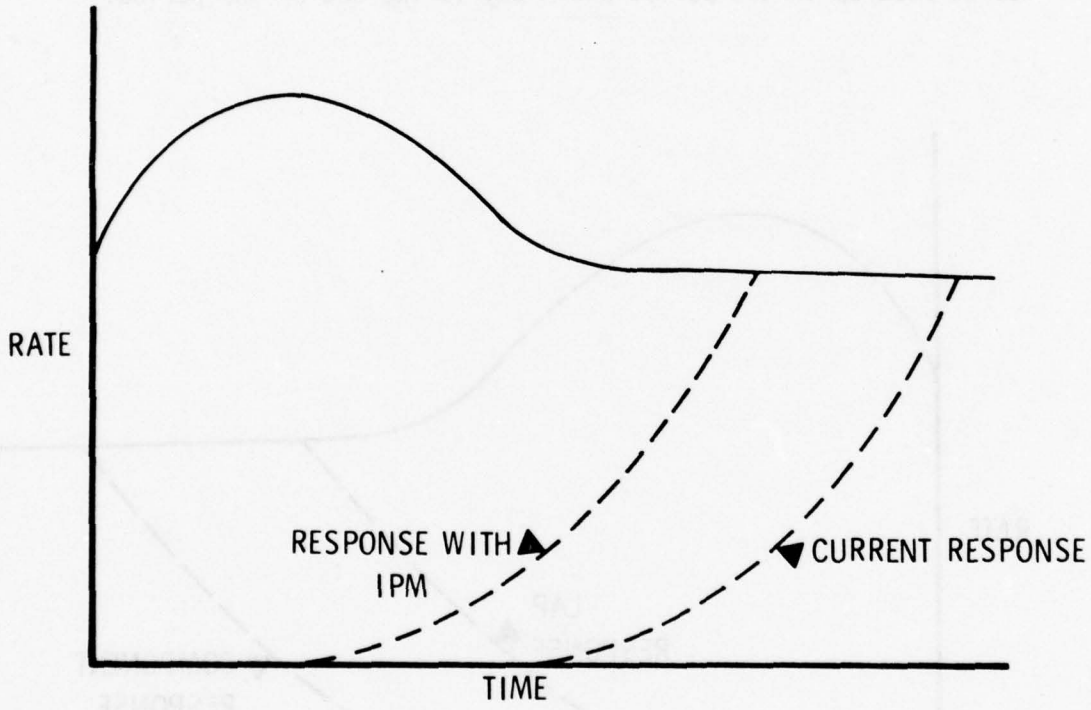


FIGURE 5. IPM TRADE-OFF

(7) An Industrial Preparedness Measure (IPM) is any action taken prior to D-day to improve post D-day response. If the time improvement resulting from the implementation of an IPM is as shown in Figure 5, then the area between the two response curves is equivalent to a specific amount of inventory. The IA/PT model compares the costs of the IPMs to the cost of this inventory and selects the cheapest.

(8) Depending on the objective of the user, the IA/PT model selects either the least-cost solution or the maximum readiness solution. In either case, the solution specifies the quantities of components and end items to be procured and the IPM's to be implemented.

CHAPTER II

MODEL

2-1. INTRODUCTION

The Item Acquisition/Production Trade-Off Model (IA/PT) is used for a single end item and its components. For any specific readiness ratio (RR), it yields the total minimum peacetime costs (or for any dollar budget, it maximizes the readiness ratio) by specifying quantities of end items and components to be stockpiled for use following mobilization, and selects IPM's for selected plants and production lines.

2-2. DEFINITION OF VARIABLES

a. Model decision variables used are:

$X_{p,g,i,j,k}$	amount of p production on line g with alternative i in month j for use in month k
$Y_{p,-1,k}$	amount of p to be produced in FYDP to be used in month k
$Y_{p,-2,k}$	amount of p available at beginning of planning period (T-day) to be used in month k
$I_{p,g,i}$	1 if p is produced on line g using alternative i, 0 otherwise
R	readiness ratio (RR)

b. The coefficients and constraints used by the model are:

c_p	average unit cost to produce product p
E	limit on available dollars
$r_{p,g,i}$	cost for alternative i on line g to produce p
D_k	requirement for end item in month k

Q_p	limit on procurement of p
P	peacetime requirement (5 year)
T_p	inventory of p on hand at T-day
$b_{p,g,i,j}$	during production buildup, the fraction of the eventual maximum capacity to produce p on line g with alternative i in month j
$W_{p,g,i}$	monthly capacity to produce p on line g using alternative i
U_p	total storage capacity for p (including the quantity of p already stored)
V	pipeline requirement (materiel in transit)

c. Subscripts and terminology used are:

p	product may be a component or an end item, p=1 represents the end item
p*	end item or component that p supports
g	line
i	identifier for IPM alternatives
j	month of production
k	month of use
s	number of months from D-day to P-day
T-day	termination of current planning period (start of FYDP)
R-day	end of FYDP
D-day	declaration of war or mobilization
N-day	day assets are exhausted when allocated at constant readiness
P-day	day when production meets requirements

2-3. TECHNIQUES USED

The model utilized mixed integer programming optimization. Implementation of the current version is through the IBM MPSX software package.

2-4. COSTS CONSIDERED

Only two types of costs are considered. The first cost is for inventory acquisition of end items and components. The second cost is to implement each IPM. All costs are current year dollars.

2-5. OPTIMIZATION

The model can be run to obtain the lowest cost plan for some specified readiness "Mode 1" or obtain the highest readiness plan within some fixed budget "Mode 2".

a. For the Mode 1 case, the objective is to minimize total cost.

$$\text{Minimize} \quad \left(\sum_p \sum_k c_p \cdot Y_{p,-1,k} + \sum_p \sum_g \sum_i r_{p,g,i} \cdot I_{p,g,i}\right) \qquad (1)$$

This represents minimizing the sum of total component and end item production cost in the FYDP (with peacetime costing, production after D-day is "free") and the cost for those IPMs selected. Applicable constraints are discussed in paragraph 2-6.

b. For the Mode 2 case, the objective is to maximize readiness:

$$\text{Maximize } R \qquad (2)$$

subject to a funding constraint in addition to the constraints discussed in paragraph 2-6. This funding constraint is a limitation on production and IPM set-up costs in the same form as in 2-5a.

$$\sum_p \sum_k c_p \cdot Y_{p,-1,k} + \sum_p \sum_g \sum_i r_{p,g,i} \cdot I_{p,g,i} \leq E \qquad (3)$$

2-6. CONSTRAINTS

The following discusses constraints which apply for both of the optimizations above.

a. For each month, demand for the end item can be satisfied from three sources: inventory on hand, inventory to be purchased, and/or production before or during the month.

$$Y_{1,-2,k} + Y_{1,-1,k} + \sum_g \sum_i \sum_{j \le k} X_{1,g,i,j,k} \geq D_k \cdot R \qquad (4)$$

For $k = 1, 2, \ldots, s$

b. For components, demand in a given month (k_o) is the total monthly production of the end item or component supported. There is a minimum 30-day shipping period provided between component production and utilization; component production must occur by month $k_o - 1$.

$$Y_{p,-2,k_o} + Y_{p,-1,k_o} + \sum_g \sum_i \sum_{j \le k_o - 1} X_{p,g,i,j,k_o-1} \geq \qquad (5)$$

$$\sum_g \sum_i \sum_{k \ge k_o} X_{p^*,g,i,k_o,k} \qquad \forall\, p>1, k_o = 1,2,\ldots,s$$

The left-hand-side (l.h.s.) has the same form as in the constraint in paragraph 2-6a; the right-hand-side (r.h.s.) represents production of the product support in month k_o for use in k_o or after.

c. After P-day, demand at the level described by the RR must be satisfied by production.

$$\sum_g \sum_i X_{1,g,i,s+1,s+1} \geq D_{s+1} \cdot R \qquad (6)$$

$$\sum_g \sum_i X_{p,g,i,r,r} \geq \sum_g \sum_i X_{p^*,g,i,s+1,s+1} \qquad \forall\, p>1, r=s,s+1 \qquad (7)$$

For relation (6), the l.h.s. is total end item production on the month following P-day; the r.h.s. shows the fraction of requirements which must be supported. For relation (7), the l.h.s. is the component production in the month in which P-day occurs and the following month; the r.h.s. shows the production of components supported.

d. The allocation of inventory on hand, by month, must be consistent with total T-day inventory.

$$\sum_k Y_{p,-2,k} \leq T_p \qquad \forall p \qquad (8)$$

e. To limit procurement, the total FYDP period production is constrained after the RR = 1.0 run (Constraint can be eliminated if not applicable).

$$\sum_k Y_{p,-1,k} \leq Q_p \qquad \forall p \tag{9}$$

f. For any month, production can be used in any later month for each line (the l.h.s.) and cannot exceed its buildup capability (the r.h.s.).

$$\sum_{k \geq j} X_{p,g,i,j,k} \leq b_{p,g,i,j} \cdot W_{p,g,i} \cdot I_{p,g,i} \qquad \forall p,g,i,j \tag{10}$$

g. Insure that no unnecessary lines are opened to satisfy post P-day requirements (constraints can be eliminated if not applicable and may be overridden by redefining the right hand side to a fixed value).

$$\sum_g \sum_i W_{p,g,i} \cdot I_{p,g,i} \leq D_{s+1} + \min_{g,i} W_{p,g,i} \qquad \forall p \tag{11}$$

The l.h.s. shows the monthly capacity (not buildup) for a product; the r.h.s. is the demand at P+1 month plus the smallest capacity line producing the product.

h. At most, one alternative may be selected for each line, the IPMs are mutually exclusive, only one can be selected.

$$\sum_i I_{p,g,i} \leq 1 \qquad \forall p,g \tag{12}$$

i. The total inventory for a product (the l.h.s.) is constrained by storage capacity (the r.h.s.) (constraint can be eliminated if not applicable).

$$\sum_k Y_{p,-1,k} + \sum_k Y_{p,-2,k} \leq U_p \qquad \forall p \tag{13}$$

j. The materiel in transit (l.h.s.) is filled at the same level as the readiness ratio (r.h.s.).

$$Y_{1,-1,0} + Y_{1,-2,0} \geq V \cdot R \tag{14}$$

CHAPTER III

OPERATION

3-1. INTRODUCTION

This chapter describes operation of the Mathematical Programming System, Extended (MPSX) version of the model on the IBM 360/65. Computer input must be prepared as described in the IA/PT Users' Manual.

3-2. METHOD OF OPERATION

Four sets of Job Control Language (JCL), two FORTRAN Programs, and three MPSX Programs are required to run the model. In addition, a COBOL Report Generator is available for use in interpreting the output. The listings for these programs are presented in Appendix A. Operation of the model is a three-step process: generate the matrix, solve the mixed integer program, and write a report.

a. MPSX program A is run with the first set of JCL. The input information (as described in the Users' Manual) is processed using a FORTRAN matrix generator called from MPSX program A. A matrix in MPSX format is generated and then processed. This program can terminate (the second FORTRAN program controls this) with one of three conditions: (1) optimal solution found, (2) the specified maximum time is exceeded with the optimal continuous solution not found, or (3) the specified maximum time is exceeded with the optimal continuous solution found but the optimal mixed integer solution is not. The status of the solution is saved on tape using the problem file.

b. MPSX program B and the second set of JCL are used to resume solution if MPSX program A or B terminates before reaching an optimal continuous solution. Using the problem file the program restores the problem and continues processing until one of the three conditions described in paragraph 3-2a is again reached.

c. MPSX Program C and the third set of JCL are used to resume solution, if MPSX Program A, B, or C terminate because the maximum time is exceeded after the optimal continuous solution is found but before the optimal solution is found. Using the problem file, the program restores the problem and then proceeds to termination under one of two conditions: Optimal solution or time exceeded.

d. Once the optimal solution is found, management reports can be generated using the fourth set of JCL, MPSX program C and the COBOL report generator.

3-3. METHOD FOR OBTAINING COST-READINESS POINTS

Running the model is portrayed in Figure 6 and is described below.

a. The first phase consists of two runs. One run is done with Mode 1 with RR fixed at 1.0. This run determines the cost as well as quantities of end item and components to be procured and IPM's to be implemented to meet requirements. The other run is done in Mode 2 with the budget fixed at $0.0. This gives the current readiness (C.R.). The use of more than two significant figures for the readiness ratio is normally not meaningful in planning.

b. The second phase consists of up to three runs for intermediate points (more runs may be done if time permits or conditions require). Suggested intermediate readiness ratios to be run are given by the relation CR + (1-CR) · X, where X takes on the values 0.1, 0.3, and 0.6. For CR above 0.7, only the last two values should be used. The uneven spacing is suggested to provide more points where the cost-readiness relation is changing most rapidly. For intermediate readiness runs, include only those IPM's selected in the first phase. Procurement (Q_p) of the end item and components should be constrained by the amounts obtained for RR = 1.0 to make planning consistent with reaching 100% readiness.

c. If it is desired to obtain readiness versus cost information with the restriction that component procurement is not permitted, then the runs in the first phase must have this restriction. Intermediate runs with this restriction are then run.

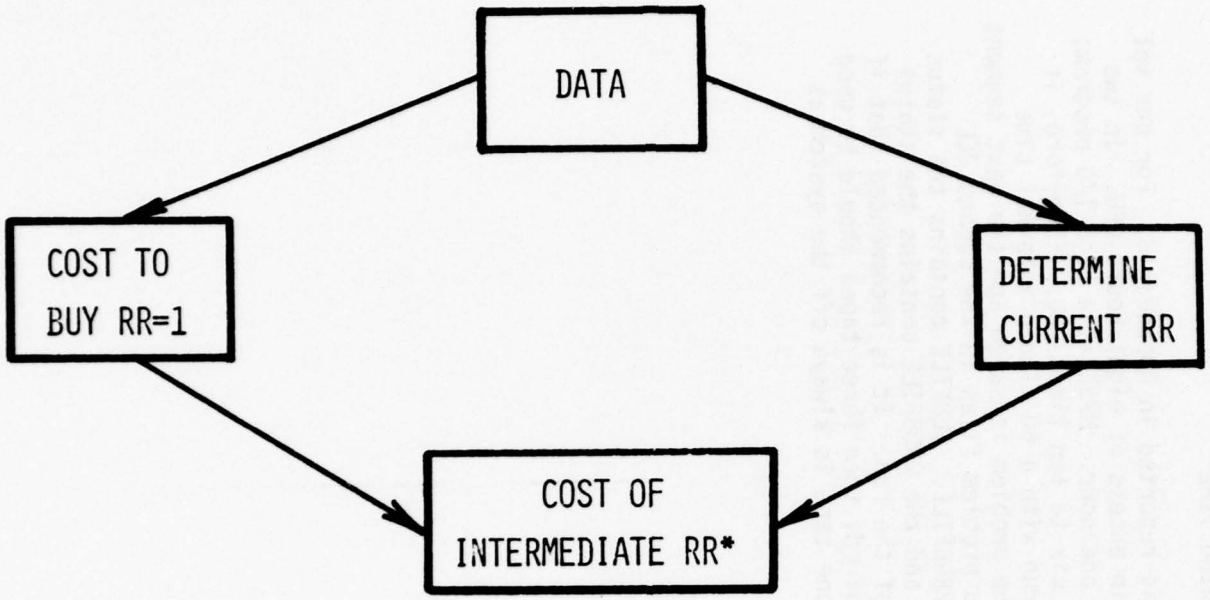
3-4. ADDITIONAL INPUT CONTROLS

In addition to the input described in the Users' Manual, the analyst may use card B to specify run parameters. The card columns, format, and description are as follows:

a. Column 50 (I1) - The mode (1 or 2); default is 1.

b. Columns 51-55 (F5.3) - The readiness ratio; default is 1.0.

c. Column 56 (I1) - A scaling factor expressed as an exponent of 10. For example, 3 means all quantities will be converted to thousands; default is 0 (units).



* As many as required

FIGURE 6 COMPUTER RUNS FOR IA/PT ANALYSIS

d. Columns 57-58 (I2) - Number of periods after D-day; default is 24.

e. Columns 59-68 (F10.0) - Limit of post-P day production permitted (Q_D); zero means limit to be calculated, negative means no limit; default is -1.

3-5. PROBLEM SIZE AND RUNNING TIME

Experience with the model is reported in Appendix B. For one set of data, solution time is in excess of eight hours CPU. In two others, it is in excess of one hour. MPSX is a high I/O program; execution time is normally six to ten times CPU. Therefore, it has been found prudent to run with a 60 minute internal time control after which time the problem is saved and the next segment can execute. There are two problem files in the second JCL identified as OLDFILE and PROBFILE. OLDFILE contains the status of problem used to restart and the PROBFILE contains the status of the problem at the end of the run. It is recommended that if three or more runs are required, then three tapes should be used in rotation. In that way one tape is always off the system as backup.

APPENDIX A

SOURCE LISTINGS AND JOB CONTROL LANGUAGE

APPENDIX A

1. THIS APPENDIX CONTAINS LISTINGS FOR ALL PROGRAMS USED BY THE IA/PT MODEL. THE FOLLOWING SOFTWARE IS REQUIRED: FORTRAN, COBOL, AND MPSX.

2. A FORTRAN SUBROUTINE IS USED TO GENERATE THE MATRIX IN MPSX FORMAT. IT OPERATES ON THE INPUT DATA AS PRESCRIBED IN THE USER'S MANUAL. THIS SUBROUTINE IS CALLED BY MPSX PROGRAM A (PARAGRAPH 3 BELOW). THE PROGRAM LISTING FOLLOWS:


```
C                                                                       00000001
C     THIS PROGRAM GENERATES AN INPUT LP MATRIX IN MPSX FORMAT FOR THE  00000100
C     JCAP IA/PT MODEL.  THE MODEL CONSIDERS ONE END ITEM AND MAJOR     00000200
C     COMPONENTS.  IT PERMITS CALCULATION OF TRADEOFF OF PRODUCTION WITH00000300
C     ITEM AND COMPONENT INVENTORY.                                     00000400
C                                                                       00000500
C     THE PROGRAM IS DIVIDED INTO THE FOLLOWING SECTIONS                00000600
C                                                                       00000700
C     1 THE INPUT IS READ IN AND SOME LIMITED CHECKING PERFORMED        00000800
C     2 THE ROWS SECTION IS WRITTEN                                     00000900
C     3 THE COLUMNS SECTION IS WRITTEN                                  00001000
C     4 THE RIGHT HAND SIDE SECTION IS WRITTEN                          00001100
C     5 THE BOUNDS SECTION IS WRITTEN                                   00001200
```

```
C                                                                       00001300
C     DEFINITIONS                                                       00001400
C                                                                       00001500
C     ALFAC - ALLOCATION FACTOR                                         00001530
C     ALLLIN - LINE ALLOCATION FACTOR                                   00001540
C     ALLOW - INITIAL ALLOWANCES                                        00001550
C     ALOS - LOSSES                                                     00001570
C     AP - NUMBER OF PRODUCTS                                           00001600
C     BCAP - CAPACITY OF PLANT AS ADJUSTED BY BUILDUP FRACTION          00001700
C     BL - CHARACTER BLANK                                              00001800
C     C - CHARACTER C                                                   00001850
C     CAP - LINE CAPACITY IN END ITEM EQUIVILENTS/MONTH                 00001900
C     CAPN - NEGATIVE OF CAPLIM                                         00002000
C     CARY - VALUE OF SMALLEST CAPACITY FOR PRODUCT                     00002100
C     COST - ROLLED UP COST                                             00002200
C     COSTE - COST/UNIT OF APPLIED IPM                                  00002300
C     COSTI - COST/UNIT OF PRODUCT                                      00002400
C     DATE - T-DAY                                                      00002450
C     DISC - DISCOUNT RATE                                              00002500
C     DOLLAR - AMOUNT OF DOLLARS AVAILABLE                              00002600
C     EQ8LM - RHS FOR EQUATION 6                                        00002700
C     FAC - REQUIREMENTS LEVEL TO BE ACHIEVED (READINESS RATIO)         00002800
C     FICR - FIXED COST AT R-DAY                                        00002900
C     FICT - FIXED COST AT T-DAY                                        00003000
C     HA - ALTERNATIVE CODE                                             00003100
C     HEAD - HEADING TO BE PRINTED BEFORE SOLUTION                      00003150
C     HL - LINE CODE                                                    00003200
C     HLC - SPACE FOR HOLDING LINE CODE FOR CHECKING                    00003300
C     HOL - INPUT CARD TYPE                                             00003350
C     HOLS - INPUT CARD SUBTYPE                                         00003360
C     HP - PRODUCT CODE                                                 00003400
C     HPC - SPACE FOR HOLDING PRODUCT CODE FOR CHECKING                 00003500
C     HS - PRODUCT SUPPORTED CODE                                       00003600
C     I - INDEX, USED FOR READING IN AND THEN AS TOTAL ALTERNATIVES     00003700
C     IEXP - EXPONENT OF 10, MULTIPLICATIVE FACTOR FOR NOS READ IN      00003800
C     IPMDES - IPM DESCRIPTION                                          00003850
```

```
C     ISC - SCALING FOR SOLUTION (3 FOR THOUSANDS)                          00003900
C     ITD - TIME FROM T-DAY TO D-DAY                                        00004000
C     ITR - TIME FROM T-DAY TO R-DAY                                        00004100
C     J - INDEX, USED MOSTLY AS SPECIFIC ALTERNATIVE                        00004200
C     JI - INDEX                                                            00004300
C     J10 - INDEX FOR MARKER CARDS                                          00004400
C     J11 - LINE NUMBER + 10                                                00004500
C     K - INDEX                                                             00004600
C     KF - LAST INDEX ON READ                                               00004700
C     KI - INDEX                                                            00004800
C     KJ - INDEX                                                            00004900
C     KS - FIRST INDEX ON READ                                              00005000
C     L - INDEX                                                             00005100
C     LI - INDEX                                                            00005200
C     M - INDEX                                                             00005300
C     MF - NO OF MONTHS IN D TO P PERIOD                                    00005400
C     MFA - INDEX                                                           00005500
C     MFI - INDEX BOUND                                                     00005600
C     MFL - INDEX BOUND                                                     00005700
C     MF1 - MF INCREASED BY 10                                              00005800
C     MF2 - MF INCREASED BY 10                                              00005900
C     SUSRAT - MINIMUM SUSTAINING RATE                                      00005950
C     MODE - OPERATION MODE (1 IF MINIMIZING DOLLARS, 2 IF MAXIMIZING       00006000
C          RR)                                                              00006100
C     MPF - LAST TO READ IN                                                 00006200
C     MPI - FIRST TO READ IN                                                00006300
C     MPP - MONTH FULL PRODUCTION REACHED FOR INDIVIDUAL LINE               00006400
C     MPS - MONTH PRODUCTION STARTS FOR INDIVIDUAL LINE                     00006500
C     MSI - INDEX BOUND                                                     00006600
C     MZ - INDEX                                                            00006700
C     NA - NUMBER OF ALTERNATIVES                                           00006750
C     NI - INDEX                                                            00006800
C     NP - NO OF PRODUCTS                                                   00006900
C     NUM - CARD SEQUENCE NUMBER                                            00006950
C     PEREQ - PEACETIME REQUIREMENTS                                        00006980
C     PIPEM - PIPE TIMES FAC                                                00007000
```

```
C      PIPE - PIPELINE REQUIREMENT                                          00007100
C      PIPEN - NEGATIVE OF PIPE                                             00007200
C      PLNAME - PRODUCER/LINE NAME                                          00007230
C      PNAME - PRODUCT NAME                                                 00007250
C      PREQ - VALUE OF REQUIREMENTS TIMES READINESS RATIO                   00007300
C      PRFAC - PROCUREMENT FACTOR                                           00007350
C      PROCLM  - LIMIT OF PROCUREMENT (IN END ITEM EQUIVALENT)              00007400
C      REQ - MONTHLY REQUIREMENTS                                           00007500
C      RLEV - BUILDUP FACTOR                                                00007600
C      RINVOH - ASSETS OF PRODUCT                                           00007700
C      STORCP  - STORAGE CAPACITY                                           00007800
C      TEMP - TEMPERATURE OF ALTERNATIVE, W IF WARM, C IF COLD              00007840
C      TITLE - TITLE OF STUDY                                               00007850
C      TITID - SHORT TITLE                                                  00007860
C      YRCRD - YEARLY COSTS R TO D DAY                                      00007900
C      YRCTD - YEARLY COSTS T TO D DAY                                      00008000
C      Z - LOCATION WHERE Z STORED, USED AS END OF FILE TEST                00008100
C                                                                           00008200
C      INITIALIZE                                                           00008300
C                                                                           00008400
       DOUBLE PRECISION HEAD                                                00008450
       DIMENSION AP(99), COSTI(99), RINVOH(99), CAP(99), FICT(99), FICR(900008500
      19), YRCTD(99), YRCRD(99), RLEV(99,32), HP(99), HL(99), HA(99), HS(00008600
      299), MPS(99), MPP(99),  COSTE(99), REQ(99), STORCP(99),             00008700
      3 PROCLM(99), TITLE(17), TITLID (2), DATE(2), HOLS(3), PNAME(99,5),00008800
      4 ALFAC(99), PRFAC(99), PLNAME(99,5), SUSRAT(99), ALLLIN(99),        00008810
      5 TEMP(99), IPMDES(99,11)                                            00008820
       DATA Z/1HZ/, BL/1H /, C/1HC/                                        00008900
       J10=100                                                             00009000
C                                                                          00009100
C      READING INPUT.                                                      00009200
C                                                                          00009300
C                                                                          00009400
C      CARD A - TITLE OF STUDY                                             00009500
C                                                                          00009600
       READ(5,400) HOL, TITLE, NUM                                         00009700
```

```
      WRITE(6,401) NUM, HOL, TITLE                                          00009800
C                                                                           00009900
C     CARD B - CONTROL CARD                                                 00009950
C                                                                           00010100
      READ(5,405) HOL, TITLID, NP, NA, DATE, DOLLAR,MODE, FAC, ISC, MF      00010200
     1 , EQ8LM, NUM                                                         00010205
      IF (MODE.EQ.0) MODE=1                                                 00010210
      IF(FAC.EQ.0.) FAC=1.                                                  00010220
      IF(MF .EQ. 0) MF=24                                                   00010230
CC    IF (MF.LT.24) MF=MF+1                                                 00010240
      IF (EQ8LM .LE. 0.) EQ8LM =-1.                                         00010250
      WRITE(6,406) NUM, HOL, TITLID, NP, NA, DATE, DOLLAR                   00010300
      ITD=5                                                                 00010310
      ITR=5                                                                 00010320
C                                                                           00010400
C     CARD C - ASSETS MODIFICATION CARD                                     00010500
C                                                                           00010600
      READ(5,410) HOL, ALLOW, PEREQ, ALOS, PIPE, NUM                        00010700
      WRITE(6,411) NUM, HOL, ALLOW, PEREQ, ALOS, PIPE                       00010800
C                                                                           00010900
C     CARD D - REQUIREMENTS                                                 00011100
C                                                                           00011200
      WRITE(6,414)                                                          00011250
      DO 10 I=1,3                                                           00011300
      READ(5,415) HOL, HOLS(I), (REQ((I-1)*8+J),J=1,8), IEXP, NUM           00011400
      JI = (I-1)*8 + 1                                                      00011500
      JF = JI + 7                                                           00011600
      DO 11 J=JI,JF                                                         00011700
   11 REQ(J) = REQ(J) * 10.** IEXP                                          00011800
   10 WRITE(6,416) NUM, HOL, HOLS(I), (REQ((I-1)*8+J),J=1,8)                00011900
      DO 12 I=25,99                                                         00011910
12    REQ(I) = REQ(24)                                                      00011920
C                                                                           00012000
C     CARD E - PRODUCT INFORMATION                                          00012100
C                                                                           00012200
      DO 20 I=1,NP                                                          00012300
```

```
      READ(5,420) HOL, (PNAME(I,J),J=1,5), AP(I), RINVOH(I), COSTI(I),   00012400
     1 ALFAC(I), PRFAC(I), STORCP(I), NUM                                00012500
      IF(STORCP(I) .LE. 0.) STORCP(I) = -1.                              00012550
      IF(STORCP(I) .EQ. 1.) STORCP(I) = 0.                               00012555
      PROCLM(I) = -1.                                                    00012560
      IF(ALFAC(I) .EQ. 0.) ALFAC(I) = 1.                                 00012570
      IF(PRFAC(I) .EQ. 0.) PRFAC(I)=1.                                   00012580
      COSTI(I) = COSTI(I) * PRFAC(I)                                     00012590
   20 WRITE(6,421) NUM, HOL, (PNAME(I,J),J=1,5), AP(I), RINVOH(I),       00012600
     1 COSTI(I), ALFAC(I), PRFAC(I)                                      00012700
C                                                                        00012800
C     PRODUCTION LINE ALTERNATIVES                                       00012900
C                                                                        00013000
      I=0                                                                00013050
29    CONTINUE                                                           00013100
      I=I+1                                                              00013150
C                                                                        00013200
C     CARD F - PRODUCTION LINE DATA                                      00013300
C                                                                        00013400
      READ(5,425,END=35)                                                 00013500
     1  HOL, (PLNAME(I,J),J=1,5), HP(I), HL(I), HA(I), HS(I),            00013550
     1 CAP(I), SUSRAT(I), ALLLIN(I), TEMP(I), NUM                        00013600
      IF (ALLLIN(I).EQ.0.) ALLLIN(I) = 1.                                00013610
      DO 21 L=1,NP                                                       00013612
      IF (HP(I) .EQ. AP(L)) GO TO 22                                     00013614
21    CONTINUE                                                           00013616
22    CONTINUE                                                           00013618
      CAP(I) = CAP(I) * ALLLIN(I)                                        00013620
     1 *ALFAC(L)/PRFAC(L)                                                00013621
      SUSRAT(I) = SUSRAT(I) * ALLLIN(I)                                  00013630
     1 *ALFAC(L)/PRFAC(L)                                                00013631
      IF (TEMP(I) .EQ. BL) TEMP(I) = C                                   00013640
      WRITE(6,426) NUM, HOL, (PLNAME(I,J),J=1,5), HP(I), HL(I), HA(I),   00013700
     1 HS(I), CAP(I), SUSRAT(I),            TEMP(I)                     00013800
C                                                                        00013900
C     CARD G - SUPPLEMNTAL INFORMATION                                   00014000
```

```
C                                                                       00014100
      READ(5,430) HOL, FICT(I), YRCTD(I), (IPMDES(I,J),J=1,11), NUM     00014200
      WRITE(6,431) NUM, HOL, FICT(I), YRCTD(I), (IPMDES(I,J),J=1,11)    00014300
      FICR(I)=0.                                                        00014310
      YRCRD(I)=0.                                                       00014320
C                                                                       00014400
C     CARD H - BUILDUP CAPABILITY                                       00014500
C                                                                       00014600
      WRITE(6,434)                                                      00014700
      RLEV(I,1) = 0.                                                    00014730
      RLEV(I,2) = 0.                                                    00014760
      DO 30 J=1,3                                                       00014800
      READ (5,435) HOL, HOLS(I), (RLEV(I,(J-1)*8+K),K=3,10),IEXP,NUM    00014900
      KI=(J-1)*8+1                                                      00015000
      KF=KI+7                                                           00015100
      DO 31 K=KI,KF                                                     00015200
   31 RLEV(I,K) =(RLEV(I,K)*10.**IEXP)*ALLLIN(I)                        00015300
     1 *ALFAC(L)/PRFAC(L)                                               00015301
30    WRITE (6,436) NUM, HOL, HOLS(I), (RLEV(I,(J-1)*8+K),K=1,8)        00015350
      GO TO 29                                                          00015360
35    I=I-1                                                             00015370
      IF (I.NE.NA) WRITE(6,437) I, NA                                   00015380
      NA=I                                                              00015400
      DO 33 I=1,NA                                                      00015410
      DO 32 J=1,24                                                      00015420
      IF (RLEV(I,J) .NE. 0.) GO TO 33                                   00015430
32    CONTINUE                                                          00015435
33    MPS(I) = J                                                        00015440
      DO 34 I=1,NA                                                      00015450
      DO 34 J=25,32                                                     00015460
34    RLEV(I,J)=RLEV(I,24)                                              00015470
C                                                                       00016900
C     WRITING ROWS PORTION OF MATRIX                                    00017000
C                                                                       00017100
  140 I=NA                                                              00017200
C                                                                       00017300
```

```
C       WRITE NAME AND ROWS CARDS AND OBJECTIVE FUNCTION                00017400
C       AND CONSTRAINT C-3                                              00017500
C                                                                       00017600
        IF (MODE.EQ.1) WRITE (8,460)                                    00017700
        IF (MODE.EQ.2) WRITE (8,470)                                    00017800
C                                                                       00017900
C       WRITE ROWS FOR CONSTRAINTS C-4 TO C-7                           00018000
C                                                                       00018100
        MFI=MF+1                                                        00018200
        DO 150 J=1,NP                                                   00018300
        DO 150 K=1,MFI                                                  00018400
        KI=K+10                                                         00018500
        WRITE (8,480) KI, AP(J)                                         00018600
        IF  (( J.EQ. 1) .OR. (K. LT. MFI))  GO TO 150                   00018610
        KI = MFI + 11                                                   00018620
        WRITE (8,480) KI, AP(J)                                         00018630
150     CONTINUE                                                        00018640
C                                                                       00018700
C       WRITE ROWS FOR CONSTRAINT C-8                                   00018800
C                                                                       00018900
        DO 160 J=1,NP                                                   00019000
160     WRITE (8,490) AP(J)                                             00019100
C                                                                       00019110
C       WRITE ROWS TO SUM BUY                                           00019120
C                                                                       00019140
        DO 161 J=1,NP                                                   00019150
161     WRITE(8,491) AP(J)                                              00019160
C                                                                       00019200
C       WRITE ROWS FOR CONSTRAINT C-9                                   00019300
C                                                                       00019400
C                                                                       00019500
        DO 165 J=1,NP                                                   00019600
        IF (PROCLM(J) .LT. 0) GO TO 165                                 00019700
        WRITE(8,495) AP(J)                                              00019800
165     CONTINUE                                                        00019900
C                                                                       00020000
```

```
C                                                                       00020100
C     WRITE ROWS FOR CONSTRAINT C-10                                    00020200
C                                                                       00020300
      DO 180 J=1,I                                                      00020400
      KS=MPS(J)                                                         00020500
         IF (KS.GT.MF) KS=MF                                            00020550
      MFA=MF+1                                                          00020600
      DO 170 K=KS,MFA                                                   00020800
      KI=K+10                                                           00020900
170   WRITE (8,500) HP(J),HL(J),HA(J),KI                                00021000
180   CONTINUE                                                          00021100
C                                                                       00021200
C     WRITE ROWS FOR CONSTRAINT C-11                                    00021300
C                                                                       00021400
      DO 190 J=1,NP                                                     00021500
      IF(EQ8LM .LT. 0.) GO TO 190                                       00021600
      WRITE (8,510) AP(J)                                               00021700
190   CONTINUE                                                          00021800
C                                                                       00021900
C     WRITE ROWS FOR CONSTRAINT C-12                                    00022000
C                                                                       00022100
      HPC=Z                                                             00022200
      HLC=Z                                                             00022300
      DO 210 J=1,I                                                      00022400
      IF ((HP(J).EQ.HPC).AND.(HL(J).EQ.HLC)) GO TO 200                  00022500
      WRITE (8,520) HP(J),HL(J)                                         00022600
200   HPC=HP(J)                                                         00022700
210   HLC=HL(J)                                                         00022800
C                                                                       00022900
C     WRITE ROWS FOR CONSTRAINT C-13                                    00023000
C                                                                       00023100
      DO 213 J=1,NP                                                     00023200
       IF(STORCP(J) .LT. 0.) GO TO 213                                  00023300
      WRITE(8,525) AP(J)                                                00023400
213   CONTINUE                                                          00023500
C                                                                       00023600
```

```
C       WRITE ROWS FOR CONSTRAINT C-14                                  00023700
C                                                                       00023800
        WRITE(8,527) AP(1)                                              00023900
C                                                                       00024000
C       WRITING COLUMNS PORTION OF MATRIX                               00024100
C                                                                       00024200
C       WRITE HEADER                                                    00024300
C                                                                       00024400
        WRITE (8,530)                                                   00024500
C                                                                       00024600
C       WRITE READINESS RATIO VARIABLE (RR)                             00024700
C                                                                       00024800
        WRITE(8,533)                                                    00025000
        MF3 = MF+1                                                      00025100
        DO 215 J=1,MF3                                                  00025200
        J11= J + 10                                                     00025300
        PREQ = -REQ(J)                                                  00025400
215     WRITE(8,536) J11, AP(1), PREQ                                   00025500
        PIPEM = -PIPE                                                   00025600
        WRITE(8,537) AP(1), PIPEM                                       00025900
C                                                                       00026000
C       WRITE CURRENT INVENTORY VARIABLES (Y-2)                         00026100
C                                                                       00026200
216     CONTINUE                                                        00026300
        WRITE(8,538) AP(1), AP(1), AP(1)                                00026400
        MF1 = MF + 1                                                    00026450
        DO 221 J=1,NP                                                   00026500
        DO 221 K=1,MF1                                                  00026600
        IF (J.EQ.1 .AND. K.EQ.MF1) GO TO 221                            00026650
        KI=K+10                                                         00026700
        IF(STORCP(J) .LT. 0.) GO TO 220                                 00026800
        WRITE(8,539) AP(J), KI, AP(J)                                   00026900
220     WRITE (8,540) AP(J),KI,KI,AP(J),AP(J)                           00027000
C                                                                       00027100
221     CONTINUE                                                        00027150
C       WRITE INVENTORY TO PROCURE VARIABLES (Y-1)                      00027200
```

```
C                                                                       00027300
      WRITE (8,542) AP(1), COSTI(1), AP(1)                              00027400
     1 , AP(1), AP(1)                                                   00027401
      DO 231 J=1,NP                                                     00027500
      WRITE(8,545) AP(J), AP(J)                                         00027550
      DO 231 K=1,MF1                                                    00027600
      IF (J.EQ.1 .AND. K.EQ.MF1) GO TO 231                              00027650
      KI=K+10                                                           00027700
      WRITE(8,544) AP(J), KI, AP(J)                                     00027750
      IF(STORCP(J) .LT. 0.) GO TO 235                                   00027800
      WRITE(8,543) AP(J), KI, AP(J)                                     00027900
235   IF(PROCLM(J) .LT. 0.) GO TO 229                                   00028000
      WRITE(8,546) AP(J), KI, AP(J)                                     00028100
229   IF (MODE.EQ.2) GO TO 230                                          00028150
      IF (J.GT.1) WRITE(8,555) AP(J), KI, COSTI(J)                      00028200
230   WRITE (8,550) AP(J),KI,KI,AP(J),AP(J),KI,COSTI(J)                 00028300
231   CONTINUE                                                          00028350
C                                                                       00028400
C     WRITE X AND I VARIABLES                                           00028500
C                                                                       00028600
      DO 300 J=1,I                                                      00028700
C                                                                       00028800
C     WRITE PRODUCTION VARIABLES (X)                                    00028900
C                                                                       00029000
      MSI=MPS(J)                                                        00029100
      IF (MSI.GT.MF) MSI=MF                                             00029150
      DO 250 K=MSI,MF                                                   00029200
      KI=K+10                                                           00029300
      DO 250 L=K,MF                                                     00029400
      LI=L+10                                                           00029500
      LII=LI                                                            00029600
      IF (HS(J).NE.BL) LII=LI+1                                         00029700
      WRITE (8,560) HP(J),HL(J),HA(J),KI,LI,HP(J),HL(J),HA(J),KI,LII,HP(00029800
     1J)                                                                00029900
      HPC=Z                                                             00030000
      MF0=MF+10                                                         00030050
```

```
      DO 250 M=1,I                                                          00030100
      IF (HS(M).NE.HP(J)) GO TO 250                                         00030200
      IF (HP(M).EQ.HPC) GO TO 240                                           00030300
      WRITE (8,570) HP(J),HL(J),HA(J),KI,LI,KI,HP(M)                        00030400
      IF(KI.NE.MF0) GO TO 240                                               00030410
      MF1=MF0+1                                                             00030420
      WRITE(8,570) HP(J), HL(J), HA(J), KI, LI, MF1, HP(M)                  00030430
240   HPC=HP(M)                                                             00030500
250   CONTINUE                                                              00030600
      MF1=MF+11                                                             00030700
      IF (HS(J) .NE. BL) GO TO 261                                          00030800
      WRITE (8,580) HP(J),HL(J),HA(J),MF1,MF1,MF1,HP(J),HP(J),HL(J),HA(J00030900
     1),MF1                                                                 00031000
      HPC=Z                                                                 00031100
      MF3=MF1+1                                                             00031160
      DO 260 M=1,I                                                          00031200
      IF (HS(M).NE.HP(J)) GO TO 260                                         00031300
      IF (HP(M).EQ.HPC) GO TO 260                                           00031400
      WRITE (8,570) HP(J), HL(J), HA(J), MF1, MF1, MF3, HP(M)               00031520
      HPC=HP(M)                                                             00031600
260   CONTINUE                                                              00031700
      GO TO 270                                                             00031710
261   MF1 = MF + 12                                                         00031720
      MF2 = MF + 11                                                         00031725
      WRITE(8,580) HP(J), HL(J), HA(J), MF2, MF2, MF1, HP(J), HP(J),        00031730
     1 HL(J), HA(J), MF2                                                    00031740
      HPC=Z                                                                 00031745
      DO 265 M=1,I                                                          00031750
      IF (HS(M) .NE. HP(J)) GO TO 265                                       00031755
      IF (HP(M) .EQ. HPC) GO TO 264                                         00031760
      WRITE(8,570) HP(J), HL(J), HA(J), MF2, MF2, MF1, HP(M)                00031765
264   HPC=HP(M)                                                             00031770
265   CONTINUE                                                              00031780
C                                                                           00031800
C     WRITE CHOICE OF ALTERNATIVE VARIBLES (I) (INTEGER)                    00031900
C                                                                           00032000
```

```
270   CONTINUE                                                              00032100
      J10=J10+1                                                             00032200
      WRITE (8,590) J10                                                     00032300
      WRITE (8,600) HP(J),HL(J),HA(J),HP(J),HL(J)                           00032400
275   MSI=MPS(J)                                                            00032500
      IF (MSI.GT.MF) MSI=MF                                                 00032550
      DO 280 K=MSI,MF                                                       00032600
      KI=K+10                                                               00032700
      BCAP=-RLEV(J,K)                                                       00032800
280   WRITE (8,630) HP(J),HL(J),HA(J),HP(J),HL(J),HA(J),KI,BCAP             00033100
      MFA=MF+11                                                             00033300
      BCAP = -CAP(J)                                                        00033350
      WRITE (8,630) HP(J),HL(J),HA(J),HP(J),HL(J),HA(J),MFA,BCAP            00033400
C     REWRITE FOR DISCOUNTING                                               00033500
290   COST=FICT(J)+FICR(J)+YRCTD(J)*ITD+YRCRD(J)*(ITD-ITR)                  00033600
C                                                                           00033700
      COST=COST/10.**ISC                                                    00033800
      DO 298 L=1,NP                                                         00033810
      IF (HP(J) .EQ. AP(L)) GO TO 299                                       00033820
298   CONTINUE                                                              00033830
299   CONTINUE                                                              00033840
      COST = COST * ALLLIN(4) * ALFAC(L)/ PRFAC(L)                          00033850
      WRITE (8,635) HP(J),HL(J),HA(J),COST                                  00034100
      IF (EQ8LM .GE. 0.) WRITE (8,620) HP(J),HL(J),HA(J),HP(J),CAP(J)       00034200
      J10=J10+1                                                             00034300
300   WRITE (8,640) J10                                                     00034400
C                                                                           00034500
C     WRITING RHS PORTION OF MATRIX                                         00034600
C                                                                           00034700
C     WRITE HEADER                                                          00034800
C                                                                           00034900
      WRITE (8,650)                                                         00035000
C                                                                           00035100
C     WRITE RHS FOR CONSTRAINT C-3 (IF MODE 2)                              00035200
C                                                                           00035300
      IF (MODE .EQ. 2) WRITE(8,665) DOLLAR                                  00035400
```

```
C                                                                       00035500
C     WRITE RHS FOR RR (IF MODE 1)                                      00035600
C                                                                       00035700
      IF (MODE .EQ. 2) GO TO 315                                        00035800
      AFAC = -FAC                                                       00035900
      WRITE(8,660) AFAC                                                 00036000
C                                                                       00037400
C     WRITE RHS FOR CONSTRAINT C-8                                      00037500
C                                                                       00037600
315   CONTINUE                                                          00037700
      DO 320 J=1,NP                                                     00037800
320   WRITE (8,670) AP(J),RINVOH(J)                                     00038100
C                                                                       00038200
C     WRITE RHS FOR CONSTRAINT C-9                                      00038300
C                                                                       00038400
      DO 325 J=1,NP                                                     00038500
      IF (PROCLM(J) .LT. 0.) GO TO 325                                  00038600
      PROCLM(J) = PROCLM(J) / 10.**ISC                                  00038700
      WRITE(8,675) AP(J), PROCLM(J)                                     00038800
325   CONTINUE                                                          00038900
C                                                                       00039000
C     WRITE RHS FOR CONSTRAINT C-11                                     00039100
C                                                                       00039200
      IF (EQ8LM .LT. 0.) GO TO 345                                      00039300
      HPC=HP(1)                                                         00039400
      CARY=CAP(1)                                                       00039500
      DO 340 J=2,I                                                      00039600
      IF (HP(J).NE.HPC) GO TO 330                                       00039700
      IF (CAP(J).LT.CARY) CARY=CAP(J)                                   00039800
      GO TO 340                                                         00039900
330   JI=J-1                                                            00040000
      CARY=CARY+REQ(MF+1) * FAC                                         00040100
      IF (EQ8LM.NE.0.) CARY=EQ8LM                                       00040200
      WRITE (8,690) HP(JI),CARY                                         00040300
      HPC=HP(J)                                                         00040400
      CARY=CAP(J)                                                       00040500
```

```
340   CONTINUE                                                    00040600
      CARY=CARY+REQ(MF+1) * FAC                                   00040700
      IF (EQ8LM.NE.0.) CARY=EQ8LM                                 00040800
      WRITE (8,690) HP(I),CARY                                    00040900
C                                                                 00041000
C     WRITE RHS FOR CONSTRAINT C-12                               00041100
C                                                                 00041200
345   HPC=Z                                                       00041300
      HLC=Z                                                       00041400
      DO 370 J=1,I                                                00041500
      IF (HP(J).NE.HPC) GO TO 350                                 00041600
      IF (HL(J).EQ.HLC) GO TO 360                                 00041700
350   WRITE (8,700) HP(J),HL(J)                                   00041800
360   HPC=HP(J)                                                   00041900
370   HLC=HL(J)                                                   00042000
C                                                                 00042100
C     WRITE RHS FOR CONSTRAINT C-13                               00042200
C                                                                 00042300
      DO 375 J=1,NP                                               00042400
      IF (STORCP(J) .LT. 0.) GO TO 375                            00042500
      STORCP (J) = STORCP(J) / 10** ISC                           00042600
      WRITE(8,705) AP(J), STORCP(J)                               00042700
375   CONTINUE                                                    00042800
C                                                                 00043700
C     WRITING BOUNDS PORTION OF MATRIX                            00043800
C                                                                 00043900
C     WRITE HEADER                                                00044000
C                                                                 00044100
377   CONTINUE                                                    00044200
      WRITE (8,710)                                               00044300
C                                                                 00044400
C     WRITE INTEGER BOUNDS                                        00044500
C                                                                 00044600
      DO 380 J=1,I                                                00044700
380   WRITE (8,720) HP(J),HL(J),HA(J)                             00044800
C                                                                 00044900
```

```
C     WRITE LOWER BOUND FOR NEGATIVE INVENTORY                          00045000
C                                                                       00045100
      PIPEN = -PIPE                                                     00045300
390   WRITE(8,730) AP(1), PIPEN                                         00045600
C                                                                       00045700
C     WRITE BOUND FOR RR                                                00045800
C                                                                       00045900
395   CONTINUE                                                          00046000
      IF (MODE .EQ. 2) WRITE(8,735)                                     00046100
C                                                                       00046200
C     WRITE ENDATA CARD                                                 00046300
C                                                                       00046400
      WRITE (8,740)                                                     00046500
      WRITE (8,750)                                                     00046510
      IF (MODE .EQ. 1) WRITE (8,760)                                    00046520
      IF (MODE .EQ. 2) WRITE (8,770)                                    00046530
      WRITE(8,780) TITLID                                               00046540
      WRITE (8,740)                                                     00046550
      REWIND 8                                                          00046600
      STOP                                                              00046700
C                                                                       00046800
C     FORMAT STATEMENTS                                                 00046900
C                                                                       00047000
C     INPUT FORMATS                                                     00047100
C                                                                       00047200
  400 FORMAT(A1, 1X, 16A4, A2, 4X, I8)                                  00047300
  401 FORMAT(12H1INPUT DATA:/1HO, I8, 1X, A1, 1X, 8HTITLE - ,16A4,A2)   00047320
  405 FORMAT(A1, 1X, 2A4, 1X,I2,1X,I3,1X,A4, A3, 1X, F15.0, 8X, I1,     00047340
     1 F5.3, I1, I2, F10.0, 4X, I8)                                     00047350
  406 FORMAT(1HO, I8, 1X, A1, 1X, 14HSHORT TITLE - , 2A4,               00047360
     1 23H, NUMBER OF PRODUCTS - , I2, 26H, NUMBER OF ALTERNATIVES - ,  00047380
     2 I2, 29H, START OF PLANNING PERIOD - , A4, A3/ 12X,               00047400
     3 10HBUDGET - $, F15.0)                                            00047410
  410 FORMAT(A1, 1X, 4(F10.0, 1X), 26X, I8)                             00047420
  411 FORMAT (1HO, I8, 1X, A1, 22H INITIAL ALLOWANCES - , F10.0,        00047440
     1 27H, PEACETIME REQUIREMENTS - , F10.0, 11H, LOSSES - , F10.0/    00047460
```

```
     2 12X,   24HPIPELINE REQUIREMENTS - , F10.0)                          00047480
  414 FORMAT (1HO, 11X, 13HREQUIREMENTS:)                                  00047500
  415 FORMAT(2A1, 8F8.0, 1X, I1, 4X, I8)                                   00047520
  416 FORMAT (1X, I8, 1X, 2A1, 8(1X, F14.0))                               00047540
420   FORMAT (A1, 1X, 5A4, 1X, A1, 2(1X, F10.0), 1X, F10.8, 1X, F10.4     00047560
     1 , F8.0, I4)                                                         00047580
  421 FORMAT(1HO, I8, 1X, A1, 1X, 10HPRODUCT - , 5A4, 9H, CODE - , A1,     00047600
     1 22H, INVENTORY ON HAND - , F10.0, 23H, AVERAGE UNIT COST - $,       00047620
     2 F10.2/ 12X, 20HALLOCATION FACTOR -  , F10.8,                       00047640
     3 23H, PROCUREMENT FACTOR -   , F10.6)                               00047660
  425 FORMAT (A1, 1X, 5A4, 1X, 4A1, 2(1X, F10.0), 1X, F5.3, 1X, A1, 15X,00047680
     1 I8)                                                                 00047690
  426 FORMAT(1HO, I8, 1X, A1, 1X, 16HPRODUCER/LINE - , 5A4,                00047700
     1 33H, CODES IDENTIFYING ALTERNATIVE - , 4A1/ 12X,                    00047720
     2   19HMAXIMUM CAPACITY - , F10.0, 27H, MINIMUM SUSTAINING RATE - ,00047740
     3 F10.0, 16H, TEMPERATURE - , A1)                                     00047760
  430 FORMAT(A1, 1X, 2(F10.0,1X), 11A4, 4X, I8)                            00047780
  431 FORMAT(1H , I8, 1X, A1, 1X, 17HONE TIME COST - $, F10.0,             00047800
     1 17H, ANNUAL COST - $, F10.0, 16H, DESCRIPTION - , 11A4)             00047820
  434 FORMAT(1H , 11X, 20HBUILDUP CAPABILITY:  )                           00047840
  435 FORMAT(2A1, 8F8.0, 1X, I1, 4X, I8)                                   00047860
  436 FORMAT(1X, I8, 1X, 2A1, 8(1X, F14.0))                                00047880
437   FORMAT (1HO, I2, ' ALTERNATIVES READ IN ', I2, ' ALTERNATIVES '      00047883
     1 'SPECIFIED IN INPUT.  ADJUSTMENTS MADE FOR PROCESSING.')            00047886
C                                                                          00048000
C     ROWS FORMATS                                                         00048100
C                                                                          00048200
460   FORMAT (4HNAME, 10X, 5HBLOCK/ 4HROWS/ 8H N   COST/,                 00048300
     1 11H N   ADDCOST/ 6H L   RR)                                        00048310
470   FORMAT (4HNAME,10X,5HBLOCK/4HROWS/8H N   RR   /,8H L   COST)        00048400
480   FORMAT (5H G   M,I2,A1)                                              00048500
490   FORMAT (9H L   INVOH, A1)                                            00048600
491   FORMAT (7H E   BUY, A1)                                              00048650
495   FORMAT (8H L   PROC, A1)                                             00048700
500   FORMAT (6H L   PW,3A1,1HM,I2)                                        00048800
510   FORMAT (9H L   PCSTP,1A1)                                            00048900
```

```
520    FORMAT (7H L  SEL,2A1)                                                00049000
525    FORMAT (8H L  STOR, A1)                                               00049100
527    FORMAT (8H G  PIPE , A1)                                              00049200
C                                                                            00049300
C      COLUMNS FORMATS                                                       00049400
C                                                                            00049500
530    FORMAT (7HCOLUMNS)                                                    00049600
533    FORMAT (4X, 2HRR, 8X, 2HRR, 8X, 3H-1.)                                00049700
536    FORMAT(4X, 2HRR, 8X, 1HM, I2, A1, 6X, E12.5)                          00049800
537    FORMAT (4X, 2HRR, 8X, 4HPIPE, A1, 5X, E12.5)                          00049900
538    FORMAT(4X, 3HH-2, A1, 2H10 , 4X, 5HINVOH, A1, 4X, 2H1., 13X,          00050000
      1 4HPIPE, A1, 5X, 2H1.)                                                00050100
539    FORMAT (4X, 3HY-2, A1, I2, 4X, 4HSTOR, A1, 5X, 2H1.)                  00050200
540    FORMAT (4X,3HY-2,A1,I2,4X,1HM,I2,A1,6X,2H1.,13X,5HINVOH,A1,4X,2H1.00050300
      1)                                                                     00050400
542    FORMAT (4X, 3HH-1, A1, 2H10, 4X, 4HCOST, 6X, E12.5, 3X,               00050500
      1 4HPIPE, A1, 5X, 2H1./ 4X, 3HH-1, A1, 2H10, 4X, 3HBUY, A1, 6X,       00050600
      2 2H1.)                                                                00050610
543    FORMAT (4X, 3HY-1, A1, I2, 4X, 4HSTOR, A1, 5X, 2H1.)                  00050700
544    FORMAT (4X, 3HY-1, A1, I2, 4X, 3HBUY, A1, 6X, 2H1.)                   00050730
545    FORMAT (4X, 3HY-1, A1, 6X, 3HBUY, A1, 6X, 3H-1.)                      00050760
546    FORMAT (4X, 3HY-1, A1, I2, 4X, 4HPROC, A1, 5X, 2H1.)                  00050800
550    FORMAT (4X,3HY-1,A1,I2,4X,1HM,I2,A1,6X,2H1.,13X/4X,3HY-1,A1,I2,4X,00050900
      14HCOST,6X,E12.5)                                                      00051000
555    FORMAT (4X, 3HY-1, A1, I2, 4X, 7HADDCOST, 3X, E12.5)                  00051100
560    FORMAT (4X,1HX,3A1,2I2,2X,2HPW,3A1,1HM,I2,2X,2H1.,13X,1HM,I2,A1,6X00051200
      1,2H1.)                                                                00051300
570    FORMAT (4X,1HX,3A1,2I2,2X,1HM,I2,A1,6X,3H-1.)                         00051400
575    FORMAT (4X, 1HX,3A1,2I2, 2X, 1HM, I2, A1, 6X, 3H+1.)                  00051450
580    FORMAT (4X,1HX,3A1,2I2,2X,1HM,I2,A1,6X,2H1.,13X,2HPW,3A1,1HM,I2,2X00051500
      1,2H1.)                                                                00051600
590    FORMAT (4X,3HINT,I3,4X,8H'MARKER',17X,8H'INTORG')                     00051700
600    FORMAT (4X,1HI,3A1,6X,3HSEL,2A1,5X,2H1.)                              00051800
620    FORMAT (4X,1HI,3A1,6X,5HPOSTP,A1,4X,E12.5)                            00052100
630    FORMAT (4X,1HI,3A1,6X,2HPW,3A1,1HM,I2,2X,E12.5)                       00052200
635    FORMAT(4X,1HI,3A1,6X,4HCOST,6X,E12.5)                                 00052300
```

```
640   FORMAT (4X,3HINT,I3,4X,8H'MARKER',17X,8H'INTEND')                    00052400
C                                                                          00052500
C     RHS FORMATS                                                          00052600
C                                                                          00052700
650   FORMAT (3HRHS)                                                       00052800
660   FORMAT (4X, 6HCONSTR, 4X, 2HRR, 8X, E12.5)                           00053100
665   FORMAT(4X, 6HCONSTR, 4X, 4HCOST, 6X, E12.5)                          00053200
670   FORMAT (4X,6HCONSTR,4X,5HINVOH,A1,4X,E12.5)                          00053400
675   FORMAT (4X, 6HCONSTR, 4X, 4HPROC, A1, 5X, E12.5)                     00053500
690   FORMAT (4X,6HCONSTR,4X,5HPOSTP,A1,4X,E12.5)                          00053600
700   FORMAT (4X,6HCONSTR,4X,3HSEL,2A1,5X,2H1.)                            00053700
705   FORMAT (4X, 6HCONSTR, 4X, 4HSTOR, A1, 5X, E12.5)                     00053800
C                                                                          00054100
C     BOUNDS FORMATS                                                       00054200
C                                                                          00054300
710   FORMAT (6HBOUNDS)                                                    00054400
720   FORMAT (4H UP , 5HBOUND, 5X, 1H1, 3A1, 6X, 2H1.)                     00054500
730   FORMAT (4H LO , 5HBOUND, 5X, 3HY-2, A1, 2H11, 4X, E12.5)             00054700
735   FORMAT(4H UP , 5HBOUND, 5X, 2HRR, 8X, 2H1.)                          00054800
740   FORMAT (6HENDATA)                                                    00054900
C                                                                          00055000
                                                                           00055100
C     FORMATS FOR INPUT TO READ                                            00055600
C                                                                          00055800
750   FORMAT (4HNAME, 10X, 8HBLOCKCTL)                                     00055900
760   FORMAT (30HXMINMAX='MIN',XOBJ='COST'          )                      00055950
770   FORMAT (30HXMINMAX='MIN',XOBJ='RR'                              )    00056000
780   FORMAT(10HXCHAR01= ', A8, 1H')                                       00056100
      END                                                                  00056200
```

3. MPSX PROGRAM A READS THE MATRIX GENERATED BY THE FORTRAN SUB-ROUTINE. IT THEN PROCEEDS UNTIL REACHING AN OPTIMAL SOLUTION OR REACHING AN INTERNAL TIME CONTROL AT WHICH POINT IT SAVES THE

STATUS OF THE PROBLEM:

```
PROGRAM('ND')                                                                  00000100
        INITIALZ                                                               00007900
TITLE('JCAP IA/PT MODEL - INITIAL RUN')                                        00008000
         WRITE('START TIMERB')                                                 00008005
         TIMERB(XDELTM)                                                        00008010
         TYPE('JCAP IA/PT MODEL RUN - UP TO ', XDELTM, 'MIN CP TIME')          00008015
         ASSIGN('MATRIX','FT08F001','CARD')                                    00008020
         WRITE('START IAPTMG')                                                 00008025
         IAPTMG                                                                00008030
MOVE(XDATA,'BLOCK')                                                            00008100
        MOVE(XPBNAME,'PBFILE')                                                 00008200
CONVERT('FILE','MATRIX')                                                       00008300
MVADR(XDODELTM,JUMP)                                                           00008500
MOVE(XDATA,'BLOCKCTL')                                                         00008600
READ('FILE','MATRIX')                                                          00008700
SETUP('BOUND','BOUND')                                                         00008800
MOVE(XDATA,'BLOCKB')                                                           00008900
        MOVE(XRHS,'CONSTR')                                                    00009000
XFREQLGO=1000000                                                               00009100
        PRIMAL                                                                 00009200
        INIMIX                                                                 00009300
        MVADR(XDOPRINT,CONT)                                                   00009305
        MVADR(XDODELTM,JUMPI)                                                  00009310
        MVADR(XDOPRIM,A)                                                       00009315
        MIXSTART('COST')                                                       00009320
        MIXFLOW                                                                00009330
        MIXSAVE                                                                00009340
        MIXSTATS                                                               00009350
        SOLUTION('ACTIVE')                                                     00009400
        EXIT                                                                   00009500
JUMP    INVERT                                                                 00009900
        SAVE                                                                   00010000
```

```
EXIT                                                                    0001010
JUMPI     MIXSAVE                                                       00010200
          MIXSTATS                                                      00010210
          EXIT                                                          00010220
CONT      CONTINUE                                                      00010230
A         XINVERT=1                                                     00010240
           INVERT                                                       00010250
          XDUNFS=0                                                      00010260
          PRIMAL                                                        00010270
          XINVERT=0                                                     00010280
          MVADR(XDUNFS,B)                                               00010290
          CONTINUE                                                      00010300
B         MIXSAVE                                                       00010305
          MIXSTATS                                                      00010310
          EXIT                                                          00010320
          PEND                                                          00010330
```


4. MPSX PROGRAM B RESUMES WITH A PROBLEM WHERE THE CONTINUOUS OPTIMAL SOLUTION HAS NOT BENN FOUND AND PROCEEDS UNTIL REACHING AN OPTIMAL SOLUTION OR REACHING AN INTERNAL TIME CONTROL AT WHICH POINT IT SAVES THE PROBLEM:

```
PROGRAM('ND')                                                           00000100
          INITIALZ                                                      00007900
TITLE('JCAP IA/PT MODEL - RESUME IN CONTINUOUS PHASE')                  00008000
           WRITE('START TIMERB')                                        00008005
           TIMERB(XDELTM)                                               00008010
           TYPE('JCAP IA/PT MODEL RUN - UP TO ', XDELTM, 'MIN CP TIME') 00008015
MOVE(XDATA,'BLOCK')                                                     00008100
          MOVE(XPBNAME,'PBFILE')                                        00008200
          MOVE(XOLDNAME,'PBFILE')                                       00008300
```

```
          COPY('ENTIRE')                                        00008400
MVADR(XDODELTM,JUMP)                                            00008500
SETUP('BOUND','BOUND')                                          00008800
          RESTORE('STATUS')                                     00008850
MOVE(XDATA,'BLOCKB')                                            00008900
XFREQLGO=1000000                                                00009100
          PRIMAL                                                00009200
          INIMIX                                                00009300
            MVADR(XDOPRINT,CONT)                                00009305
          MVADR(XDODELTM,JUMPI)                                 00009310
          MVADR(XDOPRIM,A)                                      00009315
          MIXSTART('COST')                                      00009320
          MIXFLOW                                               00009330
          MIXSAVE                                               00009340
          MIXSTATS                                              00009350
          SOLUTION('ACTIVE')                                    00009400
          EXIT                                                  00009500
JUMP      INVERT                                                00009900
          SAVE                                                  00010000
EXIT                                                            00010100
JUMPI     MIXSAVE                                               00010200
          MIXSTATS                                              00010210
          EXIT                                                  00010220
CONT       CONTINUE                                             00010230
A         XINVERT=1                                             00010240
           INVERT                                               00010250
          XDONFS=0                                              00010260
          PRIMAL                                                00010270
          XINVERT=0                                             00010280
          MVADR(XDONFS,B)                                       00010290
          CONTINUE                                              00010300
B         MIXSAVE                                               00010305
          MIXSTATS                                              00010310
          EXIT                                                  00010320
          PEND                                                  00010330
```

5. MPSX PROGRAM C RESUMES WITH A PROBLEM IN THE INTEGER TREE SEARCH AND PROCEEDS UNTIL REACHING AN OPTIMAL SOLUTION OR REACHING AN INTERNAL TIME CONTROL AT WHICH POINT IT SAVES THE STATUS OF THE PROBLEM:

```
PROGRAM('ND')                                                                  00000100
         INITIALZ                                                              00007900
TITLE('JCAP IA/PT MODEL - RESUME IN MIP PHASE')                                00008000
          WRITE('START TIMERB')                                                00008005
          TIMERB(XDELTM)                                                       00008010
          TYPE('JCAP IA/PT MODEL RUN - UP TO ', XDELTM, 'MIN CP TIME')         00008015
MOVE(XDATA,'BLOCK')                                                            00008100
         MOVE(XPBNAME,'PBFILE')                                                00008200
         MOVE(XOLDNAME,'PBFILE')                                               00008300
         COPY('ENTIRE')                                                        00008400
SETUP('BOUND','BOUND')                                                         00008800
MOVE(XDATA,'BLOCKB')                                                           00008900
XFREQLGO=1000000                                                               00009100
         INIMIX                                                                00009300
         MVADR(XDOPRINT,CONT)                                                  00009305
         MVADR(XDODELTM,JUMPI)                                                 00009310
         MVADR(XDOPRIM,A)                                                      00009315
         MIXSTART('RESTORE')                                                   00009320
         MIXFLOW                                                               00009330
         MIXSAVE                                                               00009340
         MIXSTATS                                                              00009350
         SOLUTION('ACTIVE')                                                    00009400
         EXIT                                                                  00009500
JUMPI    MIXSAVE                                                               00010200
         MIXSTATS                                                              00010210
         EXIT                                                                  00010220
```

```
CONT     CONTINUE                                                    00010230
A        XINVERT=1                                                   00010240
          INVERT                                                     00010250
         XDONFS=0                                                    00010260
         PRIMAL                                                      00010270
         XINVERT=0                                                   00010280
         MVADR(XDONFS,B)                                             00010290
         CONTINUE                                                    00010300
B        MIXSAVE                                                     00010305
         MIXSTATS                                                    00010310
         EXIT                                                        00010320
         PEND                                                        00010330
```

6. THE FOLLOWING COBOL PROGRAM READS THE MPSX OUTPUT AND GENERATES A REPORT. FOR DETAILS SEE THE USERS' MANUAL.


```
000100 IDENTIFICATION DIVISION.
000200 PROGRAM-ID. 'RPTGN'.
000300 AUTHOR. SCHWEGLER.
000400 ENVIRONMENT DIVISION.
000500 CONFIGURATION SECTION.
000600 SPECIAL-NAMES.
000700     C01 IS NEXT-PAGE.
000800 INPUT-OUTPUT SECTION.
000900 FILE-CONTROL.
001000     SELECT GLOSSARY-IN ASSIGN TO UT-S-GLOSSARY.
001100     SELECT DATA-IN ASSIGN TO UT-S-DATA.
001200     SELECT OUTPUT-FILE ASSIGN TO UT-S-PRINT.
001300 DATA DIVISION.
001400 FILE SECTION.
001500 FD  DATA-IN
001600     DATA RECORDS ARE DISC-REC, SEARCH-REC,
001700     RECORDING MODE IS F,
```

```
001800      BLOCK CONTAINS 0 RECORDS,
001900      LABEL RECORDS ARE STANDARD.
002000 01   DISC-REC.
002100      02 FILLER  PIC X(10).
002200      02 IXY-CODE  PIC X.
002300      02 PLANT-CODE.
002400        03 ITEM-MADE   PIC X.
002500        03 FILLER PIC XX.
002600      02 CODE-FIELD REDEFINES PLANT-CODE.
002700        03 FILLER  PIC X.
002800        03 INV-CODE   PIC X.
002900        03 ITEM-CODE PIC X.
003000      02 MONTH-USE     PIC 99.
003100      02 PROD-MONTH REDEFINES MONTH-USE PIC 99.
003200      02 UTIL-MONTH  PIC 99.
003300      02 FILLER PIC X(6).
003400      02 INPUT-REC.
003500        03 ACT-INT PIC X(7).
003600        03 ACT OCCURS 6 TIMES PIC X.
003700      02 ACT-SIGN PIC X.
003800      02 FILLER PIC X(95).
003900 01   SEARCH-REC.
004000      02 FILLER PIC X.
004100      02 START-FILE PIC X(19).
004200      02 END-FILE REDEFINES START-FILE.
004300        03 END-KEY PIC X(12).
004400        03 FILLER PIC X(7).
004500      02 FILLER PIC X(113).
004600 FD   OUTPUT-FILE
004700      DATA RECORD IS PRINT-OUT,
004800      RECORDING MODE IS F,
004900      LABEL RECORD IS OMITTED.
005000 01   PRINT-OUT      PIC X(133).
005100 FD   GLOSSARY-IN
005200      DATA RECORDS ARE A-CARD, B-CARD, C-CARD, D1-CARD, D2-CARD,
005300        D3-CARD, E-CARD, F-CARD, G-CARD, H1-CARD, H2-CARD, H3-CARD,
```

```
005400      RECORDING MODE IS F,
005500      LABEL RECORDS ARE OMITTED.
005600 01   A-CARD.
005700      02 CC-A           PIC X.
005800      02 FILLER         PIC X.
005900      02 STUDY-TITLE    PIC X(66).
006000      02 FILLER         PIC X(12).
006100 01   B-CARD.
006200      02 CC-B           PIC X.
006300      02 FILLER         PIC X.
006400      02 TITLE-ID       PIC X(8).
006500      02 FILLER         PIC X.
006600      02 NO-PROD1       PIC XX.
006700      02 NO-PROD2 REDEFINES NO-PROD1 PIC 99.
006800      02 FILLER         PIC X.
006900      02 NO-ALT1        PIC XXX.
007000      02 NO-ALT2 REDEFINES NO-ALT1 PIC 999.
007100      02 FILLER         PIC X.
007200      02 FYDP-DATE      PIC X(7).
007300      02 FILLER         PIC X.
007400      02 FY-BUD1        PIC X(15).
007500      02 FY-BUD2 REDEFINES FY-BUD1 PIC 9(15).
007600      02 FILLER         PIC X(39).
007700 01   C-CARD.
007800      02 CC-C           PIC X.
007900      02 FILLER         PIC X.
008000      02 INT-ALLOW1         PIC X(10).
008100      02 INT-ALLOW2 REDEFINES INT-ALLOW1 PIC 9(10).
008200      02 FILLER         PIC X.
008300      02 PEACE-REQ1         PIC X(10).
008400      02 PEACE-REQ2 REDEFINES PEACE-REQ1 PIC 9(10).
008500      02 FILLER         PIC X.
008600      02 LOSSES1            PIC X(10).
008700      02 LOSSES2 REDEFINES LOSSES1 PIC 9(10).
008800      02 FILLER         PIC X.
008900      02 PIPE-REQ1          PIC X(10).
```

```
009000      02 PIPE-REQ2 REDEFINES PIPE-REQ1 PIC 9(10).
009100      02 FILLER        PIC X(35).
009200 01   D1-CARD.
009300      02 CC-D1         PIC XX.
009400      02 D1-MCR OCCURS 8 TIMES PIC X(8).
009500      02 FILLER        PIC X.
009600      02 EC-D11            PIC X.
009700      02 EC-D12 REDEFINES EC-D11 PIC 9.
009800      02 FILLER        PIC X(12).
009900 01   D2-CARD.
010000      02 CC-D2         PIC XX.
010100      02 D2-MCR OCCURS 8 TIMES PIC X(8).
010200      02 FILLER        PIC X.
010300      02 EC-D21            PIC X.
010400      02 EC-D22 REDEFINES EC-D21 PIC 9.
010500      02 FILLER        PIC X(12).
010600 01   D3-CARD.
010700      02 CC-D3         PIC XX.
010800      02 D3-MCR OCCURS 8 TIMES PIC X(8).
010900      02 FILLER        PIC X.
011000      02 EC-D31            PIC X.
011100      02 EC-D32 REDEFINES EC-D31 PIC 9.
011200      02 FILLER        PIC X(12).
011300 01   E-CARD.
011400      02 CC-E          PIC X.
011500      02 FILLER        PIC X.
011600      02 PROD-NAME         PIC X(20).
011700      02 FILLER            PIC X.
011800      02 PROD-CODE         PIC X.
011900      02 FILLER            PIC X.
012000      02 ASSETS-ON-HAND1    PIC X(10).
012100      02 ASSETS-ON-HAND2 REDEFINES ASSETS-ON-HAND1 PIC 9(10).
012200      02 FILLER            PIC X.
012300      02 UNIT-COST1        PIC X(5).
012400      02 FILLER            PIC X.
012500      02 UNIT-COST2       PIC XXXX.
```

```
012600      02 FILLER              PIC X.
012700      02 ALLOC-FAC1          PIC X.
012800      02 FILLER              PIC X.
012900      02 ALLOC-FAC2          PIC XXX.
013000      02 FILLER              PIC X.
013100      02 PRO-FAC1            PIC X(6).
013200      02 FILLER              PIC X.
013300      02 PRO-FAC2            PIC XXX.
013400      02 FILLER              PIC X(17).
013500 01   F-CARD.
013600      02 CC-F                PIC X.
013700      02 FILLER              PIC X.
013800      02 PROD-LINE           PIC X(20).
013900      02 FILLER              PIC X.
014000      02 PRODT-CODE          PIC X.
014100      02 PRODR-CODE          PIC X.
014200      02 ALTER-CODE          PIC X.
014300      02 PROD-SUPPORT        PIC X.
014400      02 FILLER              PIC X.
014500      02 MAX-CAP1            PIC X(10).
014600      02 MAX-CAP2 REDEFINES MAX-CAP1  PIC 9(10).
014700      02 FILLER              PIC X.
014800      02 MSR1                PIC X(10).
014900      02 MSR2 REDEFINES MSR1  PIC 9(10).
015000      02 FILLER              PIC X.
015100      02 LINE-ALLOC-CODE1 PIC X.
015200      02 FILLER              PIC X.
015300      02 LINE-ALLOC-CODE2 PIC XXX.
015400      02 FILLER              PIC X.
015500      02 BASE-TEMP           PIC X.
015600      02 FILLER              PIC X(23).
015700 01   G-CARD.
015800      02 CC-G                PIC X.
015900      02 FILLER              PIC X.
016000      02 OT-IPM-COSTS1  PIC X(10).
016100      02 OT-IPM-COSTS2 REDEFINES OT-IPM-COSTS1 PIC 9(10).
```

```
016200      02 FILLER              PIC X.
016300      02 A-IPM-CDSTS1        PIC X(10).
016400      02 A-IPM-CDSTS2 REDEFINES A-IPM-CDSTS1 PIC 9(10).
016500      02 FILLER              PIC X.
016600      02 IPM-DESC            PIC X(44).
016700      02 FILLER              PIC X(12).
016800 01   H1-CARD.
016900      02 CC-H1               PIC XX.
017000      02 PROD-BU-H1 OCCURS 8 TIMES PIC X(8).
017100      02 FILLER              PIC X.
017200      02 EC-H11              PIC X.
017300      02 EC-H12 REDEFINES EC-H11 PIC 9.
017400      02 FILLER              PIC X(12).
017500 01   H2-CARD.
017600      02 CC-H2               PIC XX.
017700      02 PROD-BU-H2 OCCURS 8 TIMES PIC X(8).
017800      02 FILLER              PIC X.
017900      02 EC-H21              PIC X.
018000      02 EC-H22 REDEFINES EC-H21 PIC 9.
018100      02 FILLER              PIC X(12).
018200 01   H3-CARD.
018300      02 CC-H3               PIC XX.
018400      02 PROD-BU-H3 OCCURS 8 TIMES PIC X(8).
018500      02 FILLER              PIC X.
018600      02 EC-H31              PIC X.
018700      02 EC-H32 REDEFINES EC-H31 PIC 9.
018800      02 FILLER              PIC X(12).
018900 WORKING-STORAGE SECTION.
019000 77   STORAGE-START PIC X(27) VALUE 'WORKING STORAGE STARTS HERE'.
019100 77   PREV-ITEM   PIC X VALUE SPACES.
019200 77   PREV-INV-CODE  PIC X VALUE SPACES.
019300 77   SWITCH-A    PIC XXX VALUE ' ON'.
019400 77   SWITCH-B  PIC XXX VALUE ' ON'.
019500 77   SWITCH-C  PIC XXX VALUE ' ON'.
019600 77   SWITCH-D  PIC XXX VALUE ' ON'.
019700 77   SWITCH-E  PIC XXX VALUE ' ON'.
```

```
019800 77  SWITCH-F  PIC XXX VALUE ' ON'.
019900 77  SWITCH-G  PIC XXX VALUE ' ON'.
020000 77  PLANT-CTR  PIC 99 VALUE 0.
020100 77  MONTH-CTR  PIC 99 VALUE 0.
020200 77  ITEM-CTR1  PIC 99 VALUE 0.
020300 77  NO-OF-ITEMS1    PIC 99 VALUE 0.
020400 77  ITEM-CTR2    PIC 99 VALUE 0.
020500 77  NO-OF-ITEMS2      PIC 99 VALUE 0.
020600 77  ITEM-CTR   PIC 99 VALUE 0.
020700 77  NO-OF-ITEMS        PIC 99 VALUE 0.
020800 77  NO-OF-PLANTS       PIC 99 VALUE 0.
020900 77  NO-OF-MONTHS   PIC 99 VALUE 0.
021000 77  PREV-PLANT        PIC XXX VALUE SPACES.
021100 77  GLOSSARY-CTR      PIC 99 VALUE 0.
021200 77  GLOSSARY-MAX       PIC 99 VALUE 0.
021300 77  SUB1    PIC 99 VALUE 0.
021400 77  SUB2     PIC 99 VALUE 0.
021500 77  SUB3      PIC 99 VALUE 0.
021600 77  SUB4 PIC 99 VALUE 0.
021700 77  SUB5 PIC 99 VALUE 0.
021800 77  DEC-CHK PIC 99 VALUE 0.
021900 77  DEC-COUNT PIC 99 VALUE 0.
022000 77  ZERO-COUNT PIC 99 VALUE 0.
022100 77  CTR1   PIC 99 VALUE 0.
022200 77  CTR2      PIC 99 VALUE 0.
022300 77  CTR3      PIC 99 VALUE 0.
022400 77  MULTIPLIER    PIC 99 VALUE 0.
022500 77  LINE-TOTAL1    PIC 9(6)V99 VALUE 0.
022600 77  LINE-TOTAL2    PIC 9(6)V99 VALUE 0.
022700 77  VAR1     PIC 99 VALUE 0.
022800 77  VAR2     PIC 99 VALUE 0.
022900 77  TOTAL-REQUIREMENTS    PIC 9(6)V99 VALUE 0.
023000 77  TABLE-CTR   PIC 99 VALUE 0.
023100 77  GRAND-TOTAL  PIC 9(12) VALUE ZERO.
023200 77  TOTAL-ITEM-COST PIC 9(10) VALUE 0.
023300 77  TOTAL-PLAN-COST   PIC 9(11) VALUE 0.
```

```
023400 77   IPM-TOTAL-COST PIC 9(9)V99 VALUE 0.
023500 77   IPM-COST PIC 9(9)V99 VALUE 0.
023600 77   NO-OF-ALTERNATIVES       PIC 99 VALUE 0.
023700 77   TITLE-OF-STUDY       PIC X(66) VALUE SPACES.
023800 77   SHORT-TITLE          PIC X(8) VALUE SPACES.
023900 77   FYDP-START           PIC X(7) VALUE SPACES.
024000 77   FY-BUDGET            PIC 9(15) VALUE 0.
024100 77   INITIAL-ALLOWANCES   PIC 9(10) VALUE 0.
024200 77   PEACE-REQUIREMENT    PIC 9(10) VALUE 0.
024300 77   LOSSES-GONE      PIC 9(10) VALUE 0.
024400 77   PIPELINE-REQUIREMENT       PIC 9(10) VALUE 0.
024500 77   COST-CHECK PIC 9(12) VALUE 0.
024600 77   STORAGE-END PIC X(25) VALUE 'WORKING STORAGE ENDS HERE'.
024700 01   ACTIVITY-NO.
024800        02 BASE-ACT.
024900          03 ACT-INT-NO PIC X(7).
025000          03 ACT-NO OCCURS 6 TIMES PIC X.
025100        02 ACTIVITY1 REDEFINES BASE-ACT PIC 9(7)V9(6).
025200        02 ACTIVITY2 REDEFINES BASE-ACT PIC 9(8)V9(5).
025300        02 ACTIVITY3 REDEFINES BASE-ACT PIC 9(9)V9(4).
025400        02 ACTIVITY4 REDEFINES BASE-ACT PIC 9(10)V999.
025500        02 ACTIVITY5 REDEFINES BASE-ACT PIC 9(11)V99.
025600        02 ACTIVITY6 REDEFINES BASE-ACT PIC 9(10)V9.
025700        02 ACTIVITY7 REDEFINES BASE-ACT PIC 9(13).
025800 01   TABLE-NUMBER-LINE.
025900        02 TABLE-WORD PIC X(68) VALUE 'TABLE' JUSTIFIED RIGHT.
026000        02 TABLE-NO  PIC ZZ9.
026100        02 FILLER PIC X(62) VALUE SPACES.
026200 01   SUMMARY-UTIL-TOTALS.
026300        02 ITEM-UTIL OCCURS 20 TIMES.
026400          03 ITEM-MONTH-UTIL OCCURS 26 TIMES PICTURE 9(8).
026500 01   SUMMARY-TOTALS.
026600        02 ITEM-SUMMARY OCCURS 20 TIMES.
026700          03 ITEM-MONTH-TOTAL OCCURS 26 TIMES PIC 9(8).
026800 01   SUMMARY-GRAND-TOTALS.
026900        02 ITEM-GRAND-TOTAL OCCURS 20 TIMES PIC 9(6)V99.
```

```
027000 01  SUMMARY-REQUIREMENTS.
027100     02 REQ OCCURS 26 TIMES PIC 9(8).
027200 01  ACTIVITY-JOIN.
027300     02 PARTS-JOIN.
027400       03 PART1 PIC 9(12).
027500       03 PART2   PIC 9(5).
027600     02 ACTIVITY REDEFINES PARTS-JOIN PIC S9(12)V9(5).
027700 01  ITEM-COST-TABLE.
027800     02 A-ITEM-LIST OCCURS 20 TIMES.
027900       03 A-ITEM-COST PIC 9(5)V9999.
028000       03 A-ITEM-NAME PIC X(20).
028100       03 A-ITEM-ASSETS PIC 9(8).
028200 01  ITEM1-TABLE.
028300     02 ITEM1 OCCURS 20 TIMES PIC X.
028400 01  ITEM2-TABLE.
028500     02 ITEM2 OCCURS 20 TIMES PIC X.
028600 01  ITEM-TABLE.
028700     02 ITEM OCCURS 20 TIMES PIC X.
028800 01  PLANT-STORAGE.
028900     02 PLANT OCCURS 50 TIMES.
029000       03 PLANT-ITEM  PIC X.
029100       03 PLANT-NAME-ALT   PIC XX.
029200 01  ASSETS-ON-HAND-TABLE.
029300     02 ITEM-ON-HAND OCCURS 20 TIMES.
029400       03 QTY-ON-HAND OCCURS 26 TIMES PIC S9(8).
029500 01  ASSETS-ON-HAND-TOTALS.
029600     02 TOTAL-ASSETS-ON-HAND OCCURS 50 TIMES PIC S9(8).
029700 01  ASSETS-TO-BUY-TABLE.
029800     02 ITEM-USE OCCURS 20 TIMES.
029900         03 QTY OCCURS 26 TIMES PIC 9(8).
030000 01  IA-PT-BUY-HD1.
030100     02 FILLER PIC X(71) VALUE 'IA/PT PROCUREMENT PLAN FOR '
030200       JUSTIFIED RIGHT.
030300     02 IA-PT-ITEM PIC X(20).
030400     02 FILLER PIC X(42) VALUE SPACES.
030500 01  IA-PT-BUY-HD2.
```

```
030600      02  FILLER PIC X(34) VALUE 'ITEM' JUSTIFIED RIGHT.
030700      02 FILLER PIC X(25) VALUE 'UNIT COST' JUSTIFIED RIGHT.
030800      02 FILLER PIC X(15) VALUE 'BUY QUANTITY' JUSTIFIED RIGHT.
030900      02 FILLER PIC X(18) VALUE 'TOTAL ITEM COST' JUSTIFIED RIGHT.
031000      02 FILLER PIC X(41) VALUE SPACES.
031100 01   IA-PT-BUY-LINE.
031200      02 FILLER PIC X(28) VALUE SPACES.
031300      02 IA-PT-NAME PIC X(20).
031400      02 FILLER PIC X(2) VALUE SPACES.
031500      02 IA-PT-COST PIC $Z(5).9999.
031600      02 FILLER PIC X(2) VALUE SPACES.
031700      02 IA-PT-BUY PIC ZZ,ZZZ,ZZ9.
031800      02 FILLER PIC X(4) VALUE SPACES.
031900      02 TOTAL-IA-PT-ITEM-COST PIC $ZZ,ZZZ,ZZZ,ZZ9.
032000      02 FILLER PIC X(40) VALUE SPACES.
032100 01   IA-PT-TOTAL-PLAN-COST.
032200      02 FILLER PIC X(76) VALUE 'THE COST FOR THE PROCUREMENT PLAN
032300-      'IS ' JUSTIFIED RIGHT.
032400      02 IA-PT-TOTAL-COST-OUT PIC $ZZ,ZZZ,ZZZ,ZZ9.
032500      02 FILLER PIC X(42) VALUE SPACES.
032600 01   ASSETS-TO-BUY-TOTALS.
032700      02 TOTAL-ASSETS-TO-BUY OCCURS 50 TIMES PIC 9(8).
032800 01   PROD-UTIL-TABLE.
032900      02 PROD-UTIL-USE OCCURS 50 TIMES.
033000        03 PROD-UTIL-MONTH OCCURS 26 TIMES.
033100          04 ELEM OCCURS 2 TIMES PIC 9(8).
033200 01   PROD-UTIL-TABLE-ZERO REDEFINES PROD-UTIL-TABLE PIC X(20000).
033300 01   UTIL-TOTALS-TABLE.
033400      02 TOTAL-UTIL OCCURS 50 TIMES PIC 9(8).
033500 01   PROD-TOTALS-TABLE.
033600      02 TOTAL-PROD OCCURS 50 TIMES PIC 9(8).
033700 01   ERR-MESS1.
033800      02 FILLER   PIC X(10) VALUE SPACES.
033900      02 FILLER  PIC X(24) VALUE 'ERROR IN PRODUCT-SEARCH1'.
034000      02 FILLER PIC X(99) VALUE SPACES.
034100 01   ERR-MESS2.
```

```
034200      02 FILLER PIC X(10) VALUE SPACES.
034300      02 FILLER PIC X(25) VALUE 'RECEIVING FIELD TOO SMALL'.
034400      02 FILLER PIC X(97) VALUE SPACES.
034500 01   ERR-MESS3.
034600      02 FILLER PIC X(5) VALUE SPACES.
034700      02 FILLER PIC X(75) VALUE 'REPORT WRITER TOTAL COST DIFFERS F
034800-     'ROM MPSX TOTAL COST BY MORE THAN 10 UNITS'.
034900      02 FILLER PIC X(53) VALUE SPACES.
035000 01   ERR-MESS4.
035100      02 FILLER PIC X(10) VALUE SPACES.
035200      02 FILLER PIC X(14) VALUE 'ERROR IN LOOP2'.
035300      02 FILLER PIC X(109) VALUE SPACES.
035400 01   IPM-HD1.
035500      02 FILLER PIC X(89) VALUE 'INDUSTRIAL PREPAREDNESS MEASURES S
035600-     'ELECTED FOR ' JUSTIFIED RIGHT.
035700      02 IPM-HD1-ITEM PIC X(20).
035800      02 FILLER PIC X(24) VALUE SPACES.
035900 01   IPM-HD4.
036000      02 FILLER PIC X(8) VALUE 'PLANT' JUSTIFIED RIGHT.
036100      02 FILLER PIC X(14) VALUE 'PRODUCT' JUSTIFIED RIGHT.
036200      02 FILLER PIC X(18) VALUE 'ONE-TIME-COST' JUSTIFIED RIGHT.
036300      02 FILLER PIC X(16) VALUE 'ANNUAL-COST' JUSTIFIED RIGHT.
036400      02 FILLER PIC X(18) VALUE 'TOTAL-IPM-COST' JUSTIFIED RIGHT.
036500      02 FILLER PIC X(21) VALUE 'DESCRIPTION OF IPM' JUSTIFIED
036600          RIGHT.
036700      02 FILLER PIC X(38) VALUE SPACES.
036800      02 FILLER PIC X(38) VALUE SPACES.
036900 01   IPM-LINE.
037000      02 FILLER PIC X VALUE SPACES.
037100      02 PLANT-T PIC X(10).
037200      02 FILLER PIC XXX VALUE SPACES.
037300      02 PRODUCT-T PIC X(10).
037400      02 FILLER PIC XXX VALUE SPACES.
037500      02 OTC PIC $Z,ZZZ,ZZZ,ZZ9.
037600      02 FILLER PIC X(3) VALUE SPACES.
037700      02 AC PIC $Z,ZZZ,ZZZ,ZZ9.
```

```
037800      02 FILLER PIC XXX VALUE SPACES.
037900      02 IPM-TC PIC $Z,ZZZ,ZZZ,ZZ9.
038000      02 FILLER PIC XXX VALUE SPACES.
038100      02 IPM-NOTE1 PIC X(22).
038200      02 FILLER PIC X(33) VALUE SPACES.
038300 01   IPM-LINE2.
038400      02 FILLER PIC X VALUE SPACES.
038500      02 PLANT-T2 PIC X(10).
038600      02 FILLER PIC XX VALUE SPACES.
038700      02 PRODUCT-T2 PIC X(10).
038800      02 FILLER PIC X(54) VALUE SPACES.
038900      02 IPM-NOTE2 PIC X(22).
039000      02 FILLER PIC X(33).
039100 01   IPM-COST-LINE.
039200      02 FILLER PIC X(80) VALUE 'THE COST FOR ALL INDUSTRIAL PREPAR
039300-     'EDNESS MEASURES SELECTED IS ' JUSTIFIED RIGHT.
039400      02 IPM-AMT PIC $ZZ,ZZZ,ZZZ,ZZ9.
039500      02 FILLER PIC X(40) VALUE SPACES.
039600 01   GRAND-TOTAL-LINE.
039700      02 FILLER PIC X(80) VALUE 'THE TOTAL COST FOR THE IA/PT SOLUT
039800-      'ION IS ' JUSTIFIED RIGHT.
039900      02 GRAND-AMT PIC $ZZZ,ZZZ,ZZZ,ZZ9.
040000      02 FILLER PIC X(37) VALUE SPACES.
040100 01   MOBIL-REQUIRE-TABLE.
040200      02 MOB-REQ OCCURS 25 TIMES PIC 9(8).
040300 01   PRODUCTS-TABLE.
040400      02 PRODUCTS-LIST OCCURS 10 TIMES.
040500        03 PRODUCT          PIC X(20).
040600        03 PRODUCT-CODE PIC X.
040700        03 PRODUCT-ASSETS PIC 9(10).
040800        03 PRODUCT-UNIT-COST PIC 9(5)V9999.
040900        03 PRODUCT-ALLOC-FACTOR  PIC 9V999.
041000        03 PROCURE-FACTOR  PIC 9(6)V999.
041100 01   PLANT-TABLE.
041200      02 PLANT-LIST OCCURS 20 TIMES.
041300        03 APROD-LINE          PIC X(20).
```

```
041400           03 APLANT-CODE.
041500             04 APRODT-CODE PIC X.
041600             04 APRODR-CODE PIC X.
041700             04 AALTER-CODE PIC X.
041800         03 APROD-SUPPORT  PIC X.
041900         03 AMAX-CAP       PIC 9(10).
042000         03 AMSR           PIC 9(10).
042100         03 ALINE-ALLOC-CODE PIC 9V999.
042200         03 ABASE-TEMP     PIC X.
042300         03 ACT-IPM-COSTS      PIC 9(10).
042400         03 AA-IPM-COSTS       PIC 9(10).
042500         03 AIPM-DESC          PIC X(44).
042600         03 ABUILD-UP OCCURS 25 TIMES PIC 9(8).
042700 01   GLOSSARY-TABLE.
042800      02 GLIST OCCURS 50 TIMES.
042900         03 ITEM-SYMBOL        PIC X.
043000         03 ITEM-NAME.
043100           04 ITEM-NAME1 PIC X(10).
043200           04 ITEM-NAME2 PIC X(10).
043300         03 PLANT-SYMBOL      PIC XXX.
043400         03 PLANT-NAME.
043500           04 PLANT-NAME1 PIC X(10).
043600           04 PLANT-NAME2 PIC X(10).
043700         03 ITEM-UNIT-COST     PIC 9(5)V9999.
043800         03 ONE-TIME-COST      PIC 9(10).
043900         03 ANNUAL-COST        PIC 9(10).
044000         03 IPM-DESCRIPTION.
044100           04 IPM-DESCRIPTION1 PIC X(22).
044200           04 IPM-DESCRIPTION2 PIC X(22).
044300         03 IPM-CODE           PIC X.
044400 01   UNIT-COST-JOIN.
044500      02 UC-STORE.
044600         03 UC-PART1     PIC X(5).
044700         03 UC-PART2       PIC XXXX.
044800      02 UC-DUM REDEFINES UC-STORE PIC 9(5)V9999.
044900 01   ALLOC-FAC-JOIN.
```

```
045000      02 AF-STORE.
045100        03 AF-PART1              PIC X.
045200        03 AF-PART2              PIC XXX.
045300      02 AF-DUM REDEFINES AF-STORE PIC 9V999.
045400 01   PROC-FAC-JOIN.
045500      02 PF-STORE.
045600        03 PF-PART1         PIC X(6).
045700        03 PF-PART2         PIC XXX.
045800      02 PF-DUM REDEFINES PF-STORE PIC 9(6)V999.
045900 01   NEW-FORM.
046000      02 ALPHA           PIC X(8).
046100      02 NUM REDEFINES ALPHA PIC 9(8).
046200 01   PAGE-HEADING.
046300      02 FILLER PIC X(44) VALUE SPACES.
046400      02 PAGE-TITLE PIC X(66).
046500      02 FILLER PIC X(23) VALUE SPACES.
046600 01   GLOSSARY-CHECK.
046700      02 FILLER PIC X(5) VALUE SPACES.
046800      02 GLOSS-OUT.
046900        03 ITEM-INFO PIC X(21).
047000        03 PLANT-INFO PIC X(23).
047100        03 COST-INFO.
047200          04 INFO1 PIC 9(10).
047300          04 INFO2 PIC 9(10).
047400          04 INFO3 PIC 9(10).
047500        03 IPM-INFO PIC X(45).
047600      02 FILLER PIC X(9) VALUE SPACES.
047700 01   ITEM1-CHECK.
047800      02 FILLER PIC X(10) VALUE SPACES.
047900      02 ITEM1-OUT PIC X.
048000      02 FILLER PIC X(122) VALUE SPACES.
048100 PROCEDURE DIVISION.
048200      OPEN INPUT GLOSSARY-IN, OUTPUT OUTPUT-FILE.
048300      MOVE SPACES TO PRODUCTS-TABLE.
048400      MOVE SPACES TO PLANT-TABLE.
048500      MOVE SPACES TO GLOSSARY-TABLE.
```

```
048510* INPUT-READ THRU EOJ1 ESTABLISHES THE GLOSSARY-TABLE USED TO
048520* TRANSLATE THE CODES USED IN THE MPSX PROGRAM.
048600 INPUT-READ.
048700     READ GLOSSARY-IN AT END GO TO NEXT-STEP.
048800     IF CC-A = 'A' MOVE STUDY-TITLE TO PAGE-TITLE.
048900     IF CC-A = 'A' MOVE STUDY-TITLE TO TITLE-OF-STUDY GO TO
049000       INPUT-READ.
049100     IF CC-B = 'B' MOVE TITLE-ID TO SHORT-TITLE
049200       EXAMINE NO-PROD1 REPLACING ALL ' ' BY '0'
049300       MOVE NO-PROD2 TO NO-OF-ITEMS
049400       EXAMINE NO-ALT1 REPLACING ALL ' ' BY '0'
049500       MOVE NO-ALT2 TO NO-OF-ALTERNATIVES MOVE
049600     FYDP-DATE TO FYDP-START EXAMINE FY-BUD1 REPLACING ALL ' '
049700       BY '0' MOVE FY-BUD2 TO FY-BUDGET GO TO
049800       INPUT-READ.
049900     IF CC-C = 'C' EXAMINE INT-ALLOW1 REPLACING ALL ' ' BY '0'
050000       MOVE INT-ALLOW2 TO INITIAL-ALLOWANCES
050100       EXAMINE PEACE-REQ1 REPLACING ALL ' ' BY '0'
050200     MOVE PEACE-REQ2 TO PEACE-REQUIREMENT
050300     EXAMINE LOSSES1 REPLACING ALL ' ' BY '0'
050400       MOVE LOSSES2 TO LOSSES-GONE
050500       EXAMINE PIPE-REQ1 REPLACING ALL ' ' BY '0'
050600       MOVE PIPE-REQ2 TO PIPELINE-REQUIREMENT GO TO INPUT-READ.
050700     IF CC-D1 = 'D1' MOVE 0 TO CTR2 PERFORM REQ-ROUT THRU
050800       REQ-EXIT VARYING CTR1 FROM 1 BY 1 UNTIL CTR1 > 8.
050900     IF CC-D1 = 'D1' GO TO INPUT-READ.
051000     IF CC-D2 = 'D2' PERFORM REQ-ROUT THRU REQ-EXIT VARYING
051100       CTR1 FROM 1 BY 1 UNTIL CTR1 > 8.
051200     IF CC-D2 = 'D2' GO TO INPUT-READ.
051300     IF CC-D3 = 'D3' PERFORM REQ-ROUT THRU REQ-EXIT VARYING
051400       CTR1 FROM 1 BY 1 UNTIL CTR1 > 8.
051500     IF CC-D3 = 'D3' MOVE 0 TO CTR1 GO TO INPUT-READ.
051600     IF CC-E = 'E' PERFORM E-ROUT.
051700     IF CC-E = 'E' GO TO INPUT-READ.
051800     IF CC-F = 'F' AND SWITCH-A = ' ON'
051900     MOVE CTR2 TO NO-OF-MONTHS
```

```
052000         MOVE 0 TO CTR2
052100         MOVE 0 TO CTR1
052200         MOVE 'OFF' TO SWITCH-A.
052300       IF CC-F = 'F' AND SWITCH-A = 'OFF' PERFORM F-ROUT.
052400       IF CC-F = 'F' AND SWITCH-A = 'OFF' GO TO INPUT-READ.
052500       IF CC-G = 'G' PERFORM G-ROUT.
052600       IF CC-G = 'G' GO TO INPUT-READ.
052700       IF CC-H1 = 'H1' PERFORM BU-ROUT THRU BU-EXIT VARYING CTR1
052800         FROM 1 BY 1 UNTIL CTR1 > 8.
052900       IF CC-H1 = 'H1' GO TO INPUT-READ.
053000       IF CC-H2 = 'H2' PERFORM BU-ROUT THRU BU-EXIT VARYING CTR1
053100         FROM 1 BY 1 UNTIL CTR1 > 8.
053200       IF CC-H2 = 'H2' GO TO INPUT-READ.
053300       IF CC-H3 = 'H3' PERFORM BU-ROUT THRU BU-EXIT VARYING CTR1
053400         FROM 1 BY 1 UNTIL CTR1 > 8.
053500       IF CC-H3 = 'H3' GO TO INPUT-READ.
053510* NEXT-STEP THRU PRODUCT-SEARCH2 MOVES DATA FROM AUXILLARY
053520* STORAGE AREAS INTO THE GLOSSARY-TABLE.
053600 NEXT-STEP.
053700       MOVE NO-OF-ALTERNATIVES TO GLOSSARY-MAX.
053800       MOVE NO-OF-ALTERNATIVES TO NO-OF-PLANTS.
053900       MOVE 0 TO CTR1.
054000       MOVE 0 TO CTR2.
054100       MOVE 0 TO CTR3.
054200 GLOSSARY-ROUT.
054300       ADD 1 TO CTR2.
054400       MOVE APROD-LINE (CTR2) TO PLANT-NAME (CTR2).
054500       MOVE APRODT-CODE (CTR2) TO ITEM-SYMBOL (CTR2).
054600       MOVE APLANT-CODE (CTR2) TO PLANT-SYMBOL (CTR2).
054700       MOVE AALTER-CODE (CTR2) TO IPM-CODE (CTR2).
054800       MOVE AOT-IPM-COSTS (CTR2) TO ONE-TIME-COST (CTR2).
054900       MOVE AA-IPM-COSTS (CTR2) TO ANNUAL-COST (CTR2).
055000       MOVE AIPM-DESC (CTR2) TO IPM-DESCRIPTION (CTR2).
055100 PRODUCT-SEARCH1.
055200       ADD 1 TO CTR1.
055300       IF APRODT-CODE (CTR2) = PRODUCT-CODE (CTR1) MOVE
```

```
055400         PRODUCT (CTR1) TO ITEM-NAME (CTR2)
055500         MOVE PRODUCT-UNIT-COST (CTR1) TO ITEM-UNIT-COST (CTR2)
055600         MOVE 0 TO CTR1 GO TO PRODUCT-SEARCH2.
055700       IF CTR1 = NO-OF-ITEMS GO TO ERROR-MSG1 ELSE GO TO
055800         PRODUCT-SEARCH1.
055900 PRODUCT-SEARCH2.
056000       IF CTR2 = NO-OF-PLANTS GO TO EOJ1 ELSE GO TO
056100         GLOSSARY-ROUT.
056110* THE REQ-ROUT PARAGRAPH STORES DATA FROM THE D1, D2, AND D3
056120* CARDS IN THE MOBIL-REQ-TABLE.
056200 REQ-ROUT.
056300       ADD 1 TO CTR2.
056400       IF CC-D1 = 'D1' MOVE D1-MCR (CTR1) TO ALPHA EXAMINE ALPHA
056500         REPLACING ALL ' ' BY '0' MOVE NUM TO MOB-REQ (CTR2).
056600       IF CC-D2 = 'D2' MOVE D2-MCR (CTR1) TO ALPHA EXAMINE ALPHA
056700         REPLACING ALL ' ' BY '0' MOVE NUM TO MOB-REQ (CTR2).
056800       IF CC-D3 = 'D3' MOVE D3-MCR (CTR1) TO ALPHA EXAMINE ALPHA
056900         REPLACING ALL ' ' BY '0' MOVE NUM TO MOB-REQ (CTR2).
057000       GO TO REQ-EXIT.
057100 REQ-EXIT.
057200       EXIT.
057210* E-ROUT TAKES DATA FROM THE E-CARDS, ZERO FILLS THE NUMERIC
057220* FIELDS AND STORES THE DATA IN THE PRODUCTS-TABLE.
057300 E-ROUT.
057400       ADD 1 TO CTR1.
057500       MOVE PROD-NAME TO PRODUCT (CTR1).
057600       MOVE PROD-CODE TO PRODUCT-CODE (CTR1).
057700       EXAMINE ASSETS-ON-HAND1 REPLACING ALL ' ' BY '0'.
057800       MOVE ASSETS-ON-HAND2 TO PRODUCT-ASSETS (CTR1).
057900       EXAMINE UNIT-COST1 REPLACING ALL ' ' BY '0'.
058000       EXAMINE UNIT-COST2 REPLACING ALL ' ' BY '0'.
058100       MOVE UNIT-COST1 TO UC-PART1.
058200       MOVE UNIT-COST2 TO UC-PART2.
058300       MOVE UC-DUM TO PRODUCT-UNIT-COST (CTR1).
058400       EXAMINE ALLOC-FAC1 REPLACING ALL ' ' BY '0'.
058500       EXAMINE ALLOC-FAC2 REPLACING ALL ' ' BY '0'.
```

```
058600      MOVE ALLOC-FAC1 TO AF-PART1.
058700      MOVE ALLOC-FAC2 TO AF-PART2.
058800      MOVE AF-DUM TO PRODUCT-ALLOC-FACTOR (CTR1).
058900      EXAMINE PRO-FAC1 REPLACING ALL ' ' BY '0'.
059000      EXAMINE PRO-FAC2 REPLACING ALL ' ' BY '0'.
059100      MOVE PRO-FAC1 TO PF-PART1.
059200      MOVE PRO-FAC2 TO PF-PART2.
059300      MOVE PF-DUM TO PROCURE-FACTOR (CTR1).
059310* F-ROUT TAKES DATA FROM THE F-CARDS, ZERO FILLS THE NUMERIC
059320* FIELDS AND STORES THE DATA IN THE PLANT-TABLE.
059400 F-ROUT.
059500      ADD 1 TO CTR2.
059600      MOVE PROD-LINE TO APROD-LINE (CTR2).
059700      MOVE PRODT-CODE TO APRODT-CODE (CTR2).
059800      MOVE PRODR-CODE TO APRODR-CODE (CTR2).
059900      MOVE ALTER-CODE TO AALTER-CODE (CTR2).
060000      MOVE PROD-SUPPORT TO APROD-SUPPORT (CTR2).
060100      EXAMINE MAX-CAP1 REPLACING ALL ' ' BY '0'.
060200      MOVE MAX-CAP2 TO AMAX-CAP (CTR2).
060300      EXAMINE MSR1 REPLACING ALL ' ' BY '0'.
060400      MOVE MSR2 TO AMSR (CTR2).
060500      EXAMINE LINE-ALLOC-CODE1 REPLACING ALL ' ' BY '0'.
060600      MOVE LINE-ALLOC-CODE1 TO AF-PART1.
060700      EXAMINE LINE-ALLOC-CODE2 REPLACING ALL ' ' BY '0'.
060800      MOVE LINE-ALLOC-CODE2 TO AF-PART2.
060900      MOVE AF-DUM TO ALINE-ALLOC-CODE (CTR2).
061000      MOVE BASE-TEMP TO ABASE-TEMP (CTR2).
061010* G-ROUT ZERO FILLS NUMERIC FIELDS FROM THE G-CARD AND STORES
061020* DATA IN THE PLANT-TABLE.
061100 G-ROUT.
061200      EXAMINE OT-IPM-COSTS1 REPLACING ALL ' ' BY '0'.
061300      MOVE OT-IPM-COSTS2 TO AOT-IPM-COSTS (CTR2).
061400      EXAMINE A-IPM-COSTS1 REPLACING ALL ' ' BY '0'.
061500      MOVE A-IPM-COSTS2 TO AA-IPM-COSTS (CTR2).
061600      MOVE IPM-DESC TO AIPM-DESC (CTR2).
061610* BU-ROUT ZERO FILLS NUMERIC FIELDS FROM THE H1,H2,AND H3 CARDS
```

```
061620* AND STORES DATA IN THE PLANT-TABLE.
061700 BU-ROUT.
061800     ADD 1 TO CTR3.
061900     IF CC-H1 = 'H1' MOVE PROD-BU-H1 (CTR1) TO
062000       ALPHA EXAMINE ALPHA REPLACING ALL ' ' BY '0' MOVE NUM TO
062100       ABUILD-UP (CTR2, CTR3).
062200     IF CC-H2 = 'H2' MOVE PROD-BU-H2 (CTR1) TO
062300       ALPHA EXAMINE ALPHA REPLACING ALL ' ' BY '0' MOVE NUM TO
062400       ABUILD-UP (CTR2, CTR3).
062500     IF CC-H3 = 'H3' MOVE PROD-BU-H3 (CTR1) TO
062600       ALPHA EXAMINE ALPHA REPLACING ALL ' ' BY '0' MOVE NUM TO
062700       ABUILD-UP (CTR2, CTR3).
062800     IF CTR3 = NO-OF-MONTHS MOVE 0 TO CTR3.
062900     GO TO BU-EXIT.
063000 BU-EXIT.
063100     EXIT.
063200 EOJ1.
063300     MOVE 0 TO CTR1.
063400     MOVE 0 TO CTR2.
063500     MOVE 0 TO CTR3.
063600     MOVE ' ON' TO SWITCH-A.
063700     CLOSE GLOSSARY-IN.
063800     OPEN INPUT DATA-IN.
063900     MOVE ZEROS TO ASSETS-ON-HAND-TABLE.
064000     MOVE ZEROS TO ASSETS-ON-HAND-TOTALS.
064100     MOVE ZEROS TO ASSETS-TO-BUY-TABLE.
064200     MOVE ZEROS TO ASSETS-TO-BUY-TOTALS.
064300     MOVE ZEROS TO PROD-UTIL-TABLE-ZERO.
064400     MOVE ZEROS TO UTIL-TOTALS-TABLE.
064500     MOVE ZEROS TO PROD-TOTALS-TABLE.
064600     MOVE ZEROS TO SUMMARY-GRAND-TOTALS.
064700     MOVE ZEROS TO SUMMARY-REQUIREMENTS.
064800 DATA-READ.
064900     READ DATA-IN AT END GO TO EOJ2.
065000     IF START-FILE = 'SECTION 1 - ROWS    ' MOVE 'OFF' TO
065100      SWITCH-B GO TO DATA-READ.
```

```
065200      IF SWITCH-B = 'ON' GO TO DATA-READ.
065210* THE FOLLOWING 'IF' CLAUSE LOCATES THE COST ASSOCIATED WITH
065220* THE IA/PT RUN BEING REPORTED AND STORES THE COST IN THE
065230* COST-CHECK FIELD FOR LATER REFERENCE.
065300      IF IXY-CODE = 'C' AND PLANT-CODE = 'OST'
065400      PERFORM DECIMAL-LOCATE THRU DECIMAL-LOCATE-END
065500       MOVE ACTIVITY TO COST-CHECK.
065600      IF START-FILE = 'SECTION 2 - COLUMNS' MOVE 'OFF' TO
065700       SWITCH-D GO TO DATA-READ.
065800      IF SWITCH-D = ' ON' GO TO DATA-READ.
065900      IF IXY-CODE = 'X' GO TO PROD-UTIL-ROUT.
066000      IF (IXY-CODE = 'Y' OR IXY-CODE = 'H') AND INV-CODE IS NOT
066100       EQUAL TO PREV-INV-CODE MOVE SPACES TO PREV-ITEM.
066200      IF IXY-CODE = 'Y' OR IXY-CODE = 'H' NEXT SENTENCE ELSE GO
066300       TO DATA-READ.
066310* INV-ROUT THRU JUMP ACCUMULATES ASSETS-ON-HAND AND ASSETS-TO-BUY
066320* AND STORES THEM IN TABLES.
066400 INV-ROUT.
066500      PERFORM DECIMAL-LOCATE THRU DECIMAL-LOCATE-END.
066600      IF DEC-COUNT = 1 AND ZERO-COUNT = 12 GO TO DATA-READ.
066700      IF DEC-COUNT = 0 AND ZERO-COUNT = 13 GO TO DATA-READ.
066800      IF ITEM-CODE = PREV-ITEM GO TO JUMP.
066900      IF INV-CODE = '2' ADD 1 TO ITEM-CTR2 MOVE ITEM-CODE TO
067000       ITEM2 (ITEM-CTR2) ELSE ADD 1 TO ITEM-CTR1 MOVE
067100       ITEM-CODE TO ITEM1 (ITEM-CTR1).
067200      PERFORM ITEM-TABLE-SEARCH-Y THRU END-SEARCH-Y.
067300 JUMP.
067400      MOVE INV-CODE TO PREV-INV-CODE.
067500      SUBTRACT 10 FROM MONTH-USE GIVING MONTH-USE.
067600      IF MONTH-USE = 0 OR MONTH-USE < 0 MOVE 1 TO MONTH-USE.
067700      IF INV-CODE = '2' AND ITEM-MADE = PRODUCT (1) AND MONTH-USE =
067800       '11' AND ACT-SIGN = '-' SUBTRACT ACTIVITY FROM 0 GIVING
067900       ACTIVITY.
068000      IF INV-CODE = '2' ADD ACTIVITY TO
068100       QTY-ON-HAND (ITEM-CTR2, MONTH-USE) ROUNDED
068200       ADD ACTIVITY TO TOTAL-ASSETS-ON-HAND (ITEM-CTR2) ROUNDED
```

```
068300       ELSE
068400       ADD ACTIVITY TO QTY (ITEM-CTR1, MONTH-USE) ROUNDED
068500       ADD ACTIVITY TO TOTAL-ASSETS-TO-BUY (ITEM-CTR1) ROUNDED.
068600      MOVE ITEM-CODE TO PREV-ITEM.
068700      GO TO DATA-READ.
068710* THE PROD-UTIL-ROUT PARAGRAPH ACCUMULATES THE MONTHLY PRODUCT-
068720* ION AND THE MONTHLY UTILIZATION BY PLANT AND STORES THE DATA
068730*  IN TABLES.
068800 PROD-UTIL-ROUT.
068900      PERFORM DECIMAL-LOCATE THRU DECIMAL-LOCATE-END.
069000      IF DEC-COUNT = 1 AND ZERO-COUNT = 12 GO TO DATA-READ.
069100      IF DEC-COUNT = 0 AND ZERO-COUNT = 13 GO TO DATA-READ.
069200      IF PLANT-CODE IS NOT EQUAL TO PREV-PLANT ADD 1 TO PLANT-CTR
069300        MOVE PLANT-CODE TO PLANT (PLANT-CTR).
069400      PERFORM ITEM-TABLE-SEARCH-X THRU END-SEARCH-X.
069500      SUBTRACT 10 FROM PROD-MONTH GIVING PROD-MONTH.
069600      SUBTRACT 10 FROM UTIL-MONTH GIVING UTIL-MONTH.
069700      ADD ACTIVITY TO ELEM (PLANT-CTR, PROD-MONTH, 1) ROUNDED.
069800      ADD ACTIVITY TO ELEM (PLANT-CTR, UTIL-MONTH, 2) ROUNDED.
069900      ADD ACTIVITY TO TOTAL-UTIL (PLANT-CTR) ROUNDED.
070000      ADD ACTIVITY TO TOTAL-PROD (PLANT-CTR) ROUNDED.
070100      MOVE PLANT-CODE TO PREV-PLANT.
070200      GO TO DATA-READ.
070300 ITEM-TABLE-SEARCH-Y.
070400      MOVE ZERO TO CTR3.
070410* ITEM-TABLE-ADD-Y STORES THE CODES FOR ITEMS REPRESENTED BY THE
070420* 'Y' VARIABLES IN THE ITEM-TABLE.
070500 ITEM-TABLE-ADD-Y.
070600      IF ITEM-CTR = 0 ADD 1 TO ITEM-CTR MOVE ITEM-CODE TO
070700        ITEM (ITEM-CTR) GO TO END-SEARCH-Y.
070800      ADD 1 TO CTR3.
070900      IF ITEM-CODE = ITEM (CTR3) GO TO END-SEARCH-Y.
071000      IF CTR3 = ITEM-CTR ADD 1 TO ITEM-CTR MOVE ITEM-CODE TO
071100        ITEM (ITEM-CTR) GO TO END-SEARCH-Y.
071200      GO TO ITEM-TABLE-ADD-Y.
071300 END-SEARCH-Y.
```

```
071400      EXIT.
071410* ITEM-TABLE-SEARCH-X STORES THE CODES FOR ITEMS REPRESENTED BY
071420* THE X VARIABLES IN THE ITEM-TABLE.
071500 ITEM-TABLE-SEARCH-X.
071600      MOVE ZERO TO CTR3.
071700 ITEM-TABLE-ADD-X.
071800      IF ITEM-CTR = 0 ADD 1 TO ITEM-CTR MOVE
071900        ITEM-MADE TO ITEM (ITEM-CTR) GO TO END-SEARCH-X.
072000      ADD 1 TO CTR3.
072100      IF ITEM-MADE = ITEM (CTR3) GO TO END-SEARCH-X.
072200      IF CTR3 = ITEM-CTR ADD 1 TO ITEM-CTR MOVE ITEM-MADE TO
072300        ITEM (ITEM-CTR) GO TO END-SEARCH-X.
072400      GO TO ITEM-TABLE-ADD-X.
072500 END-SEARCH-X.
072600      EXIT.
072700 EOJ2.
072800      MOVE PLANT-CTR TO NO-OF-PLANTS.
072900      MOVE ZERO TO ITEM-CTR.
073000      MOVE ZERO TO PLANT-CTR.
073100      MOVE ITEM-CTR1 TO NO-OF-ITEMS1.
073200      MOVE ITEM-CTR2 TO NO-OF-ITEMS2.
073300      MOVE ZERO TO ITEM-CTR1.
073400      MOVE ZERO TO ITEM-CTR2.
073500      MOVE ZERO TO CTR3.
073600      MOVE 0 TO GLOSSARY-CTR.
073610* LOOP1 STARTS THE COMPLETION OF THE ITEM-COST-TABLE.
073700 LOOP1.
073800      ADD 1 TO ITEM-CTR1.
073900      MOVE ZERO TO GLOSSARY-CTR.
074000 LOOP2.
074100      ADD 1 TO GLOSSARY-CTR.
074200      IF GLOSSARY-CTR IS GREATER THAN GLOSSARY-MAX GO TO
074300       ERROR-MSG4.
074400      IF ITEM1 (ITEM-CTR1) IS NOT EQUAL TO
074500        ITEM-SYMBOL (GLOSSARY-CTR) GO TO LOOP2.
074600      MOVE ITEM-UNIT-COST (GLOSSARY-CTR) TO
```

```
074700         A-ITEM-COST (ITEM-CTR1).
074800       MOVE ITEM-NAME (GLOSSARY-CTR) TO
074900         A-ITEM-NAME (ITEM-CTR1).
075000       MOVE TOTAL-ASSETS-TO-BUY (ITEM-CTR1) TO
075100        A-ITEM-ASSETS (ITEM-CTR1).
075200       IF ITEM-CTR1 = NO-OF-ITEMS1 GO TO LOOP5 ELSE GO TO LOOP1.
075210* THE PRINT OF THE REPORT STARTS WITH LOOP 5.
075300 LOOP5.
075400       MOVE SPACES TO IA-PT-BUY-LINE.
075500       MOVE ITEM-NAME (1) TO IA-PT-ITEM.
075600       WRITE PRINT-OUT FROM PAGE-HEADING AFTER ADVANCING NEXT-PAGE.
075700       WRITE PRINT-OUT FROM IA-PT-BUY-HD1 AFTER ADVANCING 1 LINES.
075800       WRITE PRINT-OUT FROM IA-PT-BUY-HD2 AFTER ADVANCING 2 LINES.
075900 LOOP6.
076000       ADD 1 TO SUB1.
076100       MOVE A-ITEM-NAME (SUB1) TO IA-PT-NAME.
076200       MOVE A-ITEM-COST (SUB1) TO IA-PT-COST.
076300       MOVE A-ITEM-ASSETS (SUB1) TO IA-PT-BUY.
076400       COMPUTE TOTAL-ITEM-COST ROUNDED = A-ITEM-COST (SUB1) *
076500        A-ITEM-ASSETS (SUB1) ON SIZE ERROR GO TO ERROR-MSG2.
076600       ADD TOTAL-ITEM-COST TO TOTAL-PLAN-COST.
076700       MOVE TOTAL-ITEM-COST TO TOTAL-IA-PT-ITEM-COST.
076800       WRITE PRINT-OUT FROM IA-PT-BUY-LINE AFTER ADVANCING 2 LINES.
076900       IF SUB1 IS NOT EQUAL TO NO-OF-ITEMS1 GO TO LOOP6 ELSE MOVE
077000         TOTAL-PLAN-COST TO IA-PT-TOTAL-COST-OUT WRITE PRINT-OUT
077100         FROM IA-PT-TOTAL-PLAN-COST AFTER ADVANCING 3 LINES
077200         MOVE 0 TO SUB1.
077300       MOVE ITEM-NAME (1) TO IPM-HD1-ITEM.
077400       MOVE SPACES TO PRINT-OUT.
077500        WRITE PRINT-OUT AFTER ADVANCING 3 LINES.
077600       WRITE PRINT-OUT FROM IPM-HD1 AFTER ADVANCING 3 LINES.
077700       WRITE PRINT-OUT FROM IPM-HD4 AFTER ADVANCING 2 LINES.
077800       MOVE 0 TO PLANT-CTR.
077900       PERFORM IPM-SEARCH THRU IPM-EXIT VARYING
078000         PLANT-CTR FROM 1 BY 1 UNTIL PLANT-CTR IS GREATER THAN
078100         NO-OF-PLANTS.
```

```
078200      MOVE IPM-TOTAL-COST TO IPM-AMT.
078300      WRITE PRINT-OUT FROM IPM-COST-LINE AFTER ADVANCING 3 LINES.
078400      ADD TOTAL-PLAN-COST, IPM-TOTAL-COST GIVING GRAND-TOTAL.
078500      MOVE GRAND-TOTAL TO GRAND-AMT.
078600      WRITE PRINT-OUT FROM GRAND-TOTAL-LINE AFTER ADVANCING
078700         3 LINES.
078800      IF (GRAND-TOTAL - COST-CHECK > 0 AND GRAND-TOTAL - COST-CHECK
078900       < 10) OR (GRAND-TOTAL - COST-CHECK < 0 AND GRAND-TOTAL -
079000       COST-CHECK > -10) NEXT SENTENCE ELSE WRITE PRINT-OUT FROM
079100       ERR-MESS3 AFTER ADVANCING 3 LINES.
079200      CLOSE DATA-IN, OUTPUT-FILE.
079300      STOP RUN.
079310* IPM-SEARCH MOVES INFO FROM THE GLOSSARY-TABLE TO PRINT LINE.
079400 IPM-SEARCH.
079500      MOVE SPACES TO IPM-LINE.
079600      MOVE SPACES TO IPM-LINE2.
079700      MOVE ZERO TO GLOSSARY-CTR.
079800 IPM-GLOSS-SEARCH.
079900      IF GLOSSARY-CTR = GLOSSARY-MAX GO TO ERROR-MSG1.
080000      ADD 1 TO GLOSSARY-CTR.
080100      IF PLANT (PLANT-CTR) = PLANT-SYMBOL (GLOSSARY-CTR) NEXT
080200        SENTENCE ELSE GO TO IPM-GLOSS-SEARCH.
080300      IF IPM-CODE (GLOSSARY-CTR) = 1 GO TO IPM-EXIT.
080400      MOVE PLANT-NAME1 (GLOSSARY-CTR) TO PLANT-T.
080500      MOVE PLANT-NAME2 (GLOSSARY-CTR) TO PLANT-T2.
080600      MOVE ITEM-NAME1 (GLOSSARY-CTR) TO PRODUCT-T.
080700      MOVE ITEM-NAME2 (GLOSSARY-CTR) TO PRODUCT-T2.
080800      MOVE ONE-TIME-COST (GLOSSARY-CTR) TO OTC.
080900      MOVE ANNUAL-COST (GLOSSARY-CTR) TO AC.
081000      COMPUTE IPM-COST = ONE-TIME-COST (GLOSSARY-CTR) +
081100       5 * ANNUAL-COST (GLOSSARY-CTR) ON SIZE ERROR GO TO
081200      ERROR-MSG2.
081300      COMPUTE IPM-TOTAL-COST = IPM-TOTAL-COST + IPM-COST ON SIZE
081400       ERROR GO TO ERROR-MSG2.
081500      MOVE IPM-COST TO IPM-TC.
081600      MOVE IPM-DESCRIPTION1 (GLOSSARY-CTR) TO IPM-NOTE1.
```

```
081700      MOVE IPM-DESCRIPTION2 (GLOSSARY-CTR) TO IPM-NOTE2.
081800      WRITE PRINT-OUT FROM IPM-LINE AFTER ADVANCING 2 LINES.
081900      WRITE PRINT-OUT FROM IPM-LINE2 AFTER ADVANCING 1 LINES.
082000      MOVE SPACES TO IPM-LINE.
082100      MOVE SPACES TO IPM-LINE2.
082200 IPM-EXIT.
082300      EXIT.
082310* THE FOLLOWING PARA NUMBERS THE TABLES.
082400 TABLE-NO-ROUT.
082500      ADD 1 TO TABLE-CTR.
082600      MOVE TABLE-CTR TO TABLE-NO.
082700      WRITE PRINT-OUT FROM TABLE-NUMBER-LINE AFTER ADVANCING
082800        3 LINES.
082810* DECIMAL-LOCATE THRU DECIMAL-LOCATE-END LOCATES A VARIABLE
082820* DECIMAL POINT WITHIN A FIELD.
082900 DECIMAL-LOCATE.
083000      EXAMINE INPUT-REC REPLACING ALL ' ' BY '0'.
083100      EXAMINE INPUT-REC TALLYING ALL '.'.
083200      MOVE TALLY TO DEC-COUNT.
083300      EXAMINE INPUT-REC TALLYING ALL '0'.
083400      MOVE TALLY TO ZERO-COUNT.
083500      IF DEC-COUNT = 1 AND ZERO-COUNT = 12 GO TO
083600        DECIMAL-LOCATE-END.
083700      IF DEC-COUNT = 0 AND ZERO-COUNT = 13 GO TO
083800        DECIMAL-LOCATE-END.
083900      EXAMINE INPUT-REC TALLYING UNTIL FIRST '.'.
084000 DECIMAL1.
084100      ADD 1 TO SUB4.
084200 DECIMAL2.
084300      ADD 1 TO SUB5.
084400      ADD SUB5, 6 GIVING DEC-CHK.
084500      IF DEC-CHK = TALLY AND TALLY < 13 GO TO DECIMAL2.
084600      MOVE ACT (SUB5) TO ACT-NO (SUB4).
084700      IF SUB4 < 5 GO TO DECIMAL1.
084800      IF SUB4 = 5 AND TALLY < 13 MOVE 0 TO ACT-NO (6)
084900        GO TO DECIMAL3.
```

```
085000      IF SUB4 = 5 AND TALLY = 13 MOVE ACT (6) TO ACT-NO (6).
085100 DECIMAL3.
085200      MOVE ACT-INT TO ACT-INT-NO.
085300      IF TALLY = 7 MOVE ACTIVITY1 TO ACTIVITY.
085400      IF TALLY = 8 MOVE ACTIVITY2 TO ACTIVITY.
085500      IF TALLY = 9 MOVE ACTIVITY3 TO ACTIVITY.
085600      IF TALLY = 10 MOVE ACTIVITY4 TO ACTIVITY.
085700      IF TALLY = 11 MOVE ACTIVITY5 TO ACTIVITY.
085800      IF TALLY = 12 MOVE ACTIVITY6 TO ACTIVITY.
085900      IF TALLY = 13 MOVE ACTIVITY7 TO ACTIVITY.
086000      MOVE 0 TO SUB4.
086100      MOVE 0 TO SUB5.
086200      MOVE 0 TO DEC-CHK.
086300 DECIMAL-LOCATE-END.
086400      EXIT.
086500 ITEM-1-WRITE.
086600      MOVE ITEM1 (CTR3) TO ITEM1-OUT.
086700      WRITE PRINT-OUT FROM ITEM1-CHECK AFTER ADVANCING 2 LINES.
086800 GLOSSARY-WRITE.
086900      MOVE GLIST (GLOSSARY-CTR) TO GLOSS-OUT.
087000      WRITE PRINT-OUT FROM GLOSSARY-CHECK AFTER ADVANCING 2 LINES.
087010* ERROR-MSG1 INDICATES AN ERROR IN PRODUCT-SEARCH1.
087100 ERROR-MSG1.
087200      WRITE PRINT-OUT FROM ERR-MESS1 AFTER ADVANCING 2 LINES.
087300      STOP RUN.
087310* ERROR-MSG2 INDICATES THE RECEIVING FIELD IS TOO SMALL.
087400 ERROR-MSG2.
087500      WRITE PRINT-OUT FROM ERR-MESS2 AFTER ADVANCING 2 LINES.
087600      STOP RUN.
087610* ERROR-MSG4 INDICATES AN ERROR IN LOOP2.
087700 ERROR-MSG4.
087800      WRITE PRINT-OUT FROM ERR-MESS4 AFTER ADVANCING 2 LINES.
087900      STOP RUN.
```


7. THE FIRST SET OF JCL IS USED TO INITIATE A PROBLEM BY

GENERATING A MATRIX AND STARTING INTO THE SOLUTION PROCESS:

```
//IAPT1B PROC                                                              00000100
//MPSEXEC EXEC PGM=DJLEXEC,REGION=300K,PARM=TASK,DPRTY=(13,13),            00001000
//    TIME=150                                                             00001050
//STEPLIB DD DSN=MMART.IAPT.LOAD,DISP=SHR                                  00001070
//SCRATCH1  DD UNIT=ITEL,SPACE=(CYL,(10),,CONTIG)                          00001100
//SCRATCH2  DD UNIT=(ITEL,SEP=SCRATCH1),SPACE=(CYL,(10),,CONTIG)           00001200
//PROBFILE  DD UNIT=ITEL,SPACE=(CYL,(10),,CONTIG)                          00001300
//ETA1      DD UNIT=ITEL,SPACE=(CYL,(10),,CONTIG)                          00001500
//ETA2      DD UNIT=(ITEL,SEP=ETA1),SPACE=(CYL,(10),,CONTIG)               00001600
//MATRIX1   DD UNIT=ITEL,SPACE=(CYL,(10),,CONTIG)                          00001700
//MATRIX2   DD UNIT=(ITEL,SEP=MATRIX1),SPACE=(CYL,(10),,CONTIG)            00001800
//MIXWORK   DD UNIT=ITEL,SPACE=(CYL,(10),,CONTIG)                          00001900
//FT05F001  DD DDNAME=SYSIN                                                00002200
//FT06F001 DD SYSOUT=A                                                     00002300
//FT08F001 DD UNIT=ITEL,SPACE=(CYL,(10,10))                                00002350
//SYSMLCP   DD  DSN=MMART.MLCPF.LOAD(START),DISP=SHR                       00002400
//SYSPRINT  DD  SYSOUT=A                                                   00002500
//SYSPUNCH  DD  SYSOUT=B                                                   00002600
```

8. THE SECOND SET OF JCL IS USED TO RESUME A PROBLEM THAT HAS NOT FOUND A CONTINUOUS OPTIMAL SOLUTION BY RESTORING AND PROCEEDING WITH THE SOLUTION PROCESS:

```
//IAPT2B PROC                                                              00000100
//MPSEXEC EXEC PGM=DJLEXEC,REGION=300K,PARM=TASK,DPRTY=(13,13),            00001000
//    TIME=150                                                             00001050
//STEPLIB DD DSN=MMART.IAPT.LOAD,DISP=SHR                                  00001100
//SCRATCH1  DD UNIT=ITEL,SPACE=(CYL,(10),,CONTIG)                          00001150
```

```
//SCRATCH2  DD UNIT=(ITEL,SEP=SCRATCH1),SPACE=(CYL,(10),,CONTIG)          00001200
//OLDPFILE  DD UNIT=ITEL,SPACE=(CYL,(10),,CONTIG)                        00001300
//PROBFILE  DD UNIT=ITEL,SPACE=(CYL,(10),,CONTIG)                        00001350
//ETA1      DD UNIT=ITEL,SPACE=(CYL,(10),,CONTIG)                        00001500
//ETA2      DD UNIT=(ITEL,SEP=ETA1),SPACE=(CYL,(10),,CONTIG)             00001600
//MATRIX1   DD UNIT=ITEL,SPACE=(CYL,(10),,CONTIG)                        00001700
//MATRIX2   DD UNIT=(ITEL,SEP=MATRIX1),SPACE=(CYL,(10),,CONTIG)          00001800
//MIXWORK   DD UNIT=ITEL,SPACE=(CYL,(10),,CONTIG)                        00001900
//FT06F001 DD SYSOUT=A                                                   00002300
//SYSMLCP   DD  DSN=MMART.MLCPF.LOAD(RESTCONT),DISP=SHR                  00002400
//SYSPRINT  DD  SYSOUT=A                                                 00002500
//SYSPUNCH  DD  SYSOUT=B                                                 00002600
```

9. THE THIRD SET OF JCL IS USED TO RESUME A PROBLEM THAT HAS FOUND A CONTINUOUS OPTIMUM BUT NOT AN INTEGER OPTIMUM BY RESTORING AND PROCEEDING WITH THE SOLUTION PROCESS.


```
//IAPT3B PROC                                                            00000100
//MPSEXEC EXEC PGM=DJLEXEC,REGION=300K,PARM=TASK,DPRTY=(13,13),          00001000
//    TIME=150                                                           00001050
//STEPLIB DD DSN=MMART.IAPT.LOAD,DISP=SHR                                00001100
//SCRATCH1  DD UNIT=ITEL,SPACE=(CYL,(10),,CONTIG)                        00001150
//SCRATCH2  DD UNIT=(ITEL,SEP=SCRATCH1),SPACE=(CYL,(10),,CONTIG)         00001200
//OLDPFILE  DD UNIT=ITEL,SPACE=(CYL,(10),,CONTIG)                        00001300
//PROBFILE  DD UNIT=ITEL,SPACE=(CYL,(10),,CONTIG)                        00001350
//ETA1      DD UNIT=ITEL,SPACE=(CYL,(10),,CONTIG)                        00001500
//ETA2      DD UNIT=(ITEL,SEP=ETA1),SPACE=(CYL,(10),,CONTIG)             00001600
//MATRIX1   DD UNIT=ITEL,SPACE=(CYL,(10),,CONTIG)                        00001700
//MATRIX2   DD UNIT=(ITEL,SEP=MATRIX1),SPACE=(CYL,(10),,CONTIG)          00001800
//MIXWORK   DD UNIT=ITEL,SPACE=(CYL,(10),,CONTIG)                        00001900
//FT06F001 DD SYSOUT=A                                                   00002300
//SYSMLCP   DD  DSN=MMART.MLCPF.LOAD(RESTMIP),DISP=SHR                   00002400
```

```
//SYSPRINT  DD  SYSOUT=A                                              00002500
//SYSPUNCH  DD  SYSOUT=B                                              00002600
```

10. THE FOURTH SET OF JCL IS USED TO EXTRACT A SOLUTION AND GENERATE A REPORT.

```
//IAPT4B PROC                                                         00000100
//MPSEXEC EXEC PGM=DJLEXEC,REGION=300K,PARM=TASK,DPRTY=(13,13),       00001000
//   TIME=150                                                         00001050
//SCRATCH1  DD UNIT=ITEL,SPACE=(CYL,(10),,CONTIG)                     00001100
//STEPLIB DD DSN=MMART.IAPT.LOAD,DISP=SHR                             00001150
//SCRATCH2  DD UNIT=(ITEL,SEP=SCRATCH1),SPACE=(CYL,(10),,CONTIG)      00001200
//PROBFILE  DD UNIT=ITEL,SPACE=(CYL,(10),,CONTIG)                     00001300
//ETA1      DD UNIT=ITEL,SPACE=(CYL,(10),,CONTIG)                     00001500
//ETA2      DD UNIT=(ITEL,SEP=ETA1),SPACE=(CYL,(10),,CONTIG)          00001600
//MATRIX1   DD UNIT=ITEL,SPACE=(CYL,(10),,CONTIG)                     00001700
//MATRIX2   DD UNIT=(ITEL,SEP=MATRIX1),SPACE=(CYL,(10),,CONTIG)       00001800
//MIXWORK   DD UNIT=ITEL,SPACE=(CYL,(10),,CONTIG)                     00001900
//FT05F001  DD DDNAME=SYSIN                                           00002200
//FT06F001 DD SYSOUT=A                                                00002300
//FT08F001 DD UNIT=ITEL,SPACE=(CYL,(10,10))                           00002350
//SYSMLCP   DD  DSN=MMART.MLCPF.LOAD(INSRT),DISP=SHR                  00002400
//SYSPRINT  DD  SYSOUT=A                                              00002500
//SYSPUNCH  DD  DUMMY                                                 00002600
```

11. THE FOLLOWING FORTRAN SUBROUTINE CHECKS THE DAYS OF THE WEEK AND THE TIME OF DAY THE JOB STARTS. IF IT IS BETWEEN 0700 AND 1600 ON MONDAY THROUGH FRIDAY IT WILL SET XDELTM AT 9 MIN, OTHERWISE IT WILL SET XDELTM AT 60 MIN. THE VALUE OF XDELTM

DETERMINES AT WHAT TIME THE SOLUTION WILL BE SAVED.

```
      INTEGER*2 HOL(15), DAREA(19)                                          00000100
      DATA HOL/2H70, 2H71, 2H72, 2HSA, 2HSU, 2H00, 2H10, 2H20, 2H30,        00000200
     1 2H40, 2H50, 2H60, 2H70, 2H80, 2H90/                                  00000300
      CALL AS9AOD(DAREA)                                                    00000400
      M=9                                                                   00000500
      IF(DAREA(18).EQ.HOL(4)) GO TO 10                                      00000600
      IF(DAREA(18).EQ.HOL(5)) GO TO 10                                      00000700
      DO 3 N=1,3                                                            00000800
    3 IF (DAREA(14) .EQ. HOL(N)) J=N-1                                      00000900
      DO 5 N=6,15                                                           00001000
    5 IF(DAREA(15) .GE. HOL(N)) K=N-6                                       00001100
      I=10*J+K                                                              00001200
      IF(I.LT. 7) GO TO 10                                                  00001300
      IF(I.GE.16) GO TO 10                                                  00001400
      GO TO 15                                                              00001500
   10 M=120                                                                 00001600
   15 CALL PUTARG(M)                                                        00001700
      RETURN                                                                00001800
      END                                                                   00001900
```

APPENDIX B

COMPARATIVE RUNNING TIME

COMPARATIVE RUNNING TIME

The IA/PT model may require a large amount of CPU time for items with many components and IPMs. Table B-1 relates problem size to run time. This table is provided to assist in estimating time to solution. The most significant variable that affects running time is the size of the problem as measured by the number of alternative solutions. The run time estimate is made as follows:

(1) List the number of alternatives put in for each line (including the current capability).

(2) Sum these alternatives and compare with the first column (number of alternatives) in the table.

(3) Multiply the numbers of alternatives for each line and compare them with the second column (alternative solutions).

(4) The third and fourth columns give observed run times for Modes 1 and 2, respectively. These numbers should not be used as absolutes but as general guidelines. The fourth item of the fourth column (3.3 min) is for a solution for RR=0.0; if there were assets, run time would be expected to be considerably longer. Table B-2 provides further information about the mixed integer problems for which the run times are shown.

TABLE B-1 RUN TIME AS A FUNCTION OF PROBLEM SIZE

No. of Alternatives	Alternative Solutions	CP Run Time (Min) Mode 1	Mode 2
19	8	19.3	5.1
21	256	35.4	7.6
25	1024	57.1	13.2
31	4608	105.2	3.3
38	4096	98.9	255.2
43	139968	534.4	209.8

TABLE B-2. PROBLEM MODEL SIZE

Item	Rows	Columns	Inter Variables	Potential Solutions
81mm M374A2	521	3802	19	8
CBU-58B FY77	502	3304	21	256
CBU-58B FY76	596	3342	25	1024
105mm HE, M1	818	6945	31	4608
155mm M07	945	7664	38	4096
MK82 Bomb	1104	9563	43	139968