INDUSTRY COMMISSION

Procedures for Later Stage Processing of Single-region Input-Output Data for SALTER

by

Wayne Calder, Robert McDougall

and

Alexandra Strzelecki

SALTER Working Paper No. 18

JUNE 1993

SALTER working papers document work in progress on the development of the SALTER model of the world economy. They are made available to allow scrutiny of the work undertaken but should not be quoted without the permission of the author(s). Comments on the papers would be most welcome.

IC
339.5
SAL 18

INDUSTRY COMMISSION

Procedures for Later Stage Processing of Single-region Input-Output Data for SALTER

by

Wayne Calder, Robert McDougall

and

Alexandra Strzelecki

SALTER Working Paper No. 18

Industry Commission
9 JUN 1995
LIBRARY

JUNE 1993

SALTER working papers document work in progress on the development of the SALTER model of the world economy. They are made available to allow scrutiny of the work undertaken but should not be quoted without the permission of the author(s). Comments on the papers would be most welcome.

CONTENTS

1 Introduction 1

2 Preprocessing the input-output tables to eliminate zero trade flows: motivation 2

3 Preprocessing the input-output tables to eliminate zero trade flows: approach 3

4 Programs and file structures 5

File structure 6
Programs 9

5 Procedure and implementation 13

Appendixes

A Source code for program SCALE 17
B Source code for program SUMDB 31
C Source code for program EXTRACT 57
D Source code for program AVERAGE 69
E Source code for program PROJECT 95

References 123

Figure

1 Flow chart for preprocessing of single-region input-output tables 15

Tables

1 Header arrays for single-region input-output tables 7
2 Header arrays for single-region input-output coefficient files 8
3 Header arrays for the region-generic data file 8
4 Contents of the region-generic data file for SALTER 9

PROCEDURES FOR LATER STAGE PROCESSING OF SINGLE-REGION INPUT-OUTPUT DATA FOR SALTER

This working paper describes some of the later stage processing procedures for the single-region input-output tables of the SALTER database.

1 Introduction

SALTER is a multi-region multi-sector model of the world economy first documented in Jomini *et al.* 1991. It is designed as an instrument of economic policy analysis, in particular, for analysing trade and industry policies in an international context.

The SALTER database contains a variety of data, with multi-region input-output data comprising the bulk of it. The database is assembled from several smaller files, including a set of single-region input-output tables and an international trade database.

Preparing the SALTER database is a three-stage process involving data collection, adjustment, and assembly. In the first stage, we collect statistics from various sources, arrange them into standard file structures, and transform them into standard sectoral classifications. In the second stage, we adjust the data to eliminate inconsistencies between sources. In the third stage, we assemble the adjusted data to form the SALTER multi-region database.

This paper covers some of the procedures used in processing the single-region input-output tables in the second stage of database preparation. We call this second stage the *later stage processing* of the single-region input-output tables. The expression 'later stage processing' distinguishes the adjustment procedures in the second stage of database preparation from the initial construction or *early stage processing* in the first stage.

The main purpose of the later stage processing of the single-region input-output tables is to adjust the tables to other data used in the SALTER database. These other data include international trade data (Hambley 1993, Hanslow 1993),

assistance data (Gotch 1993) and macroeconomic data (Brown, Strzelecki and Watts forthcoming). The bulk of this work is done by FIT, an input-output update facility documented in James and McDougall (1993). However, some preprocessing procedures need to be undertaken before FIT can be used. This paper documents the necessary preprocessing procedures.

2 Preprocessing the input-output tables to eliminate zero trade flows: motivation

One of the purposes of the FIT program is to adjust the trade values in a single region input-output table to agree with independently derived international trade data. It achieves this, along with other objectives, using input-output modelling techniques. Indeed, the FIT program is nothing more than a kind of single-region input-output model implemented using GEMPACK (Codsi and Pearson, 1988, Pearson and Codsi 1991). Any one of the single-region input-output tables can act as the model's database. The required adjustments to the input-output table can be represented as shocks to the model. The revised input-output table can then be generated by a FIT simulation as an updated database. But before FIT can be used, zero trade flows need to be eliminated from a FIT database.

There are two reasons for eliminating zero trade flows before running FIT. One is that FIT, like most GEMPACK-based models, is expressed in percentage change form. With the percentage change formulation, all zero values in the original database remain zero in the updated database, no matter what shocks are applied to the model. So for any commodity for which the international trade database specifies non-zero trade values, we must have non-zero trade values in the FIT input database.

This reason for eliminating zero trade flows is valid but not decisive: an alternative solution is available, namely rewriting FIT to avoid the percentage change formulation in trade flows. But the second reason is decisive. It relates to the import sales disposition in the updated database.

If the target value from the trade database for aggregate imports of a commodity is positive, then the updated input-output table must contain a complete import sales disposition for that commodity. This sales disposition will be based on the

sales disposition in the initial database. But if aggregate imports of the commodity in the initial database are zero, then the initial table contains no information about the sales disposition, and a revised sales disposition cannot be constructed.

Accordingly, before inputting the single-region input-output tables into the FIT update simulations, we pre-process them to eliminate zero flows for exports and imports. For commodities that originally had zero import flows, we want the new import sales dispositions to be sensible. Finally, we want the new positive trade flows to be small, to minimise distortions to the original input-output data.

3 Preprocessing the input-output tables to eliminate zero trade flows: approach

We begin with the question, how to construct sensible import sales dispositions in preprocessing the single-region input-output tables.

The sales disposition of an imported commodity in a region depends on two main factors: the intensity with which it is used in each economic activity, and the profile of economic activity in the region. If the database for one region originally shows zero usage in all activities, we look to the other regions in the SALTER model to see which activities typically use the commodity most intensively. A sensible sales disposition takes account both of these typical usage intensities, and of the profile of economic activity in the region.

More precisely, we can divide the input-output table in each region into three elements: the *indirect tax structure*, the *input-output structure,* and the *final demand profile*. The indirect tax structure consists of commodity tax rates and production tax rates. The input-output structure consists of various expenditure share coefficients. The final demand profile shows aggregate expenditure in each final demand category (for SALTER, household consumption, government consumption, investment, and exports).

The input-output structure in turn divides into two levels. The upper level shows the shares of each commodity in each industry's total costs, and in aggregate expenditure for each final demand category (*commodity shares*). The lower level shows the shares of domestic products and imports in expenditure on each

commodity for each demand category (domestic product shares and *import shares*).

Note that we calculate the commodity shares as shares in total costs *excluding* production tax. This is because we include the production tax rate in the indirect tax structure rather than the input-output structure.

In preprocessing the single-region input-output tables we revise the input-output structure while preserving the indirect tax structure and the final demand profile. To achieve positive import usage, not only the import but also the commodity shares must be positive. To ensure this we revise both the lower and the upper levels of the input-output structure. At the same time we keep changes at both levels small, to minimise distortions to the original input-output data.

In revising the input-output structure for each region in the model, we turn to the other regions for guidance. More precisely, we refer to a *representative* input-output table, constructed by summing the input-output tables over all SALTER regions. To ensure that each region is suitably weighted in the representative table, we first scale all the individual region tables to a common unit of account (millions of US dollars) and a common reference year (1988).

To revise the input-output structure for each region, we calculate new input-output coefficients as weighted averages of coefficients from the original input-output table and the representative table. The weights are arbitrary. We choose a large positive weight for the original individual-region table, and a small positive weight for the representative table.

By assigning a much greater weight to the original individual-region table than to the representative table, we ensure that changes to the input-output structure are small. By assigning a positive weight to the representative table, we ensure that each commodity has positive import usage in each of the revised individual-region tables, provided that it had positive import usage in at least one of the original tables. Likewise we ensure that each commodity has positive export sales in all the revised tables, provided that it had positive exports in at least one of the original tables.

In the original input-output tables for SALTER, (Hambley 1993, Brown *et al.* forthcoming) every commodity in the model has positive import usage in at least

one region, and positive export sales from at least one region. Thus this approach accomplishes the objectives of the preprocessing.

4 Programs and file structures

Several utility programs were developed to undertake the preprocessing and processing.

- SCALE rescales a single-region input-output table.
- SUMDB sums over a collection of input-output tables to generate a new input-output table.
- EXTRACT extracts input-output coefficients from an input-output table.
- AVERAGE calculates weighted averages over two files of input-output coefficients to generate a new coefficients file.
- PROJECT extracts the final demand profile from an input-output table and combines it with independent data for input-output coefficients to generate a new input-output table.
- FIT adjusts an input-output table in accordance with user-supplied information about international trade, domestic final demand, etc.

All these utilities are FORTRAN programs. Some are written by hand, others using the GEMPACK code-writing program TABLO. The TABLO-written FORTRAN programs include both data handling utilities and implementations of economic models. The former merely calculate and write coefficients, while the latter can set up and solve systems of equations and update databases.

Each utility takes user-supplied instructions and one or more data files as input, and writes a data file as output. The user input may be provided either interactively or as a stored input file. All data files are in the header array format developed for GEMPACK data files (Mikkelsen and Pearson 1986).

This section describes these programs, and the files that they handle. Source code listings for the programs are given in the appendixes. The required user input for the programs is described in Brown *et al.* (forthcoming).

File structures

Three kinds of data files are handled by the utility programs: single-region input-output tables, single-region input-output coefficients files, and a region-generic data file. All these files conform to the GEMPACK header array format.

The single-region input-output tables are involved in all three stages of database construction (data collection, adjustment, and assembly). In developing the later stage processing procedure the structure of these files was taken as given; it is shown here for ease of reference. The input-output coefficients files and the region-generic data file are used only in revising the single-region input-output tables in the second stage.

The single-region input-output coefficients file is designed to record the input-output structure of a single-region input-output table. In the input-output structure we include commodity and primary factor shares in each industry's total costs, commodity shares in aggregate expenditure on each category of final demand, and domestic product and import shares in expenditure on each commodity in each demand category. To keep the input-output structure separate from the indirect tax structure, we calculate the industry cost shares as shares in total costs excluding production tax.

The region-generic data file contains some pieces of information which are common to all regions. These include the dimensions of the model, and parameters settings for use with programs AVERAGE and FIT.

Tables 1 to 3 list the headers for each kind of file. Table 4 lists the contents of the region-generic data file, as set up for SALTER.

In Table 4, note that the the number of sectors specified in the region-generic data file must agree with the actual number of sectors in the single-region input-output tables. The weight parameters used with AVERAGE and FIT can be set arbitrarily. The weights need not sum to one; only the relative weights matter.

Table 1: **Header arrays for single-region input-output tables**

Header	*Dimension*	*Description*
AI01	g * g	Intermediate usage of domestic products, by commodity and industry
AI02	g * g	Intermediate usage of imports, by commodity and industry
AI03	g	Investment usage of domestic products, by commodity
AI04	g	Investment usage of imports, by commodity
AI05	g	Household consumption of domestic products, by commodity
AI06	g	Household consumption of imports, by commodity
AI07	g	Government consumption of domestic products, by commodity
AI08	g	Government consumption of imports, by commodity
AI09	g	Change in stocks of domestic products, by commodity
AI10	g	Change in stocks of imports, by commodity
AI11	g	Exports, by commodity
AI12	g	Non-commodity indirect taxes, net, by industry
AI13	g	Employment of labour, by industry
AI14	g	Employment of capital, by industry
AI15	g	Employment of land, by industry
AI16	g * g	Commodity tax on intermediate usage of domestic products, by commodity and industry
AI17	g * g	Commodity tax on intermediate usage of imports, by commodity and industry
AI18	g	Commodity tax on household consumption of domestic products, by commodity
AI19	g	Commodity tax on household consumption of imports, by commodity
AI20	g	Commodity tax on investment usage of domestic products, by commodity
AI21	g	Commodity tax on investment usage of imports, by commodity
AI22	g	Commodity tax on government usage of domestic products, by commodity
AI23	g	Commodity tax on government usage of imports, by commodity
AI24	g	Commodity tax on exports, by commodity
AI25	g	Commodity tax on change in stocks of domestic products, by commodity
AI26	g	Commodity tax on change in stocks of imports, by commodity
AI27	g	Import duty, by commodity

g Number of sectors.

Table 2: **Header arrays for single-region input-output coefficients files**

Header	*Dimension*	*Typical element*	*Description*
H001	g * g	H001(I,J)	Share of commodity I in total costs (excluding production tax) of industry J
H003	g	H003(I)	Share of commodity I in aggregate investment expenditure
H005	g	H005(I)	Share of commodity I in aggregate household consumption expenditure
H007	g	H007(I)	Share of commodity I in aggregate government consumption expenditure
H009	g	H009(I)	Share of commodity I in aggregate border value of exports
H012	g	H012(J)	Share of labour in total costs (excluding production tax) of industry J
H013	g	H013(J)	Share of capital in total costs (excluding production tax) of industry J
H014	g	H014(J)	Share of land in total costs (excluding production tax) of industry J
S001	g * g	S001(I,J)	Share of domestic products in intermediate usage of commodity I by industry J
S002	g * g	S002(I,J)	Share of imports in intermediate usage of commodity I by industry J
S003	g	S003(I)	Share of domestic products in investment usage of commodity I
S004	g	S004(I)	Share of imports in investment usage of commodity I
S005	g	S005(I)	Share of domestic products in household consumption of commodity I
S006	g	S006(I)	Share of imports in household consumption of commodity I
S007	g	S007(I)	Share of domestic products in government consumption of commodity I
S008	g	S008(I)	Share of imports in government consumption of commodity I

g Number of sectors.

Table 3: **Header arrays for the region-generic data file**

Header	*Dimension*	*Element*	*Description*	*Program(s)*
G001	1	1	Number of sectors	All
G002	2	1	Weight attached to the original individual-region input-output structure	AVERAGE
		2	Weight attached to the representative input-output structure	
G003	2	1	Weight attached to information about the upper level of the input-output structure	FIT
		2	Weight attached to information about the lower level of the input-output structure	

Table 4: **Contents of the region-generic data file for SALTER**

Header	*Element*	*Description*	*Value*
G001	1	Number of sectors	37
G002	1	Weight attached to the original individual-region input-output structure	100
	2	Weight attached to the representative input-output structure	1
G003	1	Weight attached to information about the upper level of the input-output structure	10
	2	Weight attached to information about the lower level of the input-output structure	1

Programs

For each program, we list the input and output data files, and describe the required user input (other than file names for input and output data files, which are in all cases user-supplied). We then outline the operation of the program.

For an account of how we combine the programs to perform the later stage processing, see Section 5.

SCALE

SCALE is a TABLO-written FORTRAN program, designed to rescale a single-region input-output table.

SCALE is a data handling routine. It reads a single-region input-output table and the region-generic data file, and writes a new input-output table. The user supplies a target value for gross domestic product (gdp) for the new input-output table, denominated in the required unit of account (millions of US dollars) and for the required reference year (1988). We use SCALE to ensure that each region is suitably weighted in the representative input-output table.

SCALE calculates gdp from the initial input-output table, then the ratio of target to initial gdp. It then multiplies all arrays in the initial input-output table by this ratio, generating a new input-output table in which gdp assumes the target value.

SUMDB

SUMDB is a hand-written FORTRAN program, designed to sum a collection of input-output tables.

SUMDB reads a collection of input-output tables, and writes a new input-output table. The number of tables to be read is specified by the user. We use SUMDB to generate the representative input-output table.

Each element in each array in the new input-output table is calculated as the sum of the corresponding elements in the old tables.

In its current implementation, SUMDB can sum up to fourteen input-output tables at a time. Groups of fifteen or more tables can be summed using multiple applications of SUMDB.

EXTRACT

EXTRACT is a TABLO-written FORTRAN program, designed to extract the input-output structure from an input-output table.

EXTRACT is a data handling routine. It reads an input-output table and the region-generic data file. It writes an input-output coefficients file. We use EXTRACT to extract the input-output structure from the initial individual-region input-output tables and from the representative table.

For regions whose input-output tables containing some zero flows, some of the share coefficients may be undefined. This occurs when the devisor in the share formula is equal to zero. In such cases, EXTRACT uses the TABLO zerodivide default facility to assign the share coefficient a value of zero.

AVERAGE

AVERAGE is a TABLO-written FORTRAN program, designed to calculate averages of input-output coefficients across two input-output coefficients files.

AVERAGE is a data handling routine. It reads two input-output coefficients files and the region-generic data file. It writes an input-output coefficients file. We use AVERAGE to generate revised individual-region input-output structures consistent with non-zero trade flows.

Each element in each array of the new input-output coefficients file is calculated as a weighted average of the corresponding elements in the original coefficients files. Default weights are read from the region-generic data file.

In most cases, AVERAGE uses the default weights to calculate the average input-output coefficients. Exceptions arise where coefficients in the original coefficients files have zero values assigned by EXTRACT using the TABLO zerodivide default facility. In such cases, the underlying input-output table contains no information about the coefficient, and we wish to assign the coefficient a zero weight.

The exceptional cases are handled as follows. Where just one of the original coefficients files contains a zero assigned by the zerodivide default facility, the new value is set equal to the value from the other original file. Where both the original files contain zeros assigned by the zerodivide default facility, the new value is set equal to zero.

To apply this treatment, AVERAGE must distinguish zero values assigned by the zerodivide default facility from zero values calculated as the quotient of a zero dividend by a non-zero divisor. The structure of the coefficients file and the design of EXTRACT allows it to do this. In the coefficients file generated by EXTRACT, each set of share coefficients with a common devisor normally sums to one. But if the devisor is equal to zero, EXTRACT applies the zerodivide default and sets the coefficients to zero. The coefficients then sum to zero. To determine which zero values are zeros assigned by the zerodivide default, AVERAGE calculates sums of coefficients with a common divisor. If the sum is equal to one, none of the coefficients has a zero value assigned by the zerodivide default; if the sum is zero, they all have.

PROJECT

PROJECT is a TABLO-written FORTRAN program designed to revise an input-output table, using a new input-output structure, but preserving the original indirect tax structure and final demand profile.

PROJECT reads a single-region input-output table, an input-output coefficients file, and the region-generic data file. It generates a new input-output table. We

use PROJECT to generate revised individual-region input-output tables with non-zero trade flows.

PROJECT is an implementation of an input-output model. Unlike most TABLO-implemented models, this model is expressed 'in the levels' rather than in change or percentage change form. This is possible because the model is intrinsically linear, rather than a linearisation of a non-linear model. The variables in the model represent input-output flows, and the equations in the model represent input-output accounting identities. The single-region input-output table, the coefficients file, and the region-generic data file provide the database for the model. The TABLO database update facility is used to generate the new input-output table.

As usual with TABLO-implemented models, the user is required to specify the solution procedure, closure and shocks for each simulation.

PROJECT calculates indirect tax rates and the aggregate components of final demand from the original input-output table. It uses these, together with the share coefficients from the input-output coefficients file, as coefficients in an equation system. The variables in the equation system include variables corresponding to all the flows in a SALTER single-region input-output table. Thus the solution of the system defines a new input-output table, which the program writes to an output file using the update facility.

Since the model implemented by PROJECT is linear, the user should always specify the linear (one-stage, single-step) solution procedure.

Besides variables representing input-output flows, PROJECT contains a dummy variable, named 'one'. In PROJECT simulations, the user should specify the variable 'one' as exogenous, and apply to it a shock of one. All other variables in the model should be specified as endogenous.

FIT

FIT is a TABLO-written FORTRAN program, designed to adjust an input-output table to external information about final demand, external trade, and the indirect tax structure.

FIT implements an augmented input-output model. It reads a single-region input-output table and the region-generic data file. Taking these as the initial database, it performs a simulation and uses the database update facility to write a new input-output table. The user is required to specify the solution procedure, closure and shocks for each simulation.

A detailed description of FIT is provided in James and McDougall (1993).

5 Procedure and implementation

The procedure for the later stage processing involves assembling the necessary data files and programs, preprocessing to remove zero trade flows, the FIT update simulations, and checking the updated input-output tables.

Later stage processing can be performed on microcomputers running under the Microsoft MS-DOS operating system.

Files used in the later stage processing include the programs described above, various utilities for checking the updated tables, a DOS batch file described below, initial single-region input-output tables from the first stage of database preparation, and miscellaneous data files including the region-generic data file described above and files containing data for the update targets.

For a single-region database to be valid it must satisfy certain sign and balance conditions. The sign conditions require various input-output flows to be non-negative. The balance conditions require equality between total costs and total sales of each industry.

For the later stage processing to be successful the initial input-output tables must satisfy the sign conditions. Checks on these conditions should therefore be included in the first stage of database preparation. We apply these checks using a utility called CHECKNEG. CHECKNEG is a TABLO-generated FORTRAN program (Brown *et al.* forthcoming)

While obviously desirable, it is not necessary for the later stage processing that the initial tables satisfy the balance conditions. This is because the preprocessing automatically eliminates any initial imbalances. Specifically, the program PROJECT always generates a balanced database. Indeed, PROJECT

can be used independently in conjunction with EXTRACT to balance an unbalanced input-output table.

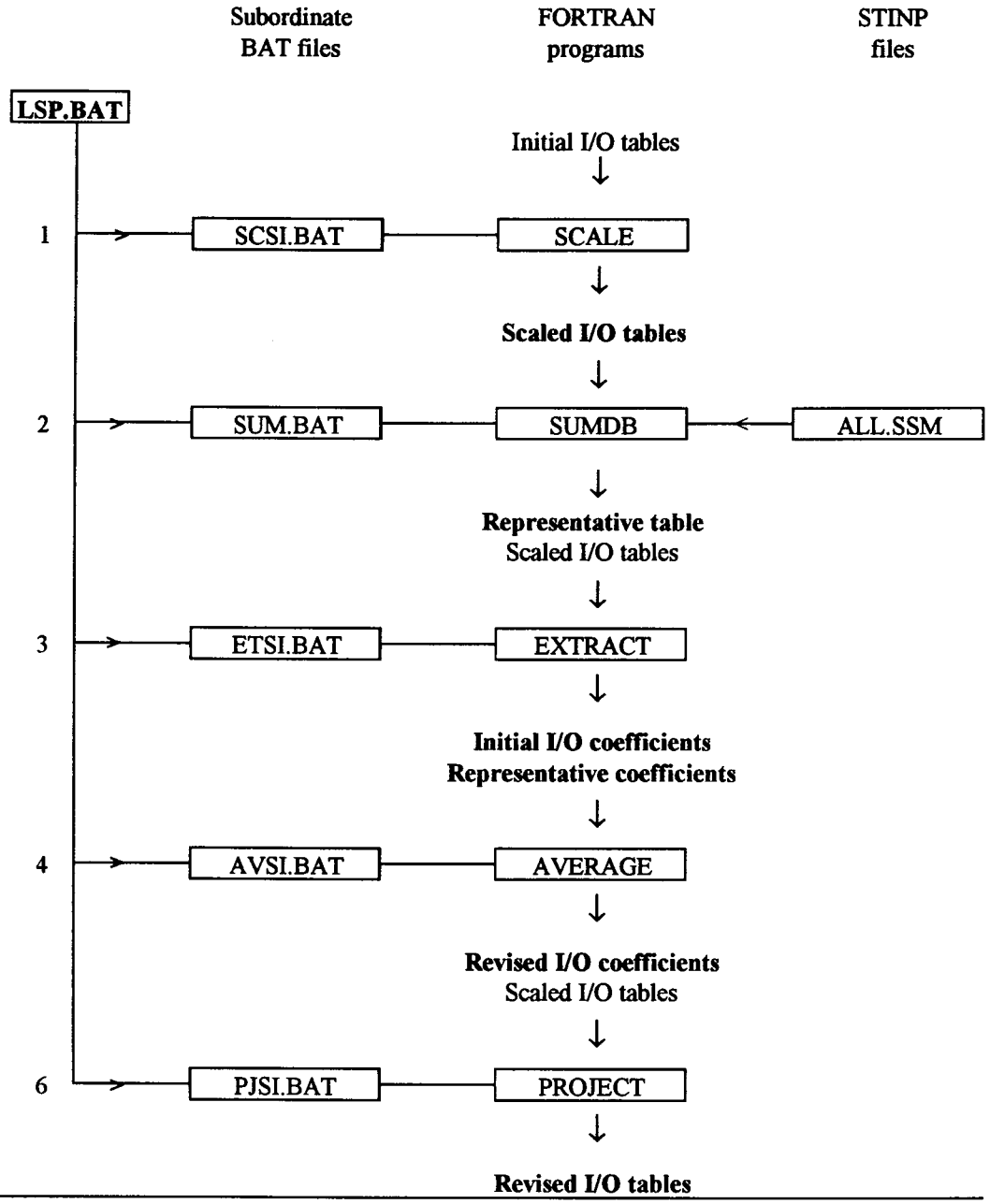
Thus the steps involved in the preprocessing are as follows:

- Using SCALE to rescale the initial input-output tables, applying target values for gdp in 1988, expressed in millions of US dollars.
- Using SUMDB to generate the representative input-output table by summing over the rescaled individual-region input-output tables.
- Using EXTRACT to generate input-output coefficients files from the individual-region input-output tables and the representative table.
- Using AVERAGE to revise the input-output coefficients for the individual regions, by averaging them with the coefficients from the representative table.
- Using PROJECT to revise the individual-region input-output tables, adjusting them to the revised input-output coefficients while preserving unaltered the final demand profile and indirect tax structure.

All these steps can be conducted with a single command by means of a batch file, LSP.BAT. The batch file contains information about file names, and target values for gdp for use with SCALE. It calls several subordinate batch files. The operations performed by the batch file are depicted in Figure 1.

Source code for the programs used in the later stage processing, and batch files for the preprocessing, are stored on diskette in a file LSP.ARC.

FIGURE 1: Flow chart for preprocessing of single-region input-output tables



Output files are shown **bold**; other major input files are shown plain; minor input files are not shown.

APPENDIX A: Source Code For Program SCALE

```
!*****************************************************************************!
!                                    FILES                                    !
!*****************************************************************************!

FILE INHAR # FILE CONTAINING ALL UPDATED I/O INFORMATION # ;
FILE (TEXT) INGDP # FILE CONTAINING GDP DATA # ;
FILE GEN # auxiliary data file for single-region revision procedure #;
FILE (NEW) OUTHAR # FILE CONTAINING SCALED DATABASE # ;


!*****************************************************************************!
!                                    SETS                                     !
!*****************************************************************************!

COEFFICIENT (INTEGER)
 NO_IND # number of sectors #;
READ NO_IND FROM FILE gen HEADER                                    "G001";

SET
 IND # sectors # MAXIMUM SIZE 40 SIZE NO_IND;

!*****************************************************************************!
!                                  DATA BASE                                  !
!*****************************************************************************!


!-----------------------------------------------------------------------------
                          INPUT-OUTPUT DATA BASE
 -----------------------------------------------------------------------------!
```

```
COEFFICIENT                                                    GDP1
! CURRENT VALUE OF GDP ! ;
READ      GDP1        FROM                                        TERMINAL ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                             DINT(I,J)
! INTERMEDIATE USAGE (DOMESTIC) !     ;
READ (ALL,I,IND)(ALL,J,IND) DINT(I,J)
          FROM FILE  INHAR  HEADER                                  "AI01" ;


COEFFICIENT (ALL,I,IND)(ALL,J,IND)                             IINTS(I,J)
! INTERMEDIATE USAGE (BY IMPORTING SECTOR) ! ;
READ (ALL,I,IND)(ALL,J,IND) IINTS(I,J)
          FROM FILE  INHAR  HEADER                                  "AI02" ;


COEFFICIENT (ALL,I,IND)                                        DINV(I)
! INVESTMENT USAGE (DOMESTIC) ! ;
READ (ALL,I,IND) DINV(I)
          FROM FILE  INHAR  HEADER                                  "AI03" ;


COEFFICIENT (ALL,I,IND)                                        IINVS(I)
! INVESTMENT USAGE (BY IMPORTING SECTOR) ! ;
READ (ALL,I,IND) IINVS(I)
          FROM FILE  INHAR  HEADER                                  "AI04" ;


COEFFICIENT (ALL,I,IND)                                        DCON(I)
! CONSUMPTION USAGE (DOMESTIC)   ! ;
READ (ALL,I,IND) DCON(I)
          FROM FILE  INHAR  HEADER                                  "AI05" ;

COEFFICIENT (ALL,I,IND)                        ICONS(I)
! CONSUMPTION USAGE (BY IMPORTING SECTOR)   ! ;
```

```
READ (ALL,I,IND) ICONS(I)
        FROM FILE  INHAR  HEADER                          "AI06" ;

COEFFICIENT (ALL,I,IND)                                   DGOV(I)
! GOVERNMENT USAGE OF DOMESTIC COMMODITIES   ! ;
READ (ALL,I,IND) DGOV(I)
        FROM FILE  INHAR  HEADER                          "AI07" ;

COEFFICIENT (ALL,I,IND)                                   IGOVS(I)
! GOVERNMENT USAGE OF IMPORTED COMMODITIES   ! ;
READ (ALL,I,IND) IGOVS(I)
        FROM FILE  INHAR  HEADER                          "AI08" ;

COEFFICIENT (ALL,I,IND)                                   DSTOCK1(I) ;
READ (ALL,I,IND) DSTOCK1(I)
        FROM FILE INHAR HEADER                            "AI09" ;

COEFFICIENT (ALL,I,IND)                                   STOCK1(I) ;
READ (ALL,I,IND) ISTOCK1(I)
        FROM FILE INHAR HEADER                            AI10" ;


COEFFICIENT (ALL,I,IND)                                   EXPS(I)
! EXPORTS OF COMMODITY I  ! ;
READ (ALL,I,IND) EXPS(I)
        FROM FILE  INHAR  HEADER                          AI11" ;

COEFFICIENT (ALL,I,IND)                                   TSR(I)
! NON COMMODITY INDIRECT TAXES, NET OF SUBSIDIES ! ;
READ (ALL,I,IND) TSR(I)
        FROM FILE  INHAR  HEADER                          AI12" ;

COEFFICIENT (ALL,J,IND)                                   LAB(J)
! USAGE OF LABOUR  ! ;
READ (ALL,J,IND) LAB(J)
```

```
        FROM FILE INHAR HEADER                                  "AI13" ;

COEFFICIENT (ALL,J,IND)                                         CAP(J)
! USAGE OF CAPITAL ! ;
READ (ALL,J,IND) CAP(J)
        FROM FILE INHAR HEADER                                  "AI14" ;

COEFFICIENT (ALL,J,IND)                                         LAND(J)
! USAGE OF LAND   ! ;
READ (ALL,J,IND)LAND(J)
        FROM FILE INHAR HEADER                                  "AI15" ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                              TRD(I,J)
! COMMODITY TAXES ON DOMESTICALLY PRODUCED GOODS USED AS
INTERMEDIATE USAGE ! ;
READ (ALL,I,IND)(ALL,J,IND)TRD(I,J)
        FROM FILE INHAR HEADER                                  "AI16" ;


COEFFICIENT (ALL,I,IND)(ALL,J,IND)                              TRI(I,J)
! COMMODITY TAXES ON IMPORTS USED AS INTERMEDIATE USAGE ! ;
READ (ALL,I,IND)(ALL,J,IND)TRI(I,J)
        FROM FILE INHAR HEADER                                  "AI17" ;

COEFFICIENT (ALL,I,IND)                                         TCRD(I)
! COMMODITY TAXES ON DOMESTICALLY PRODUCED GOODS SOLD AS
CONSUMPTION ! ;
READ (ALL,I,IND) TCRD(I)
        FROM FILE INHAR HEADER                                  "AI18" ;

COEFFICIENT (ALL,I,IND)                                         TCRI(I)
! COMMODITY TAXES ON IMPORTS SOLD AS CONSUMPTION ! ;
READ (ALL,I,IND) TCRI(I)
        FROM FILE INHAR HEADER                                  "AI19" ;
```

```
COEFFICIENT (ALL,I,IND)                                    TIRD(I)
! COMMODITY TAXES ON DOMESTICALLY PRODUCED GOODS USED AS
INVESTMENT ! ;
READ (ALL,I,IND) TIRD(I)
        FROM FILE INHAR HEADER                              "AI20" ;

COEFFICIENT (ALL,I,IND)                                    TIRI(I)
! COMMODITY TAXES ON IMPORTED PRODUCED GOODS USED AS
INVESTMENT ! ;
READ (ALL,I,IND) TIRI(I)
        FROM FILE INHAR HEADER                              "AI21" ;


COEFFICIENT (ALL,I,IND)                                    TGRD(I)
! COMMODITY TAXES ON DOMESTICALLY PRODUCED GOODS SOLD TO
GOVERNMENT ! ;
READ (ALL,I,IND) TGRD(I)
        FROM FILE INHAR HEADER                              "AI22" ;


COEFFICIENT (ALL,I,IND)                                    TGRI(I)
! COMMODITY TAXES ON IMPORTS SOLD TO GOVERNMENT ! ;
READ (ALL,I,IND)TGRI(I)
        FROM FILE INHAR HEADER                              "AI23" ;


COEFFICIENT (ALL,I,IND                                     TER(I)
! COMMODITY TAXES EXPORTS ! ;
READ (ALL,I,IND TER(I)
        FROM FILE INHAR HEADER                              "AI24" ;

COEFFICIENT (ALL,I,IND)                                    TSTD1(I) ;
READ (ALL,I,IND)TSTD1(I)
        FROM FILE INHAR HEADER                              "AI25" ;
```

```
COEFFICIENT (ALL,I,IND)                                    TSTI1(I) ;
READ (ALL,I,IND) TSTI1(I)
        FROM FILE INHAR HEADER                                  "AI26" ;

COEFFICIENT (ALL,I,IND)                                    DR(I) ;
READ (ALL,I,IND) DR(I)
        FROM FILE INHAR HEADER                                  "AI27" ;


!*************************************************************************************
                        CALCULATING ORIGINAL GDP
*************************************************************************************!


!-------------------------------------------------------------------------------------
            AGGREGATION OF IMPORTS BY USER AND BY SECTOR
 ------------------------------------------------------------------------------------!


 COEFFICIENT (ALL,I,IND)                                   IINV(I)
! INVESTMENT USAGE (IMPORTED) ! ;
 FORMULA  (ALL,I,IND)
 IINV(I) = IINVS(I) ;

 COEFFICIENT (ALL,I,IND)                                   ICON(I)
! CONSUMPTION (IMPORTED) ! ;
 FORMULA  (ALL,I,IND)
 ICON(I) = ICONS(I) ;

 COEFFICIENT (ALL,I,IND)                                   IGOV(I)
! GOVERNMENT USAGE (IMPORTED) ! ;
 FORMULA  (ALL,I,IND)
 IGOV(I) = IGOVS(I) ;
```

```
!-------------------------------------------------------------------------------
                        GOVERNMENT AGGREGATES
 -------------------------------------------------------------------------------!

 COEFFICIENT                                                    AG
! GOVERNMENT SPENDING ON GOODS!  ;
 FORMULA
 AG = SUM(I,IND,DGOV(I)) + SUM(I,IND,IGOV(I))
      + SUM(I,IND,TGRD(I)) + SUM(I,IND,TGRI(I));

!-------------------------------------------------------------------------------
                        TRADE AGGREGATES
 -------------------------------------------------------------------------------!

COEFFICIENT (ALL,I,IND)                                         VALIMP(I);
 FORMULA (ALL,I,IND)
 VALIMP(I) =  SUM(J,IND,IINTS(I,J)) + ICONS(I)
           + IINVS(I) + IGOVS(I) ;


COEFFICIENT (ALL,I,IND)                                         VALEXP(I)
! TOTAL VALUE OF EXPORTS INCL. TAXES ! ;
 FORMULA (ALL,I,IND)
 VALEXP(I) = EXPS(I) + TER(I) ;

COEFFICIENT (ALL,I,IND)                                         IMPL(I)
! IMPORTS(CIF) OF THE ITH COMMODITY BY SECTOR !
 FORMULA  (ALL,I,IND)
 IMPL(I) = VALIMP(I)  - DR(I) ;

 COEFFICIENT                                                    IMPTT
! TOTAL IMPORTS(CIF) !  ;
 FORMULA
 IMPTT = SUM(I,IND,IMPL(I))  ;
```

```
 COEFFICIENT                                                    EXPFOB
! TOTAL EXPORTS AT FOB PRICES !   ;
 FORMULA
 EXPFOB = SUM(I,IND,VALEXP(I)) ;

!-------------------------------------------------------------------------------
                        EXPENDITURE AGGREGATES
 -------------------------------------------------------------------------------!

 COEFFICIENT                                                    CTT
! TOTAL CONSUMPTION ! ;
 FORMULA
 CTT = SUM(I,IND,DCON(I)) + SUM(I,IND,ICON(I))
       + SUM(I,IND,TCRD(I)) + SUM(I,IND,TCRI(I)) ;

 COEFFICIENT                                                    INVTT
! TOTAL INVESTMENT !   ;
 FORMULA
 INVTT = SUM(I,IND,DINV(I)) + SUM(I,IND,IINV(I))
       + SUM(I,IND,TIRD(I)) + SUM(I,IND,TIRI(I)) ;

!-------------------------------------------------------------------------------
             INCOME AGGREGATES
 -------------------------------------------------------------------------------!

 COEFFICIENT                                                    GDPET
! GDP FROM EXPENDITURE SIDE !   ;
 FORMULA
 GDPET = CTT + INVTT + AG+ EXPFOB - IMPTT ;

 COEFFICIENT                                                    GDPSC
! SCALE GDP TO CURRENT VALUE ! ;
 FORMULA
 GDPSC = GDP1 / GDPET ;
```

```
!*************************************************************************************!
!                              SCALING THE DATA BASE                                  !
!*************************************************************************************!

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                              SDINT(I,J)
! SCALING DINT ! ;
FORMULA (ALL,I,IND)(ALL,J,IND)
SDINT(I,J) = DINT(I,J) * GDPSC ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                              SIINTS(I,J)
! SCALING IINTS ! ;
FORMULA (ALL,I,IND)(ALL,J,IND)
SIINTS(I,J) = IINTS(I,J) * GDPSC ;

COEFFICIENT (ALL,I,IND)                                         SDINV(I)
! SCALING DINV ! ;
FORMULA (ALL,I,IND)
SDINV(I) = DINV(I) * GDPSC ;

COEFFICIENT (ALL,I,IND)                                         SIINVS(I)
! SCALING IINVS! ;
FORMULA (ALL,I,IND)
SIINVS(I) = IINVS(I) * GDPSC ;

COEFFICIENT (ALL,I,IND)                                          SDCON(I)
! SCALING DCON ! ;
FORMULA (ALL,I,IND)
SDCON(I) = DCON(I) * GDPSC ;

COEFFICIENT (ALL,I,IND)                                         SICONS(I)
! SCALING ICONS ! ;
FORMULA (ALL,I,IND)
SICONS(I) = ICONS(I) * GDPSC ;
```

```
COEFFICIENT (ALL,I,IND)                                    SDGOV(I)
! SCALING DGOV ! ;
FORMULA (ALL,I,IND)
SDGOV(I) = DGOV(I) * GDPSC ;

COEFFICIENT (ALL,I,IND)                                    SIGOVS(I)
! SCALING IGOVS ! ;
FORMULA (ALL,I,IND)
SIGOVS(I) = IGOVS(I) * GDPSC ;

COEFFICIENT (ALL,I,IND)                                    SEXPS(I)
! SCALING EXPS ! ;
FORMULA (ALL,I,IND)
SEXPS(I) = EXPS(I) * GDPSC ;

COEFFICIENT (ALL,I,IND)                                    STSR(I)
! SCALING TSR ! ;
FORMULA (ALL,I,IND)
STSR(I) = TSR(I) * GDPSC ;

COEFFICIENT (ALL,I,IND)                                    SLAB(I)
! SCALING LAB ! ;
FORMULA (ALL,I,IND)
SLAB(I) = LAB(I) * GDPSC ;

COEFFICIENT (ALL,I,IND)                                    SCAP(I)
! SCALING CAP ! ;
FORMULA (ALL,I,IND)
SCAP(I) = CAP(I) * GDPSC ;

COEFFICIENT (ALL,I,IND)                                    SLAND(I)
! SCALING LAND ! ;
FORMULA (ALL,I,IND)
SLAND(I) = LAND(I) * GDPSC ;
```

```
COEFFICIENT (ALL,I,IND)(ALL,J,IND)                    STRD(I,J)
! SCALING TRD ! ;
FORMULA (ALL,I,IND)(ALL,J,IND)
STRD(I,J) = TRD(I,J) * GDPSC ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                    STRI(I,J)
! SCALING TRI ! ;
FORMULA (ALL,I,IND)(ALL,J,IND)
STRI(I,J) = TRI(I,J) * GDPSC ;

COEFFICIENT (ALL,I,IND)                               STCRD(I)
! SCALING TCRD ! ;
FORMULA (ALL,I,IND)
STCRD(I) = TCRD(I) * GDPSC ;

COEFFICIENT (ALL,I,IND)                               STCRI(I)
! SCALING TCRI ! ;
FORMULA (ALL,I,IND)
STCRI(I) = TCRI(I) * GDPSC ;

COEFFICIENT (ALL,I,IND)                               STIRD(I)
! SCALING TIRD ! ;
FORMULA (ALL,I,IND)
STIRD(I) = TIRD(I) * GDPSC ;

COEFFICIENT (ALL,I,IND)                               STIRI(I)
! SCALING TIRI ! ;
FORMULA (ALL,I,IND)
STIRI(I) = TIRI(I) * GDPSC ;

COEFFICIENT (ALL,I,IND)                               STGRD(I)
! SCALING TGRD ! ;
FORMULA (ALL,I,IND)
STGRD(I) = TGRD(I) * GDPSC ;
```

```
COEFFICIENT (ALL,I,IND)                                    STGRI(I)
! SCALING TGRI ! ;
FORMULA (ALL,I,IND)
STGRI(I) = TGRI(I) * GDPSC ;

COEFFICIENT (ALL,I,IND)                                    STER(I)
! SCALING TER ! ;
FORMULA (ALL,I,IND)
STER(I) = TER(I) * GDPSC ;

COEFFICIENT (ALL,I,IND)                                    SDR(I)
! SCALING DR ! ;
FORMULA (ALL,I,IND)
SDR(I) = DR(I) * GDPSC ;

COEFFICIENT (ALL,I,IND)                                    SDSTOCK1(I) ;
FORMULA (ALL,I,IND)
SDSTOCK1(I) = DSTOCK1(I) * GDPSC ;


COEFFICIENT (ALL,I,IND)                                    SISTOCK1(I) ;
FORMULA (ALL,I,IND)
SISTOCK1(I) = ISTOCK1(I) * GDPSC ;

COEFFICIENT (ALL,I,IND)                                    STSTD1(I) ;
FORMULA (ALL,I,IND)
STSTD1(I) = TSTD1(I) * GDPSC ;

COEFFICIENT (ALL,I,IND)                                    STSTI1(I) ;
FORMULA (ALL,I,IND)
STSTI1(I) = TSTI1(I) * GDPSC ;
!-------------------------------------------------------------------------------
                          WRITE STATEMENTS
-------------------------------------------------------------------------------!
```

```
WRITE SDINT TO FILE OUTHAR HEADER "AI01" ;
    SIINTS TO FILE OUTHAR HEADER "AI02" ;
    SDINV TO FILE OUTHAR HEADER "AI03" ;
    SIINVS TO FILE OUTHAR HEADER "AI04" ;
    SDCON TO FILE OUTHAR HEADER "AI05" ;
    SICONS TO FILE OUTHAR HEADER "AI06" ;
    SDGOV TO FILE OUTHAR HEADER "AI07" ;
    SIGOVS TO FILE OUTHAR HEADER "AI08" ;
    SEXPS TO FILE OUTHAR HEADER "AI11" ;
    STSR TO FILE OUTHAR HEADER "AI12" ;
    SLAB TO FILE OUTHAR HEADER "AI13" ;
    SCAP TO FILE OUTHAR HEADER "AI14" ;
    SLAND TO FILE OUTHAR HEADER "AI15" ;
    STRD TO FILE OUTHAR HEADER "AI16" ;
    STRI TO FILE OUTHAR HEADER "AI17" ;
    STCRD TO FILE OUTHAR HEADER "AI18" ;
    STCRI TO FILE OUTHAR HEADER "AI19" ;
    STIRD TO FILE OUTHAR HEADER "AI20" ;
```

STIRI TO FILE OUTHAR HEADER "AI21" ;

STGRD TO FILE OUTHAR HEADER "AI22" ;

STGRI TO FILE OUTHAR HEADER "AI23" ;

STER TO FILE OUTHAR HEADER "AI24" ;

SDR TO FILE OUTHAR HEADER "AI27" ;

SDSTOCK1 TO FILE OUTHAR HEADER "AI09" ;

SISTOCK1 TO FILE OUTHAR HEADER "AI10" ;

STSTD1 TO FILE OUTHAR HEADER "AI25" ;

STSTI1 TO FILE OUTHAR HEADER "AI26" ;

!----------

END OF TABLO INPUT

----------!

Appendix B: Source Code For Program SUMDB

```
      PROGRAM SUMDB
C
C  ACRONYM
C  -------
C
C     ADD several Header ARray files (cell by cell within an array)
C     SUMDB
C
C
C  DESCRIPTION
C  -----------
C
C     This program allows a user to add the contents of several Header
C     Array files containing 2-dimensional real, integer or character
C     arrays, or 7-dimensional real arrays.  The program adds cell by
C     cell values within each real array and copies the other arrays,
C     from the first file.
C
C
C  CURRENT VERSION
C  ---------------
C
C     Version 1         January 1992
C
C  RELATED DOCUMENTATION
C  ---------------------
C
C     (None)
C
```

```
C  HISTORY
C  -------
C
C  Date        Author(s)        Comments
C  ------------------------------------------------------------
C
C  Jan 1992    Alex Strzelecki    Version 1
C
C
C
C     All variables used are declared explicitly.
C     implicit none
C
      CHARACTER*6 PRNAME
      PARAMETER( PRNAME= 'SUMDB')
C
      CHARACTER*70 PROGNV
      PARAMETER( PROGNV =
     *     '<//PRNAME//' Version 1.0    January 1992>')
C
C     PARAMETERS THAT MAY NEED TO BE VARIED FOR DIFFERENT
C     MACHINES
C
C
C     PARAMETERS THAT MAY NEED TO BE VARIED FOR DIFFERENT FILES
C
      INTEGER MMHEAD
      PARAMETER (MMHEAD=500)
      INTEGER MMNIDB
      PARAMETER (MMNIDB=20)
C
C  MMHEAD must be at least as great as the total number of arrays
C     (in the old or new files) processed.
C
C  MMNIDB must be at least as great as the total number of input
```

```
C     databases.
C
      INTEGER MMINT, MMREAL
      PARAMETER (MMINT=100000, MMREAL=100000 )
C
C  MMINT, MMREAL, MMCHAR  must be at least as great as the numbers
C  of integers, reals or character variables in any one of the arrays
C  read from or written to any file.
C  MMLCH must be at least as great as the longest of the character
C     variables processed.
C
C     WORKING ARRAYS
C
      INTEGER INTAR(MMINT)
      REAL REALAR(MMREAL), SUMAR(MMREAL)
      EQUIVALENCE (INTAR, REALAR)
C     Common blocks useful on some computers.
      COMMON /PRA1/ REALAR
      COMMON /PRA2/ SUMAR
C
C     PARAMETERS THAT MAY NEED TO BE VARIED FOR DIFFERENT
C     MACHINES
C
      INTEGER LLFNAM
      PARAMETER( LLFNAM=50)
C
      INTEGER  NEWUN, EXTUN, OLDUNS(MMNIDB)
      PARAMETER( NEWUN=12, EXTUN=5)
C
      INTEGER MMRECL
      PARAMETER(MMRECL=8000)
C
C     END OF PARAMETERS THAT MAY NEED TO BE VARIED FOR DIFFERENT
C     FILES OR MACHINES
C
```

```
C
C
      INTEGER LLHEAD, LLTYPE, LLNAME
      PARAMETER( LLHEAD=4, LLTYPE=2, LLNAME=70)
C
      INTEGER LLERRM
      PARAMETER( LLERRM= 70)
      CHARACTER*(LLERRM) ERRMES
C
C     ERRMES is for holding error messages.
C     In particular, if errmes(1:1)='E', an orderly stop is required.
C     ERRMES must be passed to nearly all subroutines and then passed
C     back to the main program if such an error occurs.
C
C
C
      INTEGER MAXDIM, J, NUMDIM
      PARAMETER (MAXDIM=7)
      INTEGER NELS(MMHEAD,MAXDIM), NELSR(MAXDIM)
C
      INTEGER NEWFL, EXTFL, OLDFL
      INTEGER NOLDHD
      INTEGER I, K, KK
      INTEGER DIM, NUMELS, UNIT

C     integer function
      INTEGER NLFNAM
C
      INTEGER LENGNW
      CHARACTER*(LLFNAM) NEWFN, OLDFNS(MMNIDB), OLDFN, EXTFN
      CHARACTER*(LLHEAD) OLDHD(MMHEAD+2), HEADER
      CHARACTER*(LLTYPE) DATTP(MMHEAD+2), DTYPE
      CHARACTER*(LLNAME) OLDLN(MMHEAD+2), LNAME
      CHARACTER*3 FILENO
C
```

```
C
      LOGICAL NEWOP, ISHIST, READOK, EXTOP
      LOGICAL OLDOPS(MMNIDB), CHTYPE, CHSIZE, FOUND
      LOGICAL OPENOK
C
C
C  SUBROUTINES AND FUNCTIONS CALLED
C  --------------------------------
C
      EXTERNAL ECHPRV, OPFILE, RWHIST, PUT7R, PUTIM
      EXTERNAL LSTHDS, LFNAM, OPFLPN, WRCREF
      EXTERNAL CLNWHF, CLSEQ
      EXTERNAL PERET, PERETM
C
C      INITIALIZE ARRAYS
C
      DO  5  J = 1, MAXDIM
        NELSR(J) = 1
   5  CONTINUE
C
C
C      BEGIN
C
C
C      Initialize ERRMES.
C
      ERRMES = ' '
C
C      Initialise logical units for input data bases
C
      DO  10  K = 1,MMNIDB
        OLDUNS(K) = K + 20
  10  CONTINUE
C
C
```

```
C     Echo program name and version to the screen.
C
      CALL ECHPRV( PROGNV, ERRMES)
      IF( ERRMES(1:1).EQ.'E') GOTO 15000
C
C
C
C     INITIALIZE LOGICAL VARIABLES TO SAY THAT, AS YET, NONE OF THE
C     FILES IS OPEN.
C
      EXTOP=.FALSE.
      NEWOP=.FALSE.
      DO  20  K = 1,MMNIDB
        OLDOPS(K) = .FALSE.
 20   CONTINUE
C
C     OPEN THE NEW HEADER ARRAY FILE
C
      WRITE(*,10035)' NEW HEADER ARRAY FILE TO BE CREATED'
      WRITE(*,10015)' ---------------------------------'
C
      IF  (.NOT.EXTOP)  THEN
      CALL OPFILE( NEWUN, 'new', 'header', NEWFN,
     *            OPENOK, ERRMES)
      IF( ERRMES(1:1).EQ.'E') GOTO 15000
C
      ELSE
      READ  (EXTUN,*)  NEWFN
      NEWFL=NLFNAM( NEWFN, LLFNAM, ERRMES)
      IF( ERRMES(1:1).EQ.'E') GOTO 15000
      CALL OPFLPN(NEWUN,
     *  NEWFN(1:NEWFL),
     *  'NEW','HEADER',OPENOK,
     *  ERRMES)
      IF( ERRMES(1:1).EQ.'E') GOTO 15000
```

```
      END IF
C
      IF(.NOT.OPENOK) THEN
          ERRMES='E-UNABLE TO OPEN NEW HEADER ARRAY FILE'
          GOTO 15500
      END IF
C
      NEWOP=.TRUE.
C
C     Write Creation Information on new file.
C
      CALL WRCREF( NEWUN, PROGNV, 'header', ERRMES)
      IF( ERRMES(1:1).EQ.'E') GOTO 15000
      WRITE(*,10035)
     *  ' Enter the number of input databases you want to merge ?'
      IF (.NOT.EXTOP)  THEN
        READ (*,*)  NIDB
      ELSE
        READ (EXTUN,*)  NIDB
      END IF
C
22    FORMAT(/,2X, I3 / )
C
      WRITE(*,10035)' NUMBER OF INPUT DATA BASES'
C
      WRITE(*,10015)' -------------------------'
C
      WRITE(*,22)  NIDB
C
C     Read input file names and open input files.
C
10060  FORMAT (I3)
      DO  30  K = 1,NIDB
C
      WRITE (FILENO,10060)  K
```

```
      WRITE(*,10035)' INPUT HEADER ARRAY FILE TO BE ADDED # '//FILENO
      WRITE(*,10015)' ------------------------------------'
C
      IF (.NOT.EXTOP)  THEN
      CALL OPFILE( OLDUNS(K), 'old', 'header', OLDFNS(K),
     *            OPENOK, ERRMES)
       IF( ERRMES(1:1).EQ.'E') GOTO 15000
       ELSE
         READ  (EXTUN,*)  OLDFNS(K)
         OLDFL=NLFNAM( OLDFNS(K), LLFNAM, ERRMES)
         OLDFN = OLDFNS(K)
         CALL OPFLPN(OLDUNS(K),
     *         OLDFN(1:OLDFL),
     *         'OLD','HEADER',OPENOK,
     *         ERRMES)
        IF( ERRMES(1:1).EQ.'E') GOTO 15000
       END IF
C
      IF(.NOT.OPENOK) THEN
        ERRMES='E-UNABLE TO OPEN OLD HEADER ARRAY FILE # '//FILENO
        GOTO 15500
      END IF
C
      OLDOPS(K) = .TRUE.
      IF( ERRMES(1:1).EQ.'E') GOTO 15000
C
C
C           SUMMARIZE THE OLD HEADER ARRAY FILE
C
      IF (K.EQ.1)  THEN
        CALL LSTHDS( OLDUNS(K), MMHEAD, 'MMHEAD',
     *            OLDHD, NOLDHD, DATTP, NELS, MAXDIM, OLDLN,
     *            ERRMES)
        IF( ERRMES(1:1).EQ.'E') GOTO 15000
      END IF
```

```
C
  30  CONTINUE
C
C     BEGIN ADDING ALL THE REAL ARRAYS ON THE INPUT FILES
C
      DO  500  KK = 1,NOLDHD
        HEADER = OLDHD(KK)
        DO  400  K = 1,NIDB
          UNIT = OLDUNS(K)
          DTYPE = ' '
          CALL  RDAR  ( UNIT, HEADER,
     *           REALAR, MMREAL, INTAR, MMINT,
     *           NELSR , MAXDIM, LNAME, DTYPE,
     *           NUMDIM, ERRMES)
          IF (ERRMES(1:1) .EQ. 'E') GO TO 15000
          IF (NUMDIM.NE.0)  THEN
C
          IF (DTYPE.EQ.'RL' .OR. DTYPE.EQ.'2R')  THEN
C               Initialise SUM ARray
            DO  70  J = 1, MAXDIM
              IF (NELSR(J).EQ.0)  NELSR(J) = 1
  70          CONTINUE
            IF (K.EQ.1)  THEN
              NUMELS = 1
              DO  80  J = 1, MAXDIM
                DIM = 1
                IF (NELSR(J).GT.1)  DIM = NELSR(J)
                NUMELS = NUMELS * DIM
  80            CONTINUE
              DO  90  I = 1,NUMELS
                SUMAR(I) = 0.0
  90            CONTINUE
            END IF
C               Add elements of the array
            DO  100  I = 1,NUMELS
```

```
            SUMAR(I) = SUMAR(I) + REALAR(I)
 100        CONTINUE
          ELSE
            IF (DTYPE.EQ.'2I'.AND. K.EQ.1)  THEN
C             Write Integer Array
              CALL  PUTIM (NEWUN, HEADER, INTAR, NELSR(1),
     *                    NELSR(2),
     *                    NELSR(1), NELSR(2), LNAME, ERRMES )
              IF( ERRMES(1:1).EQ.'E') GOTO 15000
C
            END IF
          END IF
          END IF
 400      CONTINUE
C
        IF (NUMDIM.NE.0)  THEN
        IF (DTYPE.EQ.'RL' .OR. DTYPE.EQ.'2R')  THEN
C
C         Write header to the output file
C
10070       FORMAT (1X, 'NELSR = ', 7I5 )
10080       FORMAT (1X, 'SUMAR = ', F20.5 )
          CALL  PUT7R (NEWUN, HEADER , SUMAR,
     *          NELSR(1), NELSR(2), NELSR(3), NELSR(4),
     *          NELSR(5), NELSR(6), NELSR(7),
     *          1, 1, 1, 1, 1, 1, 1,
     *          NELSR(1), NELSR(2), NELSR(3), NELSR(4),
     *          NELSR(5), NELSR(6), NELSR(7),
     *          LNAME, MMRECL,
     *          ERRMES)
          IF (ERRMES(1:1) .EQ. 'E') GO TO 15000
        END IF
        END IF
 500    CONTINUE
C
```

```
C
          CALL CLSEQ ( NEWUN, 'keep', ERRMES)
           IF( ERRMES(1:1).EQ.'E') GOTO 15000
C
          IF (EXTOP)  THEN
             CALL  CLSEQ( EXTUN, 'keep', ERRMES)
          END IF
C
           NEWOP=.FALSE.
           WRITE(*,600) NEWFN(1:NEWFL)
600          FORMAT(/,' Saved new HEADER ARRAY file: ',A,/)
C
          DO  750  K = 1,NIDB
             IF(OLDOPS(K)) THEN
               CALL CLSEQ( OLDUNS(K), 'keep', ERRMES)
             END IF
 750         CONTINUE
C
          STOP 'SUMDB TERMINATED NORMALLY'
C
C
C
C     FORMATS FOR READING AND WRITING CHARACTERS
10005  FORMAT(A)
10015  FORMAT(1X,A)
10035  FORMAT(/,1X,A)
C
C
C     ERROR RETURN FOR ORDERLY STOP
C
15000  CONTINUE
      CALL PERET( PRNAME)
      GOTO 16000
C
15500  CONTINUE
```

```
      CALL PERETM( PRNAME, ERRMES)
      GOTO 16000
C
C
C     CLOSE ALL OPEN FILES, DISPOSING EACH ONE AS APPROPRIATE.
C     Note that if a further error occurs, we ignore it and close
C     as many files as possible.
C
16000  CONTINUE
C
16010  FORMAT( ' Incomplete ', A, ' file deleted.')
16011  FORMAT( ' Incomplete ', A, ' file was KEPT.')
C
      IF(NEWOP) THEN
          CALL CLSEQ ( NEWUN, 'delete', ERRMES)
          WRITE(*,16010) 'new HEADER ARRAY'
      END IF
C
      IF (EXTOP)  THEN
        CALL  CLSEQ( EXTUN, 'keep', ERRMES)
      END IF
C
      DO  800  K = 1,NIDB
      IF(OLDOPS(K)) THEN
          CALL CLSEQ( OLDUNS(K), 'keep', ERRMES)
      END IF
 800  CONTINUE
C
C     ERROR RETURN STOP
C
     STOP 'ERRORS ENCOUNTERED IN SUMDB'
     END
     SUBROUTINE  RDAR  ( LUHAR, HEAD,
    *      RARRAY, MAXRSZ, INTAR, MMINT,
    *      NEREAD, MAXDIM, NAME,  TYPE,
```

```
     *       NUMDIM, ERRMES)
C
      CHARACTER  *6 SUBNAM
      PARAMETER  ( SUBNAM = 'RDAR' )
C
      INTEGER  MAXRSZ, MAXDIM, NUMDIM, MMINT
      INTEGER  INTAR(MMINT)
      REAL  RARRAY(MAXRSZ)
      CHARACTER  *(*) ERRMES
      CHARACTER  *(*) NAME
      CHARACTER  *(*) HEAD
      CHARACTER  *(2) TYPE
      LOGICAL  FOUND
C
C     NEREAD - Number of each dimension actually read
C
      INTEGER       NEREAD(MAXDIM)
C
      EXTERNAL  GETRVN, GTRVAD, GETIVN, FINDAR
C
      ERRMES = ' '
      CALL FINDAR (LUHAR,HEAD,TYPE,NEREAD,NUMDIM,MAXDIM,
     &            NAME,FOUND,ERRMES)
      IF (ERRMES(1:1) .EQ. 'E') GOTO 15000
C
      IF (FOUND)  THEN
C
C       Read in the array values.
C
        IF(TYPE.EQ.'2R') THEN
          CALL GETRVN (LUHAR, HEAD,
     *       RARRAY, MAXRSZ, 1,
     *       NEREAD(1), NEREAD(2), NAME,
     *       ERRMES)
          IF (ERRMES(1:1) .EQ. 'E') GOTO 15000
```

```
C
        NUMDIM = 2
C
      ELSE IF(TYPE.EQ.'RL') THEN
C
        CALL GTRVAD( LUHAR, HEAD,
     *     RARRAY, MAXRSZ, 'MAXRSZ', 1,
     *     NEREAD, MAXDIM, NAME,
     *     .TRUE.,
     *     ERRMES)
        IF (ERRMES(1:1) .EQ. 'E') GOTO 15000
C
        NUMDIM = 0
        DO  300  K = 1,MAXDIM
          IF (NEREAD(K).GT.1)  NUMDIM = NUMDIM + 1
300       CONTINUE
        IF (NUMDIM.LT.2)  NUMDIM = 2
    ELSE
C
      IF(TYPE.EQ.'2I') THEN
        CALL GETIVN (LUHAR, HEAD,
     *     INTAR , MMINT , 1,
     *     NEREAD(1), NEREAD(2), NAME,
     *     ERRMES)
        IF (ERRMES(1:1) .EQ. 'E') GOTO 15000
C
        NUMDIM = 2
      ENDIF
    ENDIF
    ELSE
      NUMDIM = 0
    ENDIF
C
C
     RETURN
```

```
15000  CONTINUE
       CALL  PERETM (SUBNAM, ERRMES)
       RETURN
       END
C      GEMPACK Software System                Impact Project
C      ----------------------                 --------------
C
       SUBROUTINE FINDAR( UNIT, HEADER, DTYPE, NELS, NUMDIM,
     *   MAXDIM, LNAME, FOUND,
     *   ERRMES)
C
C  ACRONYM
C  -------
C
C      FIND the ARray specified.
C      FIND    AR
C
C  DESCRIPTION
C  -----------
C
C      This searches the Header Array file for the
C      specified header.
C      If it is found, it returns the data type of the array, its
C      size and long name.
C      The logical variable FOUND must be tested after a call to
C      this to check whether the header was found on the file.
C
C  CURRENT VERSION
C  ---------------
C
C      Version 1      July 1986
C      Version 2      Sept 1992
C
C  RELATED DOCUMENTATION
C  ---------------------
```

```
C
C     (1) SPECIFICATION:
C
C     None is available at present.
C
C  HISTORY
C  -------
C
C  Date         Author(s)         Comments
C  ------------------------------------------------------------
C  10 July 1986  Ken Pearson       Version 1.
C                                  Initial coding
C
C  02 Sept 1992  A Strzelecki      Version 2.
C                                  Read 7-D arrays
C
C  NOTES
C  -----
C
C     - If the header HEADER is found on the file, on return
C       the file is positioned just before the record containing
C       HEADER.
C     - If, on return, FOUND is .TRUE. then
C
C          -- if DTYPE is '2I' or '2R', NELS(1) and NELS(2) hold the
C             numbers of rows and columns of the matrix,
C          -- if DTYPE is '1C', NELS(1) is the number of strings and
C             NELS(2) is the length of each string.
C
C          -- if DTYPE is 'RL'  NELS(.)  hold the dimensions
C             of the matrix, and NUMDIM holds the number of dimensions.
C
C     - If, on return, FOUND is .FALSE., all of DTYPE, NELS and NUMDIM
C       are undefined.
C
```

```
C
C  SUBROUTINE VARIABLES
C  ---------------------
C
C  (1) INPUT VARIABLES (NOT ALTERED):
C
C     UNIT     - The logical unit number of the Header Array file.
C     HEADER   - The specified header.
C     MAXDIM   - Maximum number of dimensions in Gempack header arrays
C
      INTEGER        UNIT
      CHARACTER*(*)  HEADER
C
C  (2) OUTPUT VARIABLES (RETURNED):
C
C     DTYPE    - The data type.
C     NELS(.)  - The dimensions array.
C     NUMDIM   - The number of dimensions of the specified array.
C     LNAME    _ The long name associated with the array.
C     FOUND    - A logical variable, indicating whether or not the
C                header HEADER was found on the file.
C
      CHARACTER*(*)  DTYPE
      INTEGER        NUMDIM, MAXDIM
      INTEGER        NELS(MAXDIM)
      CHARACTER*(*)  LNAME
      LOGICAL        FOUND
C
C  (3) INPUT/OUTPUT VARIABLES:
C
C     ERRMES - Contains fatal error messages if ERRMES(1:1)='E'
C
      CHARACTER*(*)  ERRMES
C
C  SIGNIFICANT PARAMETERS
```

```
C  --------------------
C
C     SUBNAM  - the name of this subroutine.
C     LLHEAD  - the length of headers on a Header Array file.
C     FILPB   - the all-blanks character constant at the start of all
C               non-header records of a Header Array file.
C     LLSTM   - The length of character variables describing storage
C               method.
C     LLNAME  - The length of long names on Header Array files.
C     MMSUM   - the maximum number of summary integers allowed on this
C               record (as presently implemented).
C
      CHARACTER*6 SUBNAM
      PARAMETER  (SUBNAM='FINDAR')
C
      INTEGER LLHEAD
      PARAMETER( LLHEAD=4)
      CHARACTER*(LLHEAD) FILPB
      PARAMETER( FILPB='    ')
C
      INTEGER LLSTM
      PARAMETER( LLSTM=4)
C
      INTEGER LLNAME
      PARAMETER(  LLNAME=70)
C
      INTEGER MMSUM
      PARAMETER( MMSUM=7)
C
C  SIGNIFICANT LOCAL DECLARATIONS
C  ------------------------------
C
C     STMETH     - The storage method for the array.
C     SUM(MMSUM)  - Array to hold the summary integers associated with
C                   the header array in question.
```

```
C     NSUM      - The number of summary integers, as read from the file.
C
      CHARACTER*(LLSTM) STMETH
      INTEGER       SUM(MMSUM)
      INTEGER       NSUM
C
C  ERROR CHECKING AND HANDLING
C  ---------------------------
C
C  (1) VALIDATION OF SUBROUTINE INPUT VARIABLES:
C
C     A fatal error (ERRMES(1:1)='E') results if:
C
C     - HEADER contains only blanks.
C     - LEN(HEADER) is different from LLHEAD.
C     - MAXDIM is different from NSUM
C
C  (2) VALIDATION OF DATA RETURNED FROM OTHER SUBROUTINES:
C
C     A fatal error (ERRMES(1:1)='E') results if:
C
C     - DTYPE or LNAME are variables of the wrong character lengths
C       (subroutine RDR2HA).
C     - DTYPE, STMETH, NSUM and SUM(1..NSUM) are incompatible with current
C       implementation (subroutine RDR2HA).
C     - There is a read error (subroutine RDR2HA).
C
C  (3) VALIDATION OF SUBROUTINE OUTPUT VARIABLES:
C
C     A fatal error (ERRMES(1:1)='E') results if:
C
C     - NELS(1) is defined and is less than 0.
C     - NELS(2) is defined and is less than 0 (real or integer data) or
C       is less than 1 (character data).
C
```

```
C  SUBROUTINES AND FUNCTIONS CALLED
C  -------------------------------
C
      EXTERNAL     PS4BFS, RDR2HA
      EXTERNAL     PERET, PERETM
      INTRINSIC    LEN
C
C     ************************************************************
C
C     End of subroutine documentation
C
C     ************************************************************
C
C  LOCAL VARIABLES
C  ---------------
C
      INTEGER N1, NCMIN
      INTEGER IOSTAT
C
C  IMPORTANT STATEMENT LABELS
C  --------------------------
C
C     10015  10035  FORMATs for char reads and writes
C     15000  15500  Error return for orderly stop
C     23800         Trap for backspace error
C
C     ************************************************************
C
C     Subroutine begins
C
C     ************************************************************
C
C
C     Check if maximum dimensions values correspond
C
```

```
      IF (MAXDIM .NE. MMSUM) THEN
        ERRMES='E-Mismatch in MAXDIM and MMSUM parameter values'
        GOTO 15500
      ENDIF
C
C     Check that HEADER is of the expected length.
C
      N1=LEN(HEADER)
C
      IF(N1.NE.LLHEAD) THEN
           WRITE(*,10) LLHEAD
           WRITE(*,12) SUBNAM
           WRITE(*,14) N1
10    FORMAT( /,
     *  ' Headers on Header Array files must have exactly ',
     *  I2,' characters,')
12    FORMAT('  but, in one call to subroutine "',A,'" the')
14    FORMAT('   header has ',I5,' characters.')
C
           ERRMES='E-UNEXPECTED HEADER LENGTH'
           GOTO 15500
      END IF
C
C     Check that HEADER is not all blanks.
C
      IF(HEADER.EQ.FILPB) THEN
           WRITE(*,10035)' Headers on Header Array files'
           WRITE(*,10015)'  must not consist only of blanks.'
           ERRMES='E-HEADER CANNOT BE ALL BLANKS'
           GOTO 15500
      END IF
C
      DO 5 I = 1, MAXDIM
        SUM(I) = 1
 5    CONTINUE
```

```
C
C     Position the file after the given header (if it is there).
C
      CALL PS4BFS( UNIT, HEADER, FOUND, ERRMES)
      IF( ERRMES(1:1).EQ.'E') GOTO 15000
C
      IF(.NOT.FOUND) THEN
          RETURN
      END IF
C
C     Read the second record from the array.
C
      CALL RDR2HA( UNIT, DTYPE, STMETH,
     *   MMSUM, NSUM, SUM, LNAME, ERRMES)
      IF( ERRMES(1:1).EQ.'E') GOTO 15000
C
C     Deduce NELS from DTYPE, STMETH, NSUM and SUM(1..NSUM).
C     (Note that RDR2HA verifies that NSUM is compatible with
C     DTYPE and STMETH.)
C     *** If other storage methods are implemented, this will
C       have to be changed. ************************************
C
40    FORMAT(/,
     *  ' ** Error reading array with header '',A,''.')
75    FORMAT(/,
     *  ' ** Unexpected mix of data type and storage method.')
80    FORMAT(
     *  ' ** Data type is '',A,''  and storage method is '',
     *  A,''.')
C
C
      IF(DTYPE.EQ.'2I') THEN
C
        IF(STMETH.EQ.'FULL') THEN
C
```

```
          NELS(1)=SUM(1)
          NELS(2)=SUM(2)
C
        ELSE
          WRITE(*,40) HEADER
          WRITE(*,75)
          WRITE(*,80) DTYPE, STMETH
          ERRMES='E-UNEXPECTED STORAGE METHOD'
          GOTO 15500
        END IF
C
      ELSE IF(DTYPE.EQ.'2R') THEN
C
        IF(STMETH.EQ.'FULL') THEN
C
          NELS(1)=SUM(1)
          NELS(2)=SUM(2)
C
        ELSE
          WRITE(*,40) HEADER
          WRITE(*,75)
          WRITE(*,80) DTYPE, STMETH
          ERRMES='E-UNEXPECTED STORAGE METHOD'
          GOTO 15500
        END IF
C
      ELSE IF(DTYPE.EQ.'1C') THEN
C
        IF(STMETH.EQ.'FULL') THEN
C
          NELS(1)=SUM(1)
          NELS(2)=SUM(2)
C
        ELSE
          WRITE(*,40) HEADER
```

```
            WRITE(*,75)
            WRITE(*,80) DTYPE, STMETH
            ERRMES='E-UNEXPECTED STORAGE METHOD'
            GOTO 15500
          END IF
C
        ELSE IF(DTYPE.EQ.'RL') THEN
C
          IF(STMETH.EQ.'FULL') THEN
C
            DO 95 I = 1, NSUM
              NELS(I) = SUM(I)
95          CONTINUE
C
          ELSE
            WRITE(*,40) HEADER
            WRITE(*,75)
            WRITE(*,80) DTYPE, STMETH
            ERRMES='E-UNEXPECTED STORAGE METHOD'
            GOTO 15500
          END IF
C
        ELSE
            WRITE(*,40) HEADER
            WRITE(*,75)
            WRITE(*,80) DTYPE, STMETH
            ERRMES='E-UNEXPECTED DATA TYPE'
            GOTO 15500
C
          END IF
C
C       Test that NELS(1) is at least 0 and that NELS(2) is at least 0
C       (for real or integer data) or at least 1 (for character data).
C
        NCMIN=0
```

```
      IF(DTYPE.EQ.'1C') NCMIN=1
C
      IF(NELS(1).LT.0.OR.NELS(2).LT.NCMIN) THEN
          WRITE(*,40) HEADER
          WRITE(*,10035)' Unexpected size of data.'
          WRITE(*,*)'   NROW = ',NROW
          WRITE(*,*)'   NCOL = ',NCOL
          ERRMES='E-UNEXPECTED SIZE OF DATA'
          GOTO 15500
      END IF
C
C     Backspace to just before the header record.
C
      BACKSPACE(UNIT=UNIT, ERR=23800, IOSTAT=IOSTAT)
      BACKSPACE(UNIT=UNIT, ERR=23800, IOSTAT=IOSTAT)
C
C     Normal return
C     -------------
C
      RETURN
C
C     Formats for character READs and WRITEs
C     --------------------------------------
C
10015  FORMAT (1X, A)
10035  FORMAT (/, 1X, A)
C
C     Error RETURN for orderly stop
C     -----------------------------
C
15000  CONTINUE
      CALL PERET (SUBNAM)
      RETURN
C
15500  CONTINUE
```

```
      CALL PERETM (SUBNAM, ERRMES)
      RETURN
C
C     Trap for backspace error.
C     -------------------------
C
23800 CONTINUE
      WRITE(*,10035)' Error backspacing file'
      WRITE(*,*)' attached to unit number: ',UNIT
      WRITE(*,*)'    (IOSTAT value: ',IOSTAT,'.)'
      ERRMES='E-ERROR BACKSPACING FILE'
      GOTO 15500
C
      END
```

Appendix C: Source Code For Program EXTRACT

```
!*****************************************************************************!
!                                    FILES                                    !
!*****************************************************************************!

FILE iotab # input-output table #;
FILE GEN # auxiliary data file for single-region revision procedure #;
FILE (NEW) iocoeff  # input-output coefficients #;

!*****************************************************************************!
!                                    SETS                                     !
!*****************************************************************************!

COEFFICIENT (INTEGER)
 NO_IND # number of sectors #;
READ NO_IND FROM FILE gen HEADER                                    "G001";

SET
 IND # sectors # MAXIMUM SIZE 40 SIZE NO_IND;

!*****************************************************************************!
!                                  DATA BASE                                  !
!*****************************************************************************!


!-----------------------------------------------------------------------------
                    INPUT-OUTPUT DATA BASE
 -----------------------------------------------------------------------------!


COEFFICIENT (ALL,I,IND)(ALL,J,IND)                                  DINT(I,J)
! INTERMEDIATE USAGE (DOMESTIC) !     ;
READ (ALL,I,IND)(ALL,J,IND) DINT(I,J)
```

```
        FROM FILE  iotab  HEADER                                    "AI01";

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                                  IINTS(I,J)
! INTERMEDIATE USAGE (BY IMPORTING SECTOR) !     ;
READ (ALL,I,IND)(ALL,J,IND) IINTS(I,J)
        FROM FILE  iotab  HEADER                                    "AI02";

COEFFICIENT (ALL,I,IND)                                             DINV(I)
! INVESTMENT USAGE (DOMESTIC) !;
READ (ALL,I,IND) DINV(I)
        FROM FILE  iotab  HEADER                                    "AI03";


COEFFICIENT (ALL,I,IND)                                             IINVS(I)
! INVESTMENT USAGE (BY IMPORTING SECTOR) !;
READ (ALL,I,IND)IINVS(I)
        FROM FILE  iotab  HEADER                                    "AI04";

COEFFICIENT (ALL,I,IND)                                             DCON(I)
! CONSUMPTION USAGE (DOMESTIC)   !;
READ (ALL,I,IND) DCON(I)
        FROM FILE  iotab  HEADER                                    "AI05";

COEFFICIENT (ALL,I,IND)                                             ICONS(I)
! CONSUMPTION USAGE (BY IMPORTING SECTOR)   !;
READ (ALL,I,IND) ICONS(I)
        FROM FILE  iotab  HEADER                                    "AI06";

COEFFICIENT (ALL,I,IND)                                             DGOV(I)
! GOVERNMENT USAGE OF DOMESTIC COMMODITIES   !;
READ (ALL,I,IND) DGOV(I)
        FROM FILE  iotab  HEADER                                    "AI07";

COEFFICIENT (ALL,I,IND)                                             IGOVS(I)
! GOVERNMENT USAGE OF IMPORTED COMMODITIES  !;
```

```
READ (ALL,I,IND) IGOVS(I)
        FROM FILE iotab HEADER                          "AI08";

COEFFICIENT (ALL,I,IND)                                 EXPS(I)
! EXPORTS OF COMMODITY I !;
READ (ALL,I,IND) EXPS(I)
        FROM FILE iotab HEADER                          "AI11";

COEFFICIENT (ALL,J,IND)                                 LAB(J)
! USAGE OF LABOUR !;
READ (ALL,J,IND) LAB(J)
        FROM FILE iotab HEADER                          "AI13";

COEFFICIENT (ALL,J,IND)                                 CAP(J)
! USAGE OF CAPITAL !;
READ (ALL,J,IND) CAP(J)
        FROM FILE iotab HEADER                          "AI14";

COEFFICIENT (ALL,J,IND)                                 LAND(J)
! USAGE OF LAND   !;
READ (ALL,J,IND) LAND(J)
        FROM FILE iotab HEADER                          "AI15";

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                      TRD(I,J)
! COMMODITY TAXES ON DOMESTICALLY PRODUCED GOODS USED AS
INTERMEDIATE USAGE !;
READ (ALL,I,IND)(ALL,J,IND) TRD(I,J)
        FROM FILE iotab HEADER                          "AI16";

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                      TRI(I,J)
! COMMODITY TAXES ON IMPORTS USED AS INTERMEDIATE USAGE !;
READ (ALL,I,IND)(ALL,J,IND) TRI(I,J)
        FROM FILE iotab HEADER                          AI17";

COEFFICIENT (ALL,I,IND)                                 TCRD(I)
```

```
! COMMODITY TAXES ON DOMESTICALLY PRODUCED GOODS SOLD AS
CONSUMPTION !;
READ (ALL,I,IND) TCRD(I)
        FROM FILE iotab HEADER                          "AI18";

COEFFICIENT (ALL,I,IND)                                 TCRI(I)
! COMMODITY TAXES ON IMPORTS SOLD AS CONSUMPTION !;
READ (ALL,I,IND) TCRI(I)
        FROM FILE iotab HEADER                          "AI19";

COEFFICIENT (ALL,I,IND)                                 TIRD(I)
! COMMODITY TAXES ON DOMESTICALLY PRODUCED GOODS USED AS
INVESTMENT !;
READ (ALL,I,IND) TIRD(I)
        FROM FILE iotab HEADER                          "AI20";

COEFFICIENT (ALL,I,IND)                                 TIRI(I)
! COMMODITY TAXES ON IMPORTED PRODUCED GOODS USED AS
INVESTMENT !;
READ (ALL,I,IND) TIRI(I)
        FROM FILE iotab HEADER                          "AI21";

COEFFICIENT (ALL,I,IND)                                 TGRD(I)
! COMMODITY TAXES ON DOMESTICALLY PRODUCED GOODS SOLD TO
GOVERNMENT !;
READ (ALL,I,IND) TGRD(I)
        FROM FILE iotab HEADER                          "AI22";

COEFFICIENT (ALL,I,IND)                                 TGRI(I)
! COMMODITY TAXES ON IMPORTS SOLD TO GOVERNMENT !;
READ (ALL,I,IND) TGRI(I)
        FROM FILE iotab HEADER                          "AI23";

COEFFICIENT (ALL,I,IND)                                 TER(I)
! COMMODITY TAXES EXPORTS !;
```

```
READ (ALL,I,IND) TER(I)
        FROM FILE  iotab  HEADER                                    "AI24";

!*****************************************************************************
                WIDELY USED DATABASE AGGREGATES
*****************************************************************************!

   ZERODIVIDE DEFAULT 0.0;

!-----------------------------------------------------------------------------
        AGGREGATION OF IMPORTS BY USER AND BY SECTOR
 -----------------------------------------------------------------------------!

 COEFFICIENT (ALL,I,IND)(ALL,J,IND)                          IINT(I,J)
! INTERMEDIATE USAGE (IMPORTED) ! ;
 FORMULA  (ALL,I,IND)(ALL,J,IND)
 IINT(I,J) = IINTS(I,J) ;

 COEFFICIENT (ALL,I,IND)                                     IINV(I)
! INVESTMENT USAGE (IMPORTED) ! ;
 FORMULA  (ALL,I,IND)
 IINV(I) = IINVS(I) ;

 COEFFICIENT (ALL,I,IND)                                     ICON(I)
! CONSUMPTION (IMPORTED) ! ;
 FORMULA  (ALL,I,IND)
 ICON(I) = ICONS(I) ;

 COEFFICIENT (ALL,I,IND)                                     IGOV(I)
! GOVERNMENT USAGE (IMPORTED) ! ;
 FORMULA  (ALL,I,IND)
 IGOV(I) = IGOVS(I) ;
```

```
!-------------------------------------------------------------------------------
                        GOVERNMENT AGGREGATES
 ------------------------------------------------------------------------------!

 COEFFICIENT                                                        AG
! GOVERNMENT SPENDING ON GOODS! ;
 FORMULA
 AG = SUM(I,IND,DGOV(I)) + SUM(I,IND,IGOV(I))
    + SUM(I,IND,TGRD(I)) + SUM(I,IND,TGRI(I));

!-------------------------------------------------------------------------------
                        TRADE AGGREGATES
!-------------------------------------------------------------------------------

COEFFICIENT (ALL,I,IND)                                             VALEXP(I)
! TOTAL VALUE OF EXPORTS INCL. TAXES !;
 FORMULA (ALL,I,IND)
 VALEXP(I) = EXPS(I) + TER(I);

!-------------------------------------------------------------------------------
                        PRODUCTION AGGREGATES
 ------------------------------------------------------------------------------!

COEFFICIENT (ALL,I,IND)                                             COSTINP(I)
! TOTAL COSTS OF INPUTS TO INDUSTRY EXCLUDING INDIRECT TAXES!
 FORMULA (ALL,J,IND)
 COSTINP(J) = SUM(I,IND,DINT(I,J)) + SUM(I,IND,IINT(I,J))
       + SUM(I,IND,TRD(I,J)) + SUM(I,IND,TRI(I,J))
       + LAB(J) + CAP(J) + LAND(J);
```

```
!-------------------------------------------------------------------------------
                    EXPENDITURE AGGREGATES
 -------------------------------------------------------------------------------!

 COEFFICIENT                                                    CTT
! TOTAL CONSUMPTION ! ;
 FORMULA
 CTT = SUM(I,IND,DCON(I)) + SUM(I,IND,ICON(I))
       + SUM(I,IND,TCRD(I)) + SUM(I,IND,TCRI(I));

 COEFFICIENT                                                    INVTT
! TOTAL INVESTMENT !   ;
 FORMULA
 INVTT = SUM(I,IND,DINV(I)) + SUM(I,IND,IINV(I))
       + SUM(I,IND,TIRD(I)) + SUM(I,IND,TIRI(I)) ;

 ZERODIVIDE DEFAULT 0.0;

!*******************************************************************************!
!            SHARE COEFFICIENTS OF THE EQUATION SYSTEM                          !
!*******************************************************************************!

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                              SHCOMI(I,J)
! SHARE OF COMMODITY I IN TOTAL COSTS OF INDUSTRY J !;
FORMULA (ALL,I,IND)(ALL,J,IND)
SHCOMI(I,J) = (DINT(I,J) + IINTS(I,J) + TRD(I,J) +TRI(I,J))
          / COSTINP(J);

COEFFICIENT (ALL,I,IND)                                         SHTINVI(I)
! SHARE OF COMMODITY I IN AGGREGATE INVESTMENT EXPENDITURE !;
FORMULA (ALL,I,IND)
SHTINVI(I) = (DINV(I) + IINVS(I) + TIRD(I) + TIRI(I)) / INVTT;

COEFFICIENT (ALL,I,IND)                                         SHHHCONI(I)
```

```
! SHARE OF COMMODITY I IN AGGREGATE HH CONSUMPTION EXPENDITURE
!;
FORMULA (ALL,I,IND)
SHHHCONI(I) = (DCON(I) + ICONS(I) + TCRD(I) + TCRI(I)) / CTT;

COEFFICIENT (ALL,I,IND)                                    SHGOVCOMI(I)
! SHARE OF COMMODITY I IN AGGREGATE GOVT CONSUMPTION
EXPENDITURE !;
FORMULA (ALL,I,IND)
SHGOVCOMI(I) = (DGOV(I) + IGOVS(I) + TGRD(I) + TGRI(I)) / AG;

COEFFICIENT (ALL,I,IND)                                    SHEXPSCOMI(I)
! SHARE OF COMMODITY I IN AGGREGATE EXPORTS INCL TAXES !;
FORMULA (ALL,I,IND)
SHEXPSCOMI(I) = VALEXP(I) / (SUM(I_,IND,VALEXP(I_)));

COEFFICIENT (ALL,J,IND)                                    SHLABTC(J)
! SHARE OF LABOUR IN TOTAL COSTS OF INDUSTRY J !;
FORMULA (ALL,J,IND)
SHLABTC(J) = LAB(J) / COSTINP(J);

COEFFICIENT (ALL,J,IND)                                    SHCAPTC(J)
! SHARE OF CAPITAL IN TOTAL COSTS OF INDUSTRY J !;
FORMULA (ALL,J,IND)
SHCAPTC(J) = CAP(J) / COSTINP(J);

COEFFICIENT (ALL,J,IND)                                    SHLANDTC(J)
! SHARE OF LAND IN TOTAL COSTS OF INDUSTRY J !;
FORMULA (ALL,J,IND)
SHLANDTC(J) = LAND(J) / COSTINP(J);

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                         SHDOMINTI(I,J)
! SHARE OF DOMESTIC PRODUCTS IN INTERMEDIATE USAGE OF COMMODITY
I BY IND J !;
FORMULA (ALL,I,IND)(ALL,J,IND)
```

```
SHDOMINTI(I,J) = (DINT(I,J) + TRD(I,J)) /
          (DINT(I,J) + IINTS(I,J) + TRD(I,J) + TRI(I,J));

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                    SHIMPINTI(I,J)
! SHARE OF IMPORTS IN INTERMEDIATE USAGE OF COMMODITY I BY IND J !;
FORMULA (ALL,I,IND)(ALL,J,IND)
SHIMPINTI(I,J) = (IINTS(I,J) + TRI(I,J)) /
          (DINT(I,J) + IINTS(I,J) + TRD(I,J) + TRI(I,J));

COEFFICIENT (ALL,I,IND)                               SHDOMINVI(I)
! SHARE OF DOMESTIC PRODUCTS IN INVESTMENT USAGE OF COMMODITY I
!;
FORMULA (ALL,I,IND)
SHDOMINVI(I) = (DINV(I) + TIRD(I)) /
          (DINV(I) + TIRD(I) + IINVS(I) + TIRI(I));

COEFFICIENT (ALL,I,IND)                               SHIMPINVI(I)
! SHARE OF IMPORTS IN INVESTMENT USAGE OF COMMODITY I !;
FORMULA (ALL,I,IND)
SHIMPINVI(I) = (IINVS(I) + TIRI(I)) /
          (DINV(I) + TIRD(I) + IINVS(I) + TIRI(I));

COEFFICIENT (ALL,I,IND)                               SHDOMCONI(I)
! SHARE OF DOMESTIC PRODUCTS IN HOUSEHOLD CONSUMPTION OF
COMMODITY I !;
FORMULA (ALL,I,IND)
SHDOMCONI(I) = (DCON(I) + TCRD(I))/
          (DCON(I) + TCRD(I) + ICONS(I) + TCRI(I));

COEFFICIENT (ALL,I,IND)                               SHIMPCONI(I)
! SHARE OF IMPORTS IN HOUSEHOLD CONSUMPTION OF COMMODITY I !;
FORMULA (ALL,I,IND)
SHIMPCONI(I) = (ICONS(I) + TCRI(I)) /
          (DCON(I) + TCRD(I) + ICONS(I) + TCRI(I));
```

```
COEFFICIENT (ALL,I,IND)                                    SHDGOVI(I)
! SHARE OF DOMESTIC PRODUCTS IN GOVERNMENT CONSUMPTION OF
COMMODITY I !;
FORMULA (ALL,I,IND)
SHDGOVI(I) = (DGOV(I) + TGRD(I)) /
        (DGOV(I) + TGRD(I) + IGOVS(I) + TGRI(I));

COEFFICIENT (ALL,I,IND)                                    SHIGOVSI(I)
! SHARE OF IMPORTS IN GOVERNMENT USAGE OF COMMODITY I !;
FORMULA (ALL,I,IND)
SHIGOVSI(I) = (IGOVS(I) + TGRI(I)) /
        (DGOV(I) + TGRD(I) + IGOVS(I) + TGRI(I));

!_____________________________________________________________

          WRITE STATEMENTS

_____________________________________________________________!

WRITE SHCOMI TO FILE iocoeff HEADER "H001";

   SHTINVI TO FILE iocoeff HEADER "H003";

   SHHHCONI TO FILE iocoeff HEADER "H005";

   SHGOVCOMI TO FILE iocoeff HEADER "H007";

   SHEXPSCOMI TO FILE iocoeff HEADER "H009";

   SHLABTC TO FILE iocoeff HEADER "H012";

   SHCAPTC TO FILE iocoeff HEADER "H013";

   SHLANDTC TO FILE iocoeff HEADER "H014";

   SHDOMINTI TO FILE iocoeff HEADER "S001";
```

```
SHIMPINTI TO FILE iocoeff HEADER "S002";

SHDOMINVI TO FILE iocoeff HEADER "S003";

SHIMPINVI TO FILE iocoeff HEADER "S004";

SHDOMCONI TO FILE iocoeff HEADER "S005";

SHIMPCONI TO FILE iocoeff HEADER "S006";

SHDGOVI TO FILE iocoeff HEADER "S007";

SHIGOVSI TO FILE iocoeff HEADER "S008";
!-------------------------------------------------------------------------------
                        END OF TABLO INPUT
-------------------------------------------------------------------------------!
```

Appendix D: Source Code For Program AVERAGE

```
!*****************************************************************************!
!                                    FILES                                    !
!*****************************************************************************!

FILE INCOEFF1 # initial i/o coefficients for individual region # ;
FILE INCOEFF2 # representative i/o coefficients #;
FILE gen # auxiliary data file for single-region revision procedure #;
FILE (NEW) OUTCOEFF # final i/o coefficients for individual region #  ;
!*****************************************************************************!
!                                    SETS                                     !
!*****************************************************************************!

COEFFICIENT (INTEGER)
 NO_IND # number of sectors #;
READ NO_IND FROM FILE gen HEADER "G001";

SET
 IND # sectors # MAXIMUM SIZE 40 SIZE NO_IND;

SET
 REG # regions # (home, repr);

!*****************************************************************************!
!                                  DATA BASE                                  !
!*****************************************************************************!


!-----------------------------------------------------------------------------
            INCOEFFS1 DATA BASE
 ----------------------------------------------------------------------------!
```

COEFFICIENT (ALL,z,REG) WEIGHTS(z)
! WEIGHTS FOR FILES 1 AND 2 RESPECTIVELY ! ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND) SHCOMI1(I,J)
! SHARE OF COMMODITY I IN TOTAL COSTS OF INDUSTRY J ! ;

COEFFICIENT (ALL,I,IND) SHTINVI1(I)
! SHARE OF COMMODITY I IN AGGREGATE INVESTMENT EXPENDITURE ! ;

COEFFICIENT (ALL,I,IND) SHHHCONI1(I)
! SHARE OF COMMODITY I IN AGGREGATE HH CONSUMPTION EXPENDITURE !
;

COEFFICIENT (ALL,I,IND) SHGOVCOMI1(I)
! SHARE OF COMMODITY I IN AGGREGATE GOVT CONSUMPTION EXPENDITURE ! ;

COEFFICIENT (ALL,I,IND) SHEXPSCOMI1(I)
! SHARE OF COMMODITY I IN AGGREGATE EXPORTS INCL TAXES ! ;

COEFFICIENT (ALL,J,IND) SHLABTC1(J)
! SHARE OF LABOUR IN TOTAL COSTS OF INDUSTRY J ! ;

COEFFICIENT (ALL,J,IND) SHCAPTC1(J)
! SHARE OF CAPITAL IN TOTAL COSTS OF INDUSTRY J ! ;

COEFFICIENT (ALL,J,IND) SHLANDTC1(J)
! SHARE OF LAND IN TOTAL COSTS OF INDUSTRY J ! ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND) SHDOMINTI1(I,J)
! SHARE OF DOMESTIC PRODUCTS IN INTERMEDIATE USAGE OF COMMODITY I BY IND J ! ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND) SHIMPINTI1(I,J)
! SHARE OF IMPORTS IN INTERMEDIATE USAGE OF COMMODITY I BY IND J ! ;

COEFFICIENT (ALL,I,IND) SHDOMINVI1(I)
! SHARE OF DOMESTIC PRODUCTS IN INVESTMENT USAGE OF COMMODITY I !
;

COEFFICIENT (ALL,I,IND) SHIMPINVI1(I)
! SHARE OF IMPORTS IN INVESTMENT USAGE OF COMMODITY I ! ;

COEFFICIENT (ALL,I,IND) SHDOMCONI1(I)
! SHARE OF DOMESTIC PRODUCTS IN HOUSEHOLD CONSUMPTION OF COMMODITY I ! ;

COEFFICIENT (ALL,I,IND) SHIMPCONI1(I)
! SHARE OF IMPORTS IN HOUSEHOLD CONSUMPTION OF COMMODITY I ! ;

COEFFICIENT (ALL,I,IND) SHDGOVI1(I)
! SHARE OF DOMESTIC PRODUCTS IN GOVERNMENT CONSUMPTION OF COMMODITY I ! ;

COEFFICIENT (ALL,I,IND) SHIGOVSI1(I)
! SHARE OF IMPORTS IN GOVERNMENT USAGE OF COMMODITY I ! ;

!----------
INCOEFFS2 DATA BASE
----------!

COEFFICIENT (ALL,I,IND)(ALL,J,IND) SHCOMI2(I,J)
! SHARE OF COMMODITY I IN TOTAL COSTS OF INDUSTRY J ! ;

COEFFICIENT (ALL,I,IND) SHTINVI2(I)
! SHARE OF COMMODITY I IN AGGREGATE INVESTMENT EXPENDITURE ! ;

COEFFICIENT (ALL,I,IND) SHHHCONI2(I)
! SHARE OF COMMODITY I IN AGGREGATE HH CONSUMPTION EXPENDITURE !
;

COEFFICIENT (ALL,I,IND) SHGOVCOMI2(I)
! SHARE OF COMMODITY I IN AGGREGATE GOVT CONSUMPTION EXPENDITURE ! ;

COEFFICIENT (ALL,I,IND) SHEXPSCOMI2(I)
! SHARE OF COMMODITY I IN AGGREGATE EXPORTS INCL TAXES ! ;

COEFFICIENT (ALL,J,IND) SHLABTC2(J)
! SHARE OF LABOUR IN TOTAL COSTS OF INDUSTRY J ! ;

COEFFICIENT (ALL,J,IND) SHCAPTC2(J)
! SHARE OF CAPITAL IN TOTAL COSTS OF INDUSTRY J ! ;

COEFFICIENT (ALL,J,IND) SHLANDTC2(J)
! SHARE OF LAND IN TOTAL COSTS OF INDUSTRY J ! ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND) SHDOMINTI2(I,J)
! SHARE OF DOMESTIC PRODUCTS IN INTERMEDIATE USAGE OF COMMODITY I BY IND J ! ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND) SHIMPINTI2(I,J)
! SHARE OF IMPORTS IN INTERMEDIATE USAGE OF COMMODITY I BY IND J ! ;

COEFFICIENT (ALL,I,IND) SHDOMINVI2(I)
! SHARE OF DOMESTIC PRODUCTS IN INVESTMENT USAGE OF COMMODITY I ! ;

COEFFICIENT (ALL,I,IND) SHIMPINVI2(I)
! SHARE OF IMPORTS IN INVESTMENT USAGE OF COMMODITY I ! ;

COEFFICIENT (ALL,I,IND) SHDOMCONI2(I)
! SHARE OF DOMESTIC PRODUCTS IN HOUSEHOLD CONSUMPTION OF COMMODITY I ! ;

```
COEFFICIENT (ALL,I,IND)                                    SHIMPCONI2(I)
! SHARE OF IMPORTS IN HOUSEHOLD CONSUMPTION OF COMMODITY I ! ;

COEFFICIENT (ALL,I,IND)                                    SHDGOVI2(I)
! SHARE OF DOMESTIC PRODUCTS IN GOVERNMENT CONSUMPTION OF
COMMODITY I ! ;

COEFFICIENT (ALL,I,IND)                                    SHIGOVSI2(I)
! SHARE OF IMPORTS IN GOVERNMENT USAGE OF COMMODITY I ! ;



!*****************************************************************************
          READING DATA BASE
 *****************************************************************************!

READ WEIGHTS FROM FILE GEN HEADER "G002" ;

READ SHCOMI1 FROM FILE INCOEFF1 HEADER "H001" ;

    SHTINVI1 FROM FILE INCOEFF1 HEADER "H003" ;

    SHHHCONI1 FROM FILE INCOEFF1 HEADER "H005" ;

    SHGOVCOMI1 FROM FILE INCOEFF1 HEADER "H007" ;

    SHEXPSCOMI1 FROM FILE INCOEFF1 HEADER "H009" ;

    SHLABTC1 FROM FILE INCOEFF1 HEADER "H012" ;

    SHCAPTC1 FROM FILE INCOEFF1 HEADER "H013" ;

    SHLANDTC1 FROM FILE INCOEFF1 HEADER "H014" ;

    SHDOMINTI1 FROM FILE INCOEFF1 HEADER "S001" ;
```

```
    SHIMPINTI1 FROM FILE INCOEFF1 HEADER "S002" ;

    SHDOMINVI1 FROM FILE INCOEFF1 HEADER "S003" ;

    SHIMPINVI1 FROM FILE INCOEFF1 HEADER "S004" ;

    SHDOMCONI1 FROM FILE INCOEFF1 HEADER "S005" ;

    SHIMPCONI1 FROM FILE INCOEFF1 HEADER "S006" ;

    SHDGOVI1 FROM FILE INCOEFF1 HEADER "S007" ;

    SHIGOVSI1 FROM FILE INCOEFF1 HEADER "S008" ;
READ SHCOMI2 FROM FILE INCOEFF2 HEADER "H001" ;

    SHTINVI2 FROM FILE INCOEFF2 HEADER "H003" ;

    SHHHCONI2 FROM FILE INCOEFF2 HEADER "H005" ;

    SHGOVCOMI2 FROM FILE INCOEFF2 HEADER "H007" ;

    SHEXPSCOMI2 FROM FILE INCOEFF2 HEADER "H009" ;

    SHLABTC2 FROM FILE INCOEFF2 HEADER "H012" ;

    SHCAPTC2 FROM FILE INCOEFF2 HEADER "H013" ;

    SHLANDTC2 FROM FILE INCOEFF2 HEADER "H014" ;

    SHDOMINTI2 FROM FILE INCOEFF2 HEADER "S001" ;

    SHIMPINTI2 FROM FILE INCOEFF2 HEADER "S002" ;
```

```
   SHDOMINVI2 FROM FILE INCOEFF2 HEADER "S003" ;

   SHIMPINVI2 FROM FILE INCOEFF2 HEADER "S004" ;

   SHDOMCONI2 FROM FILE INCOEFF2 HEADER "S005" ;

   SHIMPCONI2 FROM FILE INCOEFF2 HEADER "S006" ;

   SHDGOVI2 FROM FILE INCOEFF2 HEADER "S007" ;

   SHIGOVSI2 FROM FILE INCOEFF2 HEADER "S008" ;


!*************************************************************************
          WIDELY USED DATABASE AGGREGATES
*************************************************************************!

  ZERODIVIDE DEFAULT 0.0;

!-------------------------------------------------------------------------
          PRODUCTION AGGREGATES
 ------------------------------------------------------------------------!

 COEFFICIENT  (ALL,J,IND)                                    TOTCOSTS1(J)
! SUM OF TOTAL COSTS SHARES OF ALL INPUTS TO INDUSTRY ! ;
 FORMULA (ALL,J,IND)
 TOTCOSTS1(J) = SUM(I,IND,SHCOMI1(I,J))
       + SHLABTC1(J) + SHCAPTC1(J) + SHLANDTC1(J) ;

 COEFFICIENT  (ALL,J,IND)                                    TOTCOSTS2(J)
! SUM OF TOTAL COSTS SHARES OF ALL INPUTS TO INDUSTRY ! ;
 FORMULA (ALL,J,IND)
 TOTCOSTS2(J) = SUM(I,IND,SHCOMI2(I,J))
       + SHLABTC2(J) + SHCAPTC2(J) + SHLANDTC2(J) ;
```

```
!-------------------------------------------------------------------------------
            EXPENDITURE AGGREGATES
-------------------------------------------------------------------------------!

 COEFFICIENT                                                          CTT1
! TOTAL CONSUMPTION ! ;
 FORMULA
 CTT1 = SUM(I,IND,SHHHCONI1(I)) ;

 COEFFICIENT                                                          INVTT1
! TOTAL INVESTMENT !   ;
 FORMULA
 INVTT1 = SUM(I,IND,SHTINVI1(I)) ;

 COEFFICIENT                                                          GOV1
! TOTAL GOVERNMENT CONSUMPTION ! ;
 FORMULA
 GOV1 = SUM(I,IND,SHGOVCOMI1(I)) ;

 COEFFICIENT                                                          EXPTS1
! TOTAL INVESTMENT !   ;
 FORMULA
 EXPTS1 = SUM(I,IND,SHEXPSCOMI1(I)) ;


 COEFFICIENT                                                          CTT2
! TOTAL CONSUMPTION ! ;
 FORMULA
 CTT2 = SUM(I,IND,SHHHCONI2(I)) ;

 COEFFICIENT                                                          INVTT2
! TOTAL INVESTMENT !   ;
 FORMULA
 INVTT2 = SUM(I,IND,SHTINVI2(I)) ;
```

```
 COEFFICIENT                                                    GOV2
! TOTAL GOVERNMENT CONSUMPTION ! ;
 FORMULA
 GOV2 = SUM(I,IND,SHGOVCOMI2(I)) ;

 COEFFICIENT                                                    EXPTS2
! TOTAL INVESTMENT !    ;
 FORMULA
 EXPTS2 = SUM(I,IND,SHEXPSCOMI2(I)) ;


COEFFICIENT (ALL,I,IND)(ALL,J,IND)                              TTINTI1(I,J)
! SUM OF DOMESTIC AND IMPORTS IN INTERMEDIATE USAGE OF
COMMODITY I BY IND J ! ;
FORMULA (ALL,I,IND)(ALL,J,IND)
TTINTI1(I,J) = SHDOMINTI1(I,J) + SHIMPINTI1(I,J) ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                              TTINTI2(I,J)
! SUM OF DOMESTIC AND IMPORTS IN INTERMEDIATE USAGE OF
COMMODITY I BY IND J ! ;
FORMULA (ALL,I,IND)(ALL,J,IND)
TTINTI2(I,J) = SHDOMINTI2(I,J) + SHIMPINTI2(I,J) ;

COEFFICIENT (ALL,I,IND)                                         TTINVI1(I)
! SHARE OF DOMESTIC + IMPORTS PRODUCTS IN INVESTMENT USAGE OF
COMMODITY I ! ;
FORMULA (ALL,I,IND)
TTINVI1(I) = SHDOMINVI1(I) + SHIMPINVI1(I) ;

COEFFICIENT (ALL,I,IND)                                         TTINVI2(I)
! SHARE OF DOMESTIC + IMPORTS IN INVESTMENT USAGE OF COMMODITY I !
;
FORMULA (ALL,I,IND)
TTINVI2(I) = SHDOMINVI2(I) + SHIMPINVI2(I) ;
```

```
COEFFICIENT (ALL,I,IND)                                    TTCONI1(I)
! SHARE OF DOMESTIC + IMPORTS IN HOUSEHOLD CONSUMPTION OF
COMMODITY I ! ;
FORMULA (ALL,I,IND)
TTCONI1(I) = SHDOMCONI1(I) + SHIMPCONI1(I) ;

COEFFICIENT (ALL,I,IND)                                    TTCONI2(I)
! SHARE OF DOMESTIC + IMPORTS IN HOUSEHOLD CONSUMPTION OF
COMMODITY I ! ;
FORMULA (ALL,I,IND)
TTCONI2(I) = SHDOMCONI2(I) + SHIMPCONI2(I) ;

COEFFICIENT (ALL,I,IND)                                    TTGOVI1(I)
! SHARE OF DOMESTIC + IMPORTS IN GOVERNMENT CONSUMPTION OF
COMMODITY I ! ;
FORMULA (ALL,I,IND)
TTGOVI1(I) = SHDGOVI1(I) + SHIGOVSI1(I) ;

COEFFICIENT (ALL,I,IND)                                    TTGOVI2(I)
! SHARE OF DOMESTIC + IMPORTS IN GOVERNMENT USAGE OF COMMODITY I
! ;
FORMULA (ALL,I,IND)
TTGOVI2(I) = SHDGOVI2(I) + SHIGOVSI2(I) ;


 ZERODIVIDE DEFAULT 0.0;

!-------------------------------------------------------------------------------
        PRODUCTION FACTORS
 -------------------------------------------------------------------------------!

 COEFFICIENT  (ALL,J,IND)                                  FCOSTS1(J)
! SUM OF TOTAL COSTS SHARES OF ALL INPUTS TO INDUSTRY ! ;
 FORMULA (ALL,J,IND)
 FCOSTS1(J) = TOTCOSTS1(J) / TOTCOSTS1(J) ;
```

```
 COEFFICIENT  (ALL,J,IND)                                        FCOSTS2(J)
! SUM OF TOTAL COSTS SHARES OF ALL INPUTS TO INDUSTRY ! ;
 FORMULA (ALL,J,IND)
 FCOSTS2(J) = TOTCOSTS2(J) / TOTCOSTS2(J) ;

!-------------------------------------------------------------------------------
          EXPENDITURE FACTORS
 -------------------------------------------------------------------------------!

 COEFFICIENT                                                     FCTT1
! TOTAL CONSUMPTION ! ;
 FORMULA
 FCTT1 = CTT1 / CTT1 ;

 COEFFICIENT                                                     FINVTT1
! TOTAL INVESTMENT !   ;
 FORMULA
 FINVTT1 = INVTT1 / INVTT1 ;

 COEFFICIENT                                                     FGOV1
! TOTAL GOVERNMENT CONSUMPTION ! ;
 FORMULA
 FGOV1 = GOV1 / GOV1 ;

 COEFFICIENT                                                     FEXPTS1
! TOTAL INVESTMENT !   ;
 FORMULA
 FEXPTS1 = EXPTS1 / EXPTS1 ;


 COEFFICIENT                                                     FCTT2
! TOTAL CONSUMPTION ! ;
 FORMULA
 FCTT2 = CTT2 / CTT2 ;
```

```
 COEFFICIENT                                                    FINVTT2
! TOTAL INVESTMENT !    ;
 FORMULA
 FINVTT2 = INVTT2 / INVTT2 ;

 COEFFICIENT                                                    FGOV2
! TOTAL GOVERNMENT CONSUMPTION ! ;
 FORMULA
 FGOV2 = GOV2 / GOV2 ;

 COEFFICIENT                                                    FEXPTS2
! TOTAL INVESTMENT !    ;
 FORMULA
 FEXPTS2 = EXPTS2 / EXPTS2 ;


COEFFICIENT (ALL,I,IND)(ALL,J,IND)                              FTTINTI1(I,J)
! SUM OF DOMESTIC AND IMPORTS IN INTERMEDIATE USAGE OF
COMMODITY I BY IND J ! ;
FORMULA (ALL,I,IND)(ALL,J,IND)
FTTINTI1(I,J) = TTINTI1(I,J) / TTINTI1(I,J) ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                              FTTINTI2(I,J)
! SUM OF DOMESTIC AND IMPORTS IN INTERMEDIATE USAGE OF
COMMODITY I BY IND J ! ;
FORMULA (ALL,I,IND)(ALL,J,IND)
FTTINTI2(I,J) = TTINTI2(I,J) / TTINTI2(I,J) ;

COEFFICIENT (ALL,I,IND)                                         FTTINVI1(I)
! SHARE OF DOMESTIC + IMPORTS PRODUCTS IN INVESTMENT USAGE OF
COMMODITY I ! ;
FORMULA (ALL,I,IND)
FTTINVI1(I) = TTINVI1(I) / TTINVI1(I) ;
```

```
COEFFICIENT (ALL,I,IND)                                    FTTINVI2(I)
! SHARE OF DOMESTIC + IMPORTS IN INVESTMENT USAGE OF COMMODITY I !
;
FORMULA (ALL,I,IND)
FTTINVI2(I) = TTINVI2(I) / TTINVI2(I) ;

COEFFICIENT (ALL,I,IND)                                    FTTCONI1(I)
! SHARE OF DOMESTIC + IMPORTS IN HOUSEHOLD CONSUMPTION OF
COMMODITY I ! ;
FORMULA (ALL,I,IND)
FTTCONI1(I) = TTCONI1(I) / TTCONI1(I) ;

COEFFICIENT (ALL,I,IND)                                    FTTCONI2(I)
! SHARE OF DOMESTIC + IMPORTS IN HOUSEHOLD CONSUMPTION OF
COMMODITY I ! ;
FORMULA (ALL,I,IND)
FTTCONI2(I) = TTCONI2(I) / TTCONI2(I) ;

COEFFICIENT (ALL,I,IND)                                    FTTGOVI1(I)
! SHARE OF DOMESTIC + IMPORTS IN GOVERNMENT CONSUMPTION OF
COMMODITY I ! ;
FORMULA (ALL,I,IND)
FTTGOVI1(I) = TTGOVI1(I) / TTGOVI1(I) ;

COEFFICIENT (ALL,I,IND)                                    FTTGOVI2(I)
! SHARE OF DOMESTIC + IMPORTS IN GOVERNMENT USAGE OF COMMODITY I
! ;
FORMULA (ALL,I,IND)
FTTGOVI2(I) = TTGOVI2(I) / TTGOVI2(I) ;

 ZERODIVIDE DEFAULT 0.0 ;

!*************************************************************************!
!              WEIGHT COEFFICIENTS                                        !
!*************************************************************************!
```

```
ZERODIVIDE DEFAULT 0.0;

!-------------------------------------------------------------------------------
        PRODUCTION WEIGHTS
-------------------------------------------------------------------------------!

COEFFICIENT                                                    W1 ;
COEFFICIENT                                                    W2 ;

FORMULA  W1 = WEIGHTS("home") ;
FORMULA  W2 = WEIGHTS("repr") ;

COEFFICIENT  (ALL,J,IND)                                       WCOSTS1(J)
! SUM OF TOTAL COSTS SHARES OF ALL INPUTS TO INDUSTRY ! ;
FORMULA (ALL,J,IND)
WCOSTS1(J) = FCOSTS1(J) * W1 ;

COEFFICIENT  (ALL,J,IND)                                       WCOSTS2(J)
! SUM OF TOTAL COSTS SHARES OF ALL INPUTS TO INDUSTRY ! ;
FORMULA (ALL,J,IND)
WCOSTS2(J) = FCOSTS2(J) * W2 ;

COEFFICIENT  (ALL,J,IND)                                       WCOSTST(J)
! SUM OF TOTAL COSTS SHARES OF ALL INPUTS TO INDUSTRY ! ;
FORMULA (ALL,J,IND)
WCOSTST(J) = WCOSTS1(J) + WCOSTS2(J) ;

COEFFICIENT  (ALL,J,IND)                                       UCOSTS1(J)
! SUM OF TOTAL COSTS SHARES OF ALL INPUTS TO INDUSTRY ! ;
FORMULA (ALL,J,IND)
UCOSTS1(J) = WCOSTS1(J) / WCOSTST(J) ;

COEFFICIENT  (ALL,J,IND)                                       UCOSTS2(J)
```

```
! SUM OF TOTAL COSTS SHARES OF ALL INPUTS TO INDUSTRY ! ;
 FORMULA (ALL,J,IND)
 UCOSTS2(J) = WCOSTS2(J) / WCOSTST(J) ;

!-------------------------------------------------------------------------------
            EXPENDITURE WEIGHTS
 -------------------------------------------------------------------------------!

 COEFFICIENT                                                    WCTT1
! TOTAL CONSUMPTION ! ;
 FORMULA
 WCTT1 = FCTT1 * W1 ;

 COEFFICIENT                                                    WINVTT1
! TOTAL INVESTMENT !   ;
 FORMULA
 WINVTT1 = FINVTT1 * W1 ;

 COEFFICIENT                                                    WGOV1
! TOTAL GOVERNMENT CONSUMPTION ! ;
 FORMULA
 WGOV1 = FGOV1 * W1 ;

 COEFFICIENT                                                    WEXPTS1
! TOTAL INVESTMENT !   ;
 FORMULA
 WEXPTS1 = FEXPTS1 * W1 ;


 COEFFICIENT                                                    WCTT2
! TOTAL CONSUMPTION ! ;
 FORMULA
 WCTT2 = FCTT2 * W2 ;

 COEFFICIENT                                                    WINVTT2
```

```
! TOTAL INVESTMENT !    ;
 FORMULA
 WINVTT2 = FINVTT2 * W2 ;

 COEFFICIENT                                                    WGOV2
! TOTAL GOVERNMENT CONSUMPTION ! ;
 FORMULA
 WGOV2 = FGOV2 * W2 ;

 COEFFICIENT                                                    WEXPTS2
! TOTAL INVESTMENT !    ;
 FORMULA
 WEXPTS2 = FEXPTS2 * W2 ;


 COEFFICIENT                                                    WCTTT
! TOTAL CONSUMPTION ! ;
 FORMULA
 WCTTT = WCTT1 + WCTT2 ;

 COEFFICIENT                                                    WINVTTT
! TOTAL INVESTMENT !    ;
 FORMULA
 WINVTTT = WINVTT1 + WINVTT2 ;

 COEFFICIENT                                                    WGOVT
! TOTAL GOVERNMENT CONSUMPTION ! ;
 FORMULA
 WGOVT = WGOV1 + WGOV2 ;

 COEFFICIENT                                                    WEXPTST
! TOTAL INVESTMENT !    ;
 FORMULA
 WEXPTST = WEXPTS1 + WEXPTS2 ;
```

```
 COEFFICIENT                                                    UCTT1
! TOTAL CONSUMPTION ! ;
 FORMULA
 UCTT1 = WCTT1 / WCTTT ;

 COEFFICIENT                                                    UINVTT1
! TOTAL INVESTMENT !    ;
 FORMULA
 UINVTT1 = WINVTT1 / WINVTTT ;

 COEFFICIENT                                                    UGOV1
! TOTAL GOVERNMENT CONSUMPTION ! ;
 FORMULA
 UGOV1 = WGOV1 / WGOVT ;

 COEFFICIENT                                                    UEXPTS1
! TOTAL INVESTMENT !    ;
 FORMULA
 UEXPTS1 = WEXPTS1 / WEXPTST ;


 COEFFICIENT                                                    UCTT2
! TOTAL CONSUMPTION ! ;
 FORMULA
 UCTT2 = WCTT2 / WCTTT ;

 COEFFICIENT                                                    UINVTT2
! TOTAL INVESTMENT !    ;
 FORMULA
 UINVTT2 = WINVTT2 / WINVTTT ;

 COEFFICIENT                                                    UGOV2
! TOTAL GOVERNMENT CONSUMPTION ! ;
 FORMULA
```

```
UGOV2 = WGOV2 / WGOVT ;

COEFFICIENT                                                    UEXPTS2
! TOTAL INVESTMENT !    ;
FORMULA
UEXPTS2 = WEXPTS2 / WEXPTST ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                             WTTINTI1(I,J)
! SUM OF DOMESTIC AND IMPORTS IN INTERMEDIATE USAGE OF
COMMODITY I BY IND J ! ;
FORMULA (ALL,I,IND)(ALL,J,IND)
WTTINTI1(I,J) = FTTINTI1(I,J) * W1 ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                             WTTINTI2(I,J)
! SUM OF DOMESTIC AND IMPORTS IN INTERMEDIATE USAGE OF
COMMODITY I BY IND J ! ;
FORMULA (ALL,I,IND)(ALL,J,IND)
WTTINTI2(I,J) = FTTINTI2(I,J) * W2 ;


COEFFICIENT (ALL,I,IND)(ALL,J,IND)                             WTTINTIT(I,J)
! SUM OF DOMESTIC AND IMPORTS IN INTERMEDIATE USAGE OF
COMMODITY I BY IND J ! ;
FORMULA (ALL,I,IND)(ALL,J,IND)
WTTINTIT(I,J) = WTTINTI1(I,J) + WTTINTI2(I,J) ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                             UTTINTI1(I,J)
! SUM OF DOMESTIC AND IMPORTS IN INTERMEDIATE USAGE OF
COMMODITY I BY IND J ! ;
FORMULA (ALL,I,IND)(ALL,J,IND)
UTTINTI1(I,J) = WTTINTI1(I,J) / WTTINTIT(I,J) ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                             UTTINTI2(I,J)
! SUM OF DOMESTIC AND IMPORTS IN INTERMEDIATE USAGE OF
COMMODITY I BY IND J ! ;
```

```
FORMULA (ALL,I,IND)(ALL,J,IND)
UTTINTI2(I,J) = WTTINTI2(I,J) / WTTINTIT(I,J) ;


COEFFICIENT (ALL,I,IND)                                    WTTINVI1(I)
! SHARE OF DOMESTIC + IMPORTS PRODUCTS IN INVESTMENT USAGE OF
COMMODITY I ! ;
FORMULA (ALL,I,IND)
WTTINVI1(I) = FTTINVI1(I) * W1 ;

COEFFICIENT (ALL,I,IND)                                    WTTINVI2(I)
! SHARE OF DOMESTIC + IMPORTS IN INVESTMENT USAGE OF COMMODITY I !
;
FORMULA (ALL,I,IND)
WTTINVI2(I) = FTTINVI2(I) * W2 ;

COEFFICIENT (ALL,I,IND)                                    WTTINVIT(I)
! SHARE OF DOMESTIC + IMPORTS PRODUCTS IN INVESTMENT USAGE OF
COMMODITY I ! ;
FORMULA (ALL,I,IND)
WTTINVIT(I) = WTTINVI1(I) + WTTINVI2(I) ;

COEFFICIENT (ALL,I,IND)                                    UTTINVI1(I)
! SUM OF DOMESTIC AND IMPORTS IN INTERMEDIATE USAGE OF
COMMODITY I BY IND J ! ;
FORMULA (ALL,I,IND)
UTTINVI1(I) = WTTINVI1(I) / WTTINVIT(I) ;

COEFFICIENT (ALL,I,IND)                                    UTTINVI2(I)
! SUM OF DOMESTIC AND IMPORTS IN INTERMEDIATE USAGE OF
COMMODITY I BY IND J ! ;
FORMULA (ALL,I,IND)
UTTINVI2(I) = WTTINVI2(I) / WTTINVIT(I) ;
```

```
COEFFICIENT (ALL,I,IND)                                    WTTCONI1(I)
! SHARE OF DOMESTIC + IMPORTS IN HOUSEHOLD CONSUMPTION OF
COMMODITY I ! ;
FORMULA (ALL,I,IND)
WTTCONI1(I) = FTTCONI1(I) * W1 ;

COEFFICIENT (ALL,I,IND)                                    WTTCONI2(I)
! SHARE OF DOMESTIC + IMPORTS IN HOUSEHOLD CONSUMPTION OF
COMMODITY I ! ;
FORMULA (ALL,I,IND)
WTTCONI2(I) = FTTCONI2(I) * W2 ;

COEFFICIENT (ALL,I,IND)                                    WTTCONIT(I)
! SHARE OF DOMESTIC + IMPORTS IN HOUSEHOLD CONSUMPTION OF
COMMODITY I ! ;
FORMULA (ALL,I,IND)
WTTCONIT(I) = WTTCONI1(I) + WTTCONI2(I) ;

COEFFICIENT (ALL,I,IND)                                    UTTCONI1(I)
! SUM OF DOMESTIC AND IMPORTS IN INTERMEDIATE USAGE OF
COMMODITY I BY IND J ! ;
FORMULA (ALL,I,IND)
UTTCONI1(I) = WTTCONI1(I) / WTTCONIT(I) ;

COEFFICIENT (ALL,I,IND)                                    UTTCONI2(I)
! SUM OF DOMESTIC AND IMPORTS IN INTERMEDIATE USAGE OF
COMMODITY I BY IND J ! ;
FORMULA (ALL,I,IND)
UTTCONI2(I) = WTTCONI2(I) / WTTCONIT(I) ;


COEFFICIENT (ALL,I,IND)                                    WTTGOVI1(I)
! SHARE OF DOMESTIC + IMPORTS IN GOVERNMENT CONSUMPTION OF
COMMODITY I ! ;
FORMULA (ALL,I,IND)
```

```
WTTGOVI1(I) = FTTGOVI1(I) * W1 ;

COEFFICIENT (ALL,I,IND)                                    WTTGOVI2(I)
! SHARE OF DOMESTIC + IMPORTS IN GOVERNMENT USAGE OF COMMODITY I
! ;
FORMULA (ALL,I,IND)
WTTGOVI2(I) = FTTGOVI2(I) * W2 ;

COEFFICIENT (ALL,I,IND)                                    WTTGOVIT(I)
! SHARE OF DOMESTIC + IMPORTS IN GOVERNMENT CONSUMPTION OF
COMMODITY I ! ;
FORMULA (ALL,I,IND)
WTTGOVIT(I) = WTTGOVI1(I) + WTTGOVI2(I) ;

COEFFICIENT (ALL,I,IND)                                    UTTGOVI1(I)
! SUM OF DOMESTIC AND IMPORTS IN INTERMEDIATE USAGE OF
COMMODITY I BY IND J ! ;
FORMULA (ALL,I,IND)
UTTGOVI1(I) = WTTGOVI1(I) / WTTGOVIT(I) ;

COEFFICIENT (ALL,I,IND)                                    UTTGOVI2(I)
! SUM OF DOMESTIC AND IMPORTS IN INTERMEDIATE USAGE OF
COMMODITY I BY IND J ! ;
FORMULA (ALL,I,IND)
UTTGOVI2(I) = WTTGOVI2(I) / WTTGOVIT(I) ;

 ZERODIVIDE DEFAULT 0.0 ;


!*************************************************************************!
!       WEIGHTED SHARE COEFFICIENTS OF THE EQUATION SYSTEM                !
!*************************************************************************!

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                         SHCOMI(I,J)
! SHARE OF COMMODITY I IN TOTAL COSTS OF INDUSTRY J ! ;
```

```
FORMULA (ALL,I,IND)(ALL,J,IND)
SHCOMI(I,J) = SHCOMI1(I,J) *UCOSTS1(J) + SHCOMI2(I,J) *UCOSTS2(J) ;

COEFFICIENT (ALL,I,IND)                                    SHTINVI(I)
! SHARE OF COMMODITY I IN AGGREGATE INVESTMENT EXPENDITURE ! ;
FORMULA (ALL,I,IND)
SHTINVI(I) = SHTINVI1(I) * UINVTT1 + SHTINVI2(I) * UINVTT2 ;

COEFFICIENT (ALL,I,IND)                                    SHHHCONI(I)
! SHARE OF COMMODITY I IN AGGREGATE HH CONSUMPTION EXPENDITURE !
;
FORMULA (ALL,I,IND)
SHHHCONI(I) = SHHHCONI1(I) * UCTT1 + SHHHCONI2(I) * UCTT2 ;

COEFFICIENT (ALL,I,IND)                                    SHGOVCOMI(I)
! SHARE OF COMMODITY I IN AGGREGATE GOVT CONSUMPTION
EXPENDITURE ! ;
FORMULA (ALL,I,IND)
SHGOVCOMI(I) = SHGOVCOMI1(I) * UGOV1 + SHGOVCOMI2(I) * UGOV2 ;

COEFFICIENT (ALL,I,IND)                                    SHEXPSCOMI(I)
! SHARE OF COMMODITY I IN AGGREGATE EXPORTS INCL TAXES ! ;
FORMULA (ALL,I,IND)
SHEXPSCOMI(I) = SHEXPSCOMI1(I) * UEXPTS1 + SHEXPSCOMI2(I)*UEXPTS2 ;

COEFFICIENT (ALL,J,IND)                                    SHLABTC(J)
! SHARE OF LABOUR IN TOTAL COSTS OF INDUSTRY J ! ;
FORMULA (ALL,J,IND)
SHLABTC(J) = SHLABTC1(J) * UCOSTS1(J) + SHLABTC2(J) * UCOSTS2(J) ;

COEFFICIENT (ALL,J,IND)                                    SHCAPTC(J)
! SHARE OF CAPITAL IN TOTAL COSTS OF INDUSTRY J ! ;
FORMULA (ALL,J,IND)
SHCAPTC(J) = SHCAPTC1(J) * UCOSTS1(J) + SHCAPTC2(J) * UCOSTS2(J) ;
```

```
COEFFICIENT (ALL,J,IND)                                    SHLANDTC(J)
! SHARE OF LAND IN TOTAL COSTS OF INDUSTRY J ! ;
FORMULA (ALL,J,IND)
SHLANDTC(J) = SHLANDTC1(J) * UCOSTS1(J) + SHLANDTC2(J) * UCOSTS2(J) ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                         SHDOMINTI(I,J)
! SHARE OF DOMESTIC PRODUCTS IN INTERMEDIATE USAGE OF COMMODITY
I BY IND J ! ;
FORMULA (ALL,I,IND)(ALL,J,IND)
SHDOMINTI(I,J) = SHDOMINTI1(I,J) * UTTINTI1(I,J)  +
        SHDOMINTI2(I,J) * UTTINTI2(I,J) ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                         SHIMPINTI(I,J)
! SHARE OF IMPORTS IN INTERMEDIATE USAGE OF COMMODITY I BY IND J ! ;
FORMULA (ALL,I,IND)(ALL,J,IND)
SHIMPINTI(I,J) = SHIMPINTI1(I,J) * UTTINTI1(I,J)  +
        SHIMPINTI2(I,J) * UTTINTI2(I,J) ;

COEFFICIENT (ALL,I,IND)                                    SHDOMINVI(I)
! SHARE OF DOMESTIC PRODUCTS IN INVESTMENT USAGE OF COMMODITY I !
;
FORMULA (ALL,I,IND)
SHDOMINVI(I) = SHDOMINVI1(I) * UTTINVI1(I)  +
       SHDOMINVI2(I) * UTTINVI2(I)  ;

COEFFICIENT (ALL,I,IND)                                    SHIMPINVI(I)
! SHARE OF IMPORTS IN INVESTMENT USAGE OF COMMODITY I ! ;
FORMULA (ALL,I,IND)
SHIMPINVI(I) = SHIMPINVI1(I) * UTTINVI1(I)  +
       SHIMPINVI2(I) * UTTINVI2(I) ;

COEFFICIENT (ALL,I,IND)                                    SHDOMCONI(I)
! SHARE OF DOMESTIC PRODUCTS IN HOUSEHOLD CONSUMPTION OF
COMMODITY I ! ;
FORMULA (ALL,I,IND)
```

```
SHDOMCONI(I) = SHDOMCONI1(I) * UTTCONI1(I) +
      SHDOMCONI2(I) * UTTCONI2(I) ;

COEFFICIENT (ALL,I,IND)                                    SHIMPCONI(I)
! SHARE OF IMPORTS IN HOUSEHOLD CONSUMPTION OF COMMODITY I ! ;
FORMULA (ALL,I,IND)
SHIMPCONI(I) = SHIMPCONI1(I) * UTTCONI1(I) +
      SHIMPCONI2(I) * UTTCONI2(I) ;

COEFFICIENT (ALL,I,IND)                                    SHDGOVI(I)
! SHARE OF DOMESTIC PRODUCTS IN GOVERNMENT CONSUMPTION OF
COMMODITY I ! ;
FORMULA (ALL,I,IND)
SHDGOVI(I) = SHDGOVI1(I) * UTTGOVI1(I) +
      SHDGOVI2(I) * UTTGOVI2(I) ;

COEFFICIENT (ALL,I,IND)                                    SHIGOVSI(I)
! SHARE OF IMPORTS IN GOVERNMENT USAGE OF COMMODITY I ! ;
FORMULA (ALL,I,IND)
SHIGOVSI(I) = SHIGOVSI1(I) * UTTGOVI1(I) +
      SHIGOVSI2(I) * UTTGOVI2(I) ;
!_____________________________________________________________

      WRITE STATEMENTS
_____________________________________________________________!

WRITE SHCOMI TO FILE OUTCOEFF HEADER "H001" ;

   SHTINVI TO FILE OUTCOEFF HEADER "H003" ;

   SHHHCONI TO FILE OUTCOEFF HEADER "H005" ;

   SHGOVCOMI TO FILE OUTCOEFF HEADER "H007" ;

   SHEXPSCOMI TO FILE OUTCOEFF HEADER "H009" ;
```

```
SHLABTC TO FILE OUTCOEFF HEADER "H012" ;

SHCAPTC TO FILE OUTCOEFF HEADER "H013" ;

SHLANDTC TO FILE OUTCOEFF HEADER "H014" ;

SHDOMINTI TO FILE OUTCOEFF HEADER "S001" ;

SHIMPINTI TO FILE OUTCOEFF HEADER "S002" ;

SHDOMINVI TO FILE OUTCOEFF HEADER "S003" ;

SHIMPINVI TO FILE OUTCOEFF HEADER "S004" ;

SHDOMCONI TO FILE OUTCOEFF HEADER "S005" ;

SHIMPCONI TO FILE OUTCOEFF HEADER "S006" ;

SHDGOVI TO FILE OUTCOEFF HEADER "S007" ;

SHIGOVSI TO FILE OUTCOEFF HEADER "S008" ;
!-------------------------------------------------------------------------------
                    END OF TABLO INPUT
-------------------------------------------------------------------------------!
```

Appendix E: Source Code For Program PROJECT

```
!*****************************************************************************!
!                                  FILES                                      !
!*****************************************************************************!

FILE table # FILE CONTAINING ALL I/O INFORMATION # ;
FILE coeff # FILE CONTAINING I/O SHARE COEFFICIENTS  # ;
FILE gen # auxiliary data file for single-region revision procedure #;

!*****************************************************************************!
!                                  SETS                                       !
!*****************************************************************************!

COEFFICIENT (INTEGER)
 NO_IND # number of sectors #;
READ NO_IND FROM FILE gen HEADER                    "G001";

SET
 IND # sectors # MAXIMUM SIZE 40 SIZE NO_IND;

!*****************************************************************************!
!                              DATA BASE                                      !
!*****************************************************************************!

!_____________________________________________________________________________!
!          COEFFICIENT DECLARATIONS AND READ STATEMENTS                       !
!_____________________________________________________________________________!


COEFFICIENT (ALL,I,IND)(ALL,J,IND)                  DINT(I,J)
! INITIAL INTERMEDIATE USAGE (DOMESTIC) OF COMMODITY I BY INDUSTRY
J ! ;
READ (ALL,I,IND)(ALL,J,IND) DINT(I,J)
        FROM FILE table HEADER                      "AI01" ;
```

```
COEFFICIENT (ALL,I,IND)(ALL,J,IND)                    IINTS(I,J)
! INITIAL INTERMEDIATE USAGE (IMPORTED) OF COMMODITY I BY INDUSTRY
J ! ;
READ (ALL,I,IND)(ALL,J,IND) IINTS(I,J)
        FROM FILE table HEADER                          "AI02" ;

COEFFICIENT (ALL,I,IND)                                 DINV(I)
! INVESTMENT USAGE (DOMESTIC) ! ;
READ (ALL,I,IND) DINV(I)
        FROM FILE table HEADER                          "AI03" ;

COEFFICIENT (ALL,I,IND)                                 IINVS(I)
! INVESTMENT USAGE (IMPORTED) ! ;
READ (ALL,I,IND) IINVS(I)
        FROM FILE table HEADER                          "AI04" ;

COEFFICIENT (ALL,I,IND)                                 DCON(I)
! CONSUMPTION USAGE DOMESTIC ! ;
READ (ALL,I,IND) DCON(I)
        FROM FILE table HEADER                          "AI05" ;

COEFFICIENT (ALL,I,IND)                                 ICONS(I)
! CONSUMPTION USAGE IMPORTED ! ;
READ (ALL,I,IND) ICONS(I)
        FROM FILE table HEADER                          "AI06" ;

COEFFICIENT (ALL,I,IND)                                 DGOV(I)
! GOVERNMENT USAGE OF DOMESTIC COMMODITIES ! ;
READ (ALL,I,IND) DGOV(I)
        FROM FILE table HEADER                          "AI07" ;

COEFFICIENT (ALL,I,IND)                                 IGOVS(I)
! GOVERNMENT USAGE OF IMPORTED COMMODITIES ! ;
READ (ALL,I,IND) IGOVS(I)
```

```
        FROM FILE table HEADER                                  "AI08" ;

COEFFICIENT (ALL,I,IND)                                         DSTOCK(I)
! CHANGE IN DOMESTIC STOCKS ! ;
READ (ALL,I,IND) DSTOCK(I)
        FROM FILE table HEADER                                  "AI09" ;

COEFFICIENT (ALL,I,IND)                                         ISTOCK(I)
! IMPORTED CHANGE IN STOCKS ! ;
READ (ALL,I,IND) ISTOCK(I)
        FROM FILE table HEADER                                  "AI10" ;

COEFFICIENT (ALL,I,IND)                                         EXPS(I)
! EXPORTS OF COMMODITY I ! ;
READ (ALL,I,IND) EXPS(I)
        FROM FILE table HEADER                                  "AI11" ;

COEFFICIENT (ALL,J,IND)                                         TSR(J)
! NON COMMODITY INDIRECT TAXES NET OF SUBSIDIES ! ;
READ (ALL,J,IND) TSR(J)
        FROM FILE table HEADER                                  "AI12" ;

COEFFICIENT (ALL,J,IND)                                         LAB(J)
! USAGE OF LABOUR ! ;
READ (ALL,J,IND) LAB(J)
        FROM FILE  table  HEADER                                "AI13" ;

COEFFICIENT (ALL,J,IND)                                         CAP(J)
! USAGE OF CAPITAL ! ;
READ (ALL,J,IND) CAP(J)
        FROM FILE  table  HEADER                                "AI14" ;

COEFFICIENT (ALL,J,IND)                                         LAND(J)
! USAGE OF LAND ! ;
READ (ALL,J,IND) LAND(J)
```

```
        FROM FILE table HEADER                                  "AI15" ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                              TRD(I,J)
! COMMODITY TAXES ON DOMESTICALLY PRODUCED GOODS USED AS
INTERMEDIATE USAGE ! ;
READ (ALL,I,IND)(ALL,J,IND) TRD(I,J)
        FROM FILE table HEADER                                  "AI16" ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                              TRI(I,J)
! COMMODITY TAXES ON IMPORTS USED AS INTERMEDIATE USAGE ! ;
READ (ALL,I,IND)(ALL,J,IND) TRI(I,J)
        FROM FILE table HEADER                                  "AI17" ;

COEFFICIENT (ALL,I,IND)                                         TCRD(I)
! COMMODITY TAXES ON DOMESTICALLY PRODUCED GOODS SOLD AS
CONSUMPTION ! ;
READ (ALL,I,IND) TCRD(I)
        FROM FILE table HEADER                                  "AI18" ;

COEFFICIENT (ALL,I,IND)                                         TCRI(I)
! COMMODITY TAXES ON IMPORTS SOLD AS CONSUMPTION ! ;
READ (ALL,I,IND) TCRI(I)
        FROM FILE table HEADER                                  "AI19" ;

COEFFICIENT (ALL,I,IND)                                         TIRD(I)
! COMMODITY TAXES ON DOMESTICALLY PRODUCED GOODS USED AS
INVESTMENT ! ;
READ (ALL,I,IND) TIRD(I)
        FROM FILE table HEADER                                  "AI20" ;

COEFFICIENT (ALL,I,IND)                                         TIRI(I)
! COMMODITY TAXES ON IMPORTS USED AS INVESTMENT ! ;
READ (ALL,I,IND) TIRI(I)
        FROM FILE table HEADER                                  "AI21" ;
```

```
COEFFICIENT (ALL,I,IND)                                    TGRD(I)
! COMMODITY TAXES ON DOMESTICALLY PRODUCED GOODS SOLD TO
GOVERNMENT ! ;
READ (ALL,I,IND) TGRD(I)
        FROM FILE table HEADER                                  "AI22" ;

COEFFICIENT (ALL,I,IND)                                    TGRI(I)
! COMMODITY TAXES ON IMPORTS SOLD TO GOVERNMENT ! ;
READ (ALL,I,IND) TGRI(I)
        FROM FILE table HEADER                                  "AI23" ;

COEFFICIENT (ALL,I,IND)                                    TER(I)
! COMMODITY TAXES ON EXPORTS ! ;
READ (ALL,I,IND) TER(I)
        FROM FILE table HEADER                                  "AI24" ;

COEFFICIENT (ALL,I,IND)                                    STSTD(I)
! COMMODITY TAXES ON DOMESTIC CHANGES IN STOCKS ! ;
READ (ALL,I,IND) STSTD(I)
        FROM FILE table HEADER                                  "AI25" ;

COEFFICIENT (ALL,I,IND)                                    STSTI(I)
! COMMODITY TAXES ON IMPORTED CHANGES IN STOCKS ! ;
READ (ALL,I,IND) STSTI(I)
        FROM FILE table HEADER                                  "AI26" ;

COEFFICIENT (ALL,I,IND)                                    DR(I)
! DUTY BY SECTOR ! ;
READ (ALL,I,IND) DR(I)
        FROM FILE table HEADER                                  "AI27" ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                         SHCOMI(I,J)
! SHARE OF COMMODITY I IN TOTAL COSTS OF INDUSTRY J ! ;
READ (ALL,I,IND)(ALL,J,IND) SHCOMI(I,J)
        FROM FILE coeff HEADER                                  "H001" ;
```

```
COEFFICIENT (ALL,I,IND)                                    SHTINVI(I)
! SHARE OF COMMODITY I IN AGGREGATE INVESMENT EXPENDITURE ! ;
READ (ALL,I,IND) SHTINVI(I)
        FROM FILE coeff HEADER                                  "H003" ;

COEFFICIENT (ALL,I,IND)                                    SHHHCONI(I)
! SHARE OF COMMODITY I IN HH CONSUMPTION EXPENDITURE ! ;
READ (ALL,I,IND) SHHHCONI(I)
        FROM FILE coeff HEADER                                  "H005" ;

COEFFICIENT (ALL,I,IND)                                    SHGOVCOMI(I)
! SHARE OF COMMODITY I IN AGGREGATE GOVERNMENT CONSUMPTION
EXPENDITURE ! ;
READ (ALL,I,IND) SHGOVCOMI(I)
        FROM FILE coeff HEADER                                  "H007" ;

COEFFICIENT (ALL,I,IND)                                    SHEXPSCOMI(I)
! SHARE OF COMMODITY I IN AGGREGATE EXPORTS ! ;
READ (ALL,I,IND) SHEXPSCOMI(I)
        FROM FILE coeff HEADER                                  "H009" ;

COEFFICIENT (ALL,J,IND)                                    SHLABTC(J)
! SHARE OF LABOUR IN TOTAL COSTS OF INDUSTRY J ! ;
READ (ALL,J,IND) SHLABTC(J)
        FROM FILE coeff HEADER                                  "H012" ;

COEFFICIENT (ALL,J,IND)                                    SHCAPTC(J)
! SHARE OF CAPITAL IN TOTAL COSTS OF INDUSTRY J ! ;
READ (ALL,J,IND) SHCAPTC(J)
        FROM FILE coeff HEADER                                  "H013" ;

COEFFICIENT (ALL,J,IND)                                    SHLANDTC(J)
! SHARE OF LAND IN TOTAL COSTS OF INDUSTRY ! ;
READ (ALL,J,IND) SHLANDTC(J)
```

```
        FROM FILE coeff HEADER                                  "H014" ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                              SHDOMINTI(I,J)
! SHARE OF DOMESTIC PRODUCTS IN INTERMEDIATE USAGE OF COMMODITY
I BY IND J ! ;
READ (ALL,I,IND)(ALL,J,IND) SHDOMINTI(I,J)
        FROM FILE coeff HEADER                                  "S001" ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                              SHIMPINTI(I,J)
! SHARE OF IMPORTS IN INTERMEDIATE USAGE OF COMMODITY I BY IND J ! ;
READ (ALL,I,IND)(ALL,J,IND) SHIMPINTI(I,J)
        FROM FILE coeff HEADER                                  "S002" ;

COEFFICIENT (ALL,I,IND)                                         SHDOMINVI(I)
! SHARE OF DOMESTIC PRODUCTS IN TOTAL INVESTMENT ! ;
READ (ALL,I,IND) SHDOMINVI(I)
        FROM FILE coeff HEADER                                  "S003" ;

COEFFICIENT (ALL,I,IND)                                         SHIMPINVI(I)
! SHARE OF IMPORTS IN TOTAL INVESTMENT ! ;
READ (ALL,I,IND) SHIMPINVI(I)
        FROM FILE coeff HEADER                                  "S004" ;

COEFFICIENT (ALL,I,IND)                                         SHDOMCONI(I)
! SHARE OF DOMESTIC PRODUCTS IN HH CONSUMPTION OF COMMODITY I ! ;
READ (ALL,I,IND) SHDOMCONI(I)
        FROM FILE coeff HEADER                                  "S005" ;

COEFFICIENT (ALL,I,IND)                                         SHIMPCONI(I)
! SHARE OF IMPORTS IN HH CONSUMPTION OF COMMODITY I ! ;
READ (ALL,I,IND) SHIMPCONI(I)
        FROM FILE coeff HEADER                                  "S006" ;

COEFFICIENT (ALL,I,IND)                                         SHDGOVI(I)
```

```
! SHARE OF DOMESTIC PRODUCTS IN GOVERNMENT CONSUMPTION
EXPENDITURE ! ;
READ (ALL,I,IND) SHDGOVI(I)
        FROM FILE coeff HEADER                                  "S007" ;

COEFFICIENT (ALL,I,IND)                                         SHIGOVSI(I)
! SHARE OF IMPORTS IN GOVERNMENT CONSUMPTION EXPENDITURE ! ;
READ (ALL,I,IND) SHIGOVSI(I)
        FROM FILE coeff HEADER                                  "S008" ;

!*****************************************************************************!
!                        CALCULATED COEFFICIENTS                              !
!*****************************************************************************!

!_____________________________________________________________________________!
!                          INTERMEDIATE USAGE                                 !
!_____________________________________________________________________________!


COEFFICIENT (ALL,J,IND)                                         COSTINP(J)
! TOTAL COSTS OF INDUSTRY J EXCLUDING INDIRECT TAXES ! ;
FORMULA (ALL,J,IND)
COSTINP(J) = SUM(I,IND,DINT(I,J)) + SUM(I,IND,IINTS(I,J)) +
        SUM(I,IND,TRD(I,J)) + SUM(I,IND,TRI(I,J)) +
        LAB(J) + CAP(J) + LAND(J) ;

COEFFICIENT (ALL,J,IND)                                         TOTCOSTS(J)
! TOTAL COSTS OF INDUSTRY J INCLUDING INDIRECT TAXES ! ;
FORMULA (ALL,J,IND)
TOTCOSTS(J) = TSR(J) + COSTINP(J) ;

ZERODIVIDE DEFAULT 1.0 ;

COEFFICIENT (ALL,J,IND)                                         SH_NV_BV(J)
```

```
! RATIO OF COSTINP TO TOTCOSTS !;
FORMULA (ALL,J,IND)
SH_NV_BV(J) = COSTINP(J) / TOTCOSTS(J) ;

ZERODIVIDE DEFAULT 0.0 ;

COEFFICIENT (ALL,J,IND)                                    SH_CT_BV(J)
! SHARE OF NON COMMODITY INDIRECT TAXES IN TOTAL COSTS  ! ;
FORMULA (ALL,J,IND)
SH_CT_BV(J) = TSR(J) / TOTCOSTS(J) ;

ZERODIVIDE DEFAULT 1.0 ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                         BPDOMCOM(I,J)
! BASIC PRICE COEFFICIENT OF THE ITH DOMESTIC COMMODITY
(INTERMEDIATE) ! ;
FORMULA (ALL,I,IND)(ALL,J,IND)
BPDOMCOM(I,J) = DINT(I,J) / (DINT(I,J) + TRD(I,J)) ;

ZERODIVIDE DEFAULT 1.0 ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                         BPIMPCOM(I,J)
! BASIC PRICE COEFFICIENT OF THE ITH IMPORTED COMMODITY
(INTERMEDIATE) ! ;
FORMULA (ALL,I,IND)(ALL,J,IND)
BPIMPCOM(I,J) = IINTS(I,J) / (IINTS(I,J) + TRI(I,J)) ;

ZERODIVIDE DEFAULT 0.0 ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                         TDINT(I,J)
! COMMODITY TAX COEFFICIENT ON DOMESTICALLY PRODUCED
INTERMEDIATE USAGE ! ;
FORMULA (ALL,I,IND)(ALL,J,IND)
TDINT(I,J) = TRD(I,J) / (DINT(I,J) + TRD(I,J)) ;
```

```
ZERODIVIDE DEFAULT 0.0 ;

COEFFICIENT (ALL,I,IND)(ALL,J,IND)                              TIINT(I,J)
! COMMODITY TAX COEFFICIENT ON IMPORTED INTERMEDIATE USAGE ! ;
FORMULA (ALL,I,IND)(ALL,J,IND)
TIINT(I,J) = TRI(I,J) / (IINTS(I,J) + TRI(I,J)) ;


!_____________________________________________________________________!
!                        HOUSEHOLD EXPENDITURE                        !
!_____________________________________________________________________!


COEFFICIENT                                                     AGG_CON
! AGGREGATE CONSUMPTION EXPENDITURE  ! ;
FORMULA
AGG_CON =SUM(I,IND,DCON(I)) + SUM(I,IND,ICONS(I)) +
      SUM(I,IND,TCRD(I)) + SUM(I,IND,TCRI(I))  ;

ZERODIVIDE DEFAULT 1.0 ;

COEFFICIENT (ALL,I,IND)                                         BPDOMCON(I)
! BASIC PRICE COEFFICIENT OF DOMESTICALLY PRODUCED
CONSUMPTION GOODS ! ;
FORMULA (ALL,I,IND)
BPDOMCON(I) = DCON(I) / (DCON(I) + TCRD(I)) ;

ZERODIVIDE DEFAULT 0.0 ;

COEFFICIENT (ALL,I,IND)                                         TDCON(I)
! COMMODITY TAX COEFFICIENT OF DOMESTICALLY PRODUCED
CONSUMPTION GOODS ! ;
FORMULA (ALL,I,IND)
TDCON(I) = TCRD(I) / (DCON(I) + TCRD(I)) ;
```

```
ZERODIVIDE DEFAULT 1.0 ;

COEFFICIENT (ALL,I,IND)                                    PICON(I)
! BASIC PRICE COEFFICIENT OF IMPORTED CONSUMPTION GOODS ! ;
FORMULA (ALL,I,IND)
PICON(I) = ICONS(I) / (ICONS(I) + TCRI(I)) ;

ZERODIVIDE DEFAULT 0.0 ;

COEFFICIENT (ALL,I,IND)                                    TICON(I)
! COMMODITY TAX COEFFICIENT OF IMPORTED CONSUMPTION GOODS ! ;
FORMULA (ALL,I,IND)
TICON(I) = TCRI(I) / (ICONS(I) + TCRI(I)) ;


!_____________________________________________________________!
!                    GOVERNMENT EXPENDITURE                   !
!_____________________________________________________________!


COEFFICIENT                                                AGG_GOVEXP
! AGGREGATE GOVERNMENT CONSUMPTION EXPENDITURE ! ;
FORMULA
AGG_GOVEXP = SUM(I,IND,DGOV(I)) + SUM(I,IND,IGOVS(I)) +
        SUM(I,IND,TGRD(I)) + SUM(I,IND,TGRI(I)) ;

ZERODIVIDE DEFAULT 1.0 ;

COEFFICIENT (ALL,I,IND)                                    ZETA1_DGOV(I)
! BASIC PRICE COEFFICIENT OF DOMESTICALLY PRODUCED
GOVERNMENT CONSUMPTION ! ;
FORMULA (ALL,I,IND)
ZETA1_DGOV(I) = DGOV(I) / (DGOV(I) + TGRD(I)) ;

ZERODIVIDE DEFAULT 0.0 ;
```

```
COEFFICIENT (ALL,I,IND)                                    ZETA2_DGOV(I)
! COMMODITY TAX COEFFICIENT OF DOMESTICALLY PRODUCED
GOVERNMENT CONSUMPTION ! ;
FORMULA (ALL,I,IND)
ZETA2_DGOV(I) = TGRD(I) / (DGOV(I) + TGRD(I)) ;

ZERODIVIDE DEFAULT 1.0 ;

COEFFICIENT (ALL,I,IND)                                    ZETA1_IGOVS(I)
! BASIC PRICE COEFFICIENT OF IMPORTED GOVERNMENT CONSUMPTION ! ;
FORMULA (ALL,I,IND)
ZETA1_IGOVS(I) = IGOVS(I) / (IGOVS(I) + TGRI(I))  ;

ZERODIVIDE DEFAULT 0.0 ;

COEFFICIENT (ALL,I,IND)                                    ZETA2_IGOVS(I)
! COMMODITY TAX COEFFICIENT OF IMPORTED GOVERNMENT
CONSUMPTION ! ;
FORMULA (ALL,I,IND)
ZETA2_IGOVS(I) = TGRI(I) / (IGOVS(I) + TGRI(I)) ;



!_____________________________________________________________!
!              INVESTMENT EXPENDITURE                         !
!_____________________________________________________________!


COEFFICIENT                                                AGG_INVEXP
! AGGREGATE INVESTMENT EXPENDITURE !  ;
FORMULA
AGG_INVEXP = SUM(I,IND,DINV(I)) + SUM(I,IND,IINVS(I)) +
        SUM(I,IND,TIRD(I)) + SUM(I,IND,TIRI(I)) ;
```

```
ZERODIVIDE DEFAULT 1.0 ;

COEFFICIENT (ALL,I,IND)                                    ZETA1_DINV(I)
! BASIC PRICE COEFFICIENT OF DOMESTIC INVESMENT ! ;
FORMULA (ALL,I,IND)
ZETA1_DINV(I) = DINV(I) / (DINV(I) + TIRD(I)) ;

ZERODIVIDE DEFAULT 0.0 ;

COEFFICIENT (ALL,I,IND)                                    ZETA2_DINV(I)
! COMMODITY TAX COEFFICIENT OF DOMESTIC INVESTMENT ! ;
FORMULA (ALL,I,IND)
ZETA2_DINV(I) = TIRD(I) / (DINV(I) + TIRD(I)) ;

ZERODIVIDE DEFAULT 1.0 ;

COEFFICIENT (ALL,I,IND)                                    ZETA1_IINVS(I)
! BASIC PRICE COEFFICIENT OF IMPORTED INVESTMENT ! ;
FORMULA (ALL,I,IND)
ZETA1_IINVS(I) = IINVS(I) / (IINVS(I) + TIRI(I))   ;

ZERODIVIDE DEFAULT 0.0 ;

COEFFICIENT (ALL,I,IND)                                    ZETA2_IINVS(I)
! COMMODITY TAX COEFFICIENT OF IMPORTED INVESTMENT ! ;
FORMULA (ALL,I,IND)
ZETA2_IINVS(I) = TIRI(I) / (IINVS(I) + TIRI(I)) ;


!_____________________________________________________________!
!                    EXPORTS                                  !
!_____________________________________________________________!
```

```
COEFFICIENT                                                    AGG_EXP
! AGGREGATE EXPORTS ! ;
FORMULA
AGG_EXP = SUM(I,IND,EXPS(I)) + SUM(I,IND,TER(I)) ;

ZERODIVIDE DEFAULT 1.0 ;

COEFFICIENT (ALL,I,IND)                                        ZETA1_EXPS(I)
! BASIC PRICE COEFFICIENT OF EXPORTS ! ;
FORMULA (ALL,I,IND)
ZETA1_EXPS(I) = EXPS(I) / (EXPS(I) + TER(I)) ;

ZERODIVIDE DEFAULT 0.0 ;

COEFFICIENT (ALL,I,IND)                                        ZETA2_EXPS(I)
! COMMODITY TAX COEFFICIENT OF EXPORTS ! ;
FORMULA (ALL,I,IND)
ZETA2_EXPS(I) = TER(I) / (EXPS(I) + TER(I)) ;



!*************************************************************************************!
!                                    VARIABLES                                        !
!*************************************************************************************!

ZERODIVIDE DEFAULT 0.0 ;

VARIABLE                                                       ONE ;

VARIABLE (ALL,I,IND)(ALL,J,IND)                                INTD(I,J)
! INTERMEDIATE USAGE BY INDUSTRY J OF THE ITH DOMESTIC COMMODITY
! ;

VARIABLE (ALL,I,IND)(ALL,J,IND)                                INTI(I,J)
```

! INTERMEDIATE USAGE BY INDUSTRY J OF THE ITH IMPORTED COMMODITY
! ;

VARIABLE (ALL,I,IND)(ALL,J,IND) TOT_INT(I,J)
! TOTAL INTERMEDIATE USAGE ! ;

VARIABLE (ALL,I,IND) CONT(I)
! CONSUMPTION OF ITH COMMODITY ! ;

VARIABLE (ALL,I,IND) COND(I)
! CONSUMPTION OF ITH DOMESTIC COMMODITY ! ;

VARIABLE (ALL,I,IND) CONI(I)
! CONSUMPTION OF ITH IMPORTED COMMODITY! ;

VARIABLE (ALL,I,IND) GOVT(I)
! GOVERNMENT CONSUMPTION OF THE ITH COMMODITY ! ;

VARIABLE (ALL,I,IND) GOVD(I)
! GOVERNMENT USAGE OF ITH DOMESTIC COMMODITY ! ;

VARIABLE (ALL,I,IND) GOVI(I)
! GOVERNMENT USAGE OF ITH COMPOSITE IMPORTED COMMODITY ! ;

VARIABLE (ALL,I,IND) INVT(I)
! TOTAL INVESTMENT ! ;

VARIABLE (ALL,I,IND) INVD(I)
! INVESTMENT SALES OF ITH DOMESTIC COMMODITY! ;

VARIABLE (ALL,I,IND) INVI(I)
! INVESTMENT SALES OF ITH COMPOSITE IMPORTED COMMODITY! ;

VARIABLE (ALL,I,IND) E(I)

! EXPORTS OF COMMODITY I ! ;

VARIABLE (ALL,I,IND) OBV(I)
! INDUSTRY OUTPUT BASIC VALUE ! ;

VARIABLE (ALL,I,IND) ONV(I)
! OUTPUT NET OF PRODUCTION TAXES ! ;

VARIABLE (ALL,J,IND) PRODTAX(J)
! PRODUCTION TAXES ON OUTPUT ! ;

VARIABLE (ALL,I,IND) BVDSTCK(I)
! BASIC VALUE OF DOMESTIC CHANGE IN STOCKS ! ;

VARIABLE (ALL,I,IND)(ALL,J,IND) BVDINT(I,J)
! BASIC VALUE OF DOMESTICALLY PRODUCED INTERMEDIATE USAGE ! ;

VARIABLE (ALL,I,IND)(ALL,J,IND) CTDINT(I,J)
! COMMODITY TAX ON DOMESTICALLY PRODUCED INTERMEDIATE USAGE ! ;

VARIABLE (ALL,I,IND)(ALL,J,IND) BVIINT(I,J)
! BASIC VALUE OF IMPORTED INTERMEDIATE USAGE ! ;

VARIABLE (ALL,I,IND)(ALL,J,IND) CTIINT(I,J)
! COMMODITY TAX ON IMPORTED INTERMEDIATE USAGE ! ;

VARIABLE (ALL,I,IND) BVDCON(I)
! BASIC VALUE OF DOMESTICALLY PRODUCED CONSUMPTION GOODS ! ;

VARIABLE (ALL,I,IND) CTDCON(I)
! COMMODITY TAX ON DOMESTICALLY PRODUCED CONSUMPTION GOODS ! ;

VARIABLE (ALL,I,IND) BVICON(I)
! BASIC VALUE ON IMPORTED CONSUMPTION GOODS ! ;

VARIABLE (ALL,I,IND) CTICON(I)
! COMMODITY TAX ON IMPORTED CONSUMPTION GOODS ! ;

VARIABLE (ALL,I,IND) BVDGOV(I)
! BASIC VALUE OF DOMESTIC GOVERNMENT CONSUMPTION ! ;

VARIABLE (ALL,I,IND) CTDGOV(I)
! COMMODITY TAX ON DOMESTICALLY PRODUCED GOVT CONSUMPTION ! ;

VARIABLE (ALL,I,IND) BVIGOVS(I)
! BASIC VALUE OF IMPORTED GOVERNMENT CONSUMPTION ! ;

VARIABLE (ALL,I,IND) CTIGOVS(I)
! COMMODITY TAX ON IMPORTED GOVERNMENT CONSUMPTION ! ;

VARIABLE (ALL,I,IND) BVDINV(I)
! BASIC VALUE OF DOMESTIC INVESTMENT ! ;

VARIABLE (ALL,I,IND) CTDINV(I)
! COMMODITY TAX ON DOMESTIC INVESTMENT ! ;

VARIABLE (ALL,I,IND) BVIINVS(I)
! BASIC VALUE OF IMPORTED INVESTMENT ! ;

VARIABLE (ALL,I,IND) CTIINVS(I)
! COMMODITY TAX ON IMPORTED INVESTMENT ! ;

VARIABLE (ALL,I,IND) BVEXPS(I)
! BASIC VALUE OF EXPORTS ! ;

VARIABLE (ALL,I,IND) CTEXPS(I)
! COMMODITY TAX ON EXPORTS ! ;

VARIABLE (ALL,J,IND) LTT(J)
! TOTAL USAGE OF LABOUR ! ;

```
VARIABLE (ALL,J,IND)                                          KTT(J)
! TOTAL USAGE OF CAPITAL ! ;

VARIABLE (ALL,J,IND)                                          MTT(J)
! TOTAL USAGE OF LAND ! ;


!*****************************************************************************!
!                                EQUATIONS                                    !
!*****************************************************************************!
!                1.          INTERMEDIATE USAGE                               !
!*****************************************************************************!

 EQUATION                                                     BVDSTOCK
# BASIC VALUE OF DOMESTIC CHANGE IN STOCKS #
 (ALL,I,IND)
BVDSTCK(I) = DSTOCK(I) * ONE ;



 EQUATION                                                     TOT_OUT_BV
# TOTAL INDUSTRY OUTPUT IN BASIC VALUE #
   (ALL,I,IND)
OBV(I) = SUM(J,IND,BVDINT(I,J)) + BVDCON(I) + BVDINV(I) +
       BVDGOV(I) + BVEXPS(I) + BVDSTCK(I) ;

 EQUATION                                                     OUTNET_TAX
# OUTPUT NET OF PRODUCTION TAXES #
 (ALL,J,IND)
ONV(J) = SH_NV_BV(J) * OBV(J) ;

 EQUATION                                                     TAX_OUT
# INDIRECT TAXES ON OUTPUT #
 (ALL,J,IND)
```

```
PRODTAX(J) = SH_CT_BV(J) * OBV(J) ;

 EQUATION                                                    TOTINTUSE
# TOTAL INTERMEDIATE USAGE #
   (ALL,I,IND)(ALL,J,IND)
TOT_INT(I,J) = (SHCOMI(I,J) * ONV(J))  ;

 EQUATION                                                    DOMINTUSE
# DOMESTIC INTERMEDIATE USAGE #
   (ALL,I,IND)(ALL,J,IND)
INTD(I,J) = SHDOMINTI(I,J) * TOT_INT(I,J) ;

EQUATION                                                     IMPINTUSE
# IMPORTED INTERMEDIATE USAGE #
   (ALL,I,IND)(ALL,J,IND)
INTI(I,J) = SHIMPINTI(I,J) * TOT_INT(I,J) ;

EQUATION                                                     BVDINTUSE
# BASIC VALUE OF INTERMEDIATE USAGE (DOMESTIC) #
   (ALL,I,IND)(ALL,J,IND)
BVDINT(I,J) = BPDOMCOM(I,J) * INTD(I,J) ;

EQUATION                                                     BVIINTUSE
# BASIC VALUE OF INTERMEDIATE USAGE (IMPORTED) #
   (ALL,I,IND)(ALL,J,IND)
BVIINT(I,J) = BPIMPCOM(I,J) * INTI(I,J) ;

EQUATION                                                     COMTAXDINT
# COMMODITY TAX ON DOMESTICALLY PRODUCED INTERMEDIATE USAGE #
   (ALL,I,IND)(ALL,J,IND)
CTDINT(I,J) = TDINT(I,J) * INTD(I,J) ;

EQUATION                                                     COMTAXIINT
# COMMODITY TAX ON IMPORTED INTERMEDIATE USAGE #
   (ALL,I,IND)(ALL,J,IND)
```

```
CTIINT(I,J) = TIINT(I,J) * INTI(I,J) ;

EQUATION                                                    TOTUSELAB
# TOTAL USAGE OF LABOUR #
  (ALL,J,IND)
LTT(J) = (SHLABTC(J) * ONV(J))  ;

EQUATION                                                    TOTUSECAP
# TOTAL USAGE OF CAPITAL #
  (ALL,J,IND)
KTT(J) = (SHCAPTC(J) * ONV(J)) ;

EQUATION                                                    TOTUSELAND
# TOTAL USAGE OF LAND #
  (ALL,J,IND)
MTT(J) = (SHLANDTC(J) * ONV(J))  ;



!*********************************************************************************!
!                    2.  HOUSEHOLD CONSUMPTION                                    !
!*********************************************************************************!

EQUATION                                                    TOT_HHCON
# TOTAL HOUSEHOLD CONSUMPTION #
  (ALL,I,IND)
CONT(I) = (AGG_CON * SHHHCONI(I)) * ONE ;

EQUATION                                                    DOM_HHCON
# HOUSEHOLD CONSUMPTION OF DOMESTICALLY PRODUCED GOODS #
  (ALL,I,IND)
COND(I) = SHDOMCONI(I) * CONT(I) ;

EQUATION                                                    IMP_HHCON
# HOUSEHOLD CONSUMPTION OF IMPORTS #
```

```
  (ALL,I,IND)
CONI(I) = SHIMPCONI(I) * CONT(I)  ;

EQUATION                                                    BVAGGDOM_CON
# BASIC VALUE OF DOMESTICALLY PRODUCED CONSUMPTION GOODS #
  (ALL,I,IND)
BVDCON(I) = BPDOMCON(I) * COND(I) ;

EQUATION                                                    CTAGGDOM_CON
# COMMODITY TAX ON DOMESTICALLY PRODUCED CONSUMPTION GOODS #
  (ALL,I,IND)
CTDCON(I) = TDCON(I) * COND(I) ;

EQUATION                                                    BVAGGIMP_CON
# BASIC VALUE OF IMPORTED CONSUMPTION GOODS #
 (ALL,I,IND)
BVICON(I) = PICON(I) * CONI(I) ;

EQUATION                                                    CTAGGIMP_CON
# COMMODITY TAX ON IMPORTED CONSUMPTION GOODS #
  (ALL,I,IND)
CTICON(I) = TICON(I) * CONI(I)  ;




!*************************************************************************************!
!                        3. GOVERNMENT EXPENDITURE                                    !
!*************************************************************************************!


EQUATION                                                    TOT_GOVCON
# TOTAL GOVERNMENT CONSUMPTION #
  (ALL,I,IND)
GOVT(I) = (AGG_GOVEXP * SHGOVCOMI(I)) * ONE ;
```

```
EQUATION                                                    DOM_GOVCON
# GOVERNMENT CONSUMPTION OF DOMESTICALLY PRODUCED GOODS #
  (ALL,I,IND)
GOVD(I) = SHDGOVI(I) * GOVT(I) ;

EQUATION                                                    IMP_GOVCON
# GOVERNMENT CONSUMPTION OF IMPORTS #
  (ALL,I,IND)
GOVI(I) = SHIGOVSI(I) * GOVT(I)  ;


EQUATION                                                    BVAGGGOV_CON
#  BASIC  VALUE  OF  DOMESTICALLY  PRODUCED  GOVERNMENT
CONSUMPTION #
  (ALL,I,IND)
BVDGOV(I) = ZETA1_DGOV(I) * GOVD(I) ;

EQUATION                                                    CTAGGGOV_CON
# COMMODITY TAX ON DOMESTICALLY PRODUCED GOVERNMENT  #
  (ALL,I,IND)
CTDGOV(I) = ZETA2_DGOV(I) * GOVD(I) ;

EQUATION                                                    BVIMPGOV_CON
# BASIC VALUE OF IMPORTED GOVERNMENT #
  (ALL,I,IND)
BVIGOVS(I) = ZETA1_IGOVS(I) * GOVI(I) ;

EQUATION                                                    CTIMPGOV_CON
# COMMODITY TAX ON IMPORTED GOVERNMENT CONSUMPTION #
  (ALL,I,IND)
CTIGOVS(I) = ZETA2_IGOVS(I) * GOVI(I) ;
```

```
!*************************************************************************************!
!                          4. INVESTMENT EXPENDITURE                                  !
!*************************************************************************************!

EQUATION                                                          TOT_INVEXP
# TOTAL INVESTMENT EXPENDITURE #
  (ALL,I,IND)
INVT(I) = (AGG_INVEXP * SHTINVI(I)) * ONE ;

EQUATION                                                          DOM_INVEXP
# INVESTMENT EXPENDITURE (DOMESTIC) #
  (ALL,I,IND)
INVD(I) = SHDOMINVI(I) * INVT(I)   ;

EQUATION                                                          IMP_INVEXP
# INVESMENT EXPENDITURE (IMPORTED) #
  (ALL,I,IND)
INVI(I) = SHIMPINVI(I) * INVT(I)  ;

EQUATION                                                          BVAGGDOM_INV
# BASIC VALUE OF DOMESTIC INVESTMENT  #
  (ALL,I,IND)
BVDINV(I) = ZETA1_DINV(I) * INVD(I) ;

EQUATION                                                          CTAGGDOM_INV
# COMMODITY TAX ON DOMESTIC INVESTMENT  #
  (ALL,I,IND)
CTDINV(I) = ZETA2_DINV(I) * INVD(I) ;

EQUATION                                                          BVIMP_INV
# BASIC VALUE OF IMPORTED GOVERNMENT #
 (ALL,I,IND)
BVIINVS(I) = ZETA1_IINVS(I) * INVI(I) ;
```

```
EQUATION                                                    CTIMP_INV
# COMMODITY TAX ON IMPORTED INVESTMENT #
  (ALL,I,IND)
CTIINVS(I) = ZETA2_IINVS(I) * INVI(I) ;



!*****************************************************************************!
!                          5. EXPORTS                                         !
!*****************************************************************************!

EQUATION                                                    TOT_EXPS
# TOTAL EXPORTS OF COMMODITY I #
  (ALL,I,IND)
E(I) = (SHEXPSCOMI(I) * AGG_EXP) * ONE  ;

EQUATION                                                    BVAGG_EXPS
# BASIC VALUE OF EXPORTS #
  (ALL,I,IND)
BVEXPS(I) = ZETA1_EXPS(I) * E(I) ;

EQUATION                                                    CTAGG_EXPS
# COMMODITY TAXES ON EXPORTS #
  (ALL,I,IND)
CTEXPS(I) = ZETA2_EXPS(I) * E(I) ;


!*****************************************************************************!
!                       UPDATE STATEMENTS                                     !
!*****************************************************************************!

UPDATE (EXPLICIT) (ALL,I,IND)(ALL,J,IND)
DINT(I,J) = BVDINT(I,J) ;
```

```
UPDATE (EXPLICIT) (ALL,I,IND)(ALL,J,IND)
IINTS(I,J) = BVIINT(I,J) ;

UPDATE (EXPLICIT) (ALL,I,IND)
DINV(I) = BVDINV(I) ;

UPDATE (EXPLICIT) (ALL,I,IND)
IINVS(I) = BVIINVS(I) ;

UPDATE (EXPLICIT) (ALL,I,IND)
DCON(I) = BVDCON(I)    ;

UPDATE (EXPLICIT) (ALL,I,IND)
ICONS(I) = BVICON(I)   ;

UPDATE (EXPLICIT) (ALL,I,IND)
DGOV(I) = BVDGOV(I) ;

UPDATE (EXPLICIT) (ALL,I,IND)
IGOVS(I) = BVIGOVS(I) ;

UPDATE (EXPLICIT) (ALL,I,IND)
DSTOCK(I) = BVDSTCK(I) ;

UPDATE (EXPLICIT) (ALL,I,IND)
ISTOCK(I) = ISTOCK(I) ;

UPDATE (EXPLICIT) (ALL,I,IND)
EXPS(I) = BVEXPS(I) ;

UPDATE (EXPLICIT) (ALL,I,IND)
TSR(I) = PRODTAX(I) ;

UPDATE (EXPLICIT) (ALL,J,IND)
LAB(J) = LTT(J) ;
```

```
UPDATE (EXPLICIT) (ALL,J,IND)
CAP(J) = KTT(J) ;

UPDATE (EXPLICIT) (ALL,J,IND)
LAND(J) = MTT(J) ;

UPDATE (EXPLICIT) (ALL,I,IND)(ALL,J,IND)
TRD(I,J) = CTDINT(I,J) ;

UPDATE (EXPLICIT) (ALL,I,IND)(ALL,J,IND)
TRI(I,J) = CTIINT(I,J) ;

UPDATE (EXPLICIT) (ALL,I,IND)
TCRD(I) = CTDCON(I) ;

UPDATE (EXPLICIT) (ALL,I,IND)
TCRI(I) = CTICON(I) ;

UPDATE (EXPLICIT) (ALL,I,IND)
TIRD(I) = CTDINV(I) ;

UPDATE (EXPLICIT) (ALL,I,IND)
TIRI(I) = CTIINVS(I) ;

UPDATE (EXPLICIT) (ALL,I,IND)
TGRD(I) = CTDGOV(I) ;

UPDATE (EXPLICIT) (ALL,I,IND)
TGRI(I) = CTIGOVS(I) ;

UPDATE (EXPLICIT) (ALL,I,IND)
TER(I) = CTEXPS(I) ;

UPDATE (EXPLICIT) (ALL,I,IND)
```

STSTD(I) = STSTD(I) ;

UPDATE (EXPLICIT) (ALL,I,IND)
STSTI(I) = STSTI(I) ;

UPDATE (EXPLICIT) (ALL,I,IND)
DR(I) = DR(I) ;

!----------

END OF TABLO INPUT

----------!

References

Brown, S., Strzelecki, A. and Watts, G. forthcoming, *Matching Input-Output Data to International Trade Data and Assembling a SALTER Database*, SALTER Working Paper, Industry Commission, Canberra.

Codsi, G. and Pearson, K.R. 1988, 'GEMPACK: General-Purpose Software for Applied General Equilibrium and Other Economic Modellers', *Computer Science in Economics and Management*, Vol. 1, pp. 189-207.

Gotch, M. 1993, *Industry Assistance Data for SALTER*, SALTER Working Paper No 14, Industry Commission, Canberra.

Hambley, J. 1993, *Early Stage Processing of International Trade and Input-Output Data for SALTER*, SALTER Working Paper No 15, Industry Commission, Canberra.

Hanslow, K. 1993, *Later Stage Processing of International Trade Data for SALTER*, SALTER Working Paper No 16, Industry Commission, Canberra.

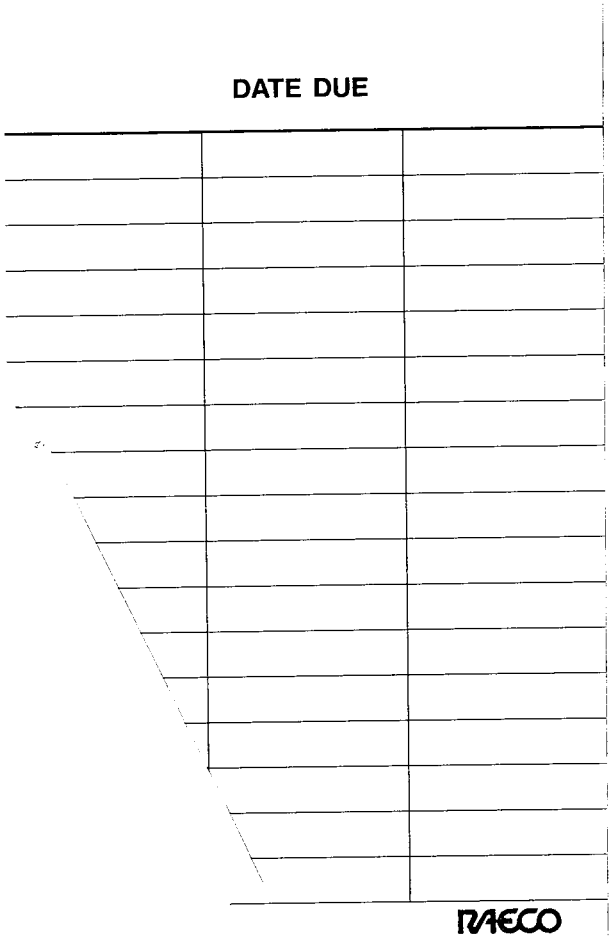
James, M. and McDougall R. 1993, *FIT: An Input-Output Update Facility for SALTER*, SALTER Working Paper No 17, Industry Commission, Canberra.

Jomini, P., Zeitsch, J.F., McDougall, R., Welsh, A., Brown, S., Hambley, J., and Kelly, J. 1991, *SALTER: A General Equilibrium Model of the World Economy, Volume 1, Model Structure, Database and Parameters*, Industry Commission, Canberra.

Mikkelsen, A. and Pearson, K.R. 1986, *How to Use Header Array Files to Access Data for Economic Models*, Impact Project GEMPACK Document No. 16, October.

Pearson, K.R. and Codsi, G. 1991, *The Update and Multi-Step Version of TABLO: User Guidelines*, Impact Project GEMPACK Document No. 30, August.

IC
339.5
SAL 18



INDUSTRIES ASSISTANCE COMMISSION
00038567