RESIDENTIAL ENERGY USE MODEL FOR AUSTRIA (REUMA)

Erwin Pönitz

December 1978 WP-78-60

2361
Laxenburg
Austria

International Institute for Applied Systems Analysis

LIST OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

%

Residential Energy Use Model for Austria (REUMA)

1 Introduction

The residential energy use model is a computer simulation model, which calculates the annual end use energy demand for the residential sector. Any simulation period up to 50 years can be chosen; The model is structured around the housing stock and its components of change, which are annual construction, demolition and retrofitting. The model is linked to a population model which provides, as a major driving function, the number of households for each simulation year. Energy use for space and water heating is calculated for seven energy types by using parameters such as floorspace, heat loss, heating hours, hot water demand and appliance efficiency. Energy demand from fourteen other appliances is calculated from the fraction of households owning each appliance and the average energy use per appliance. The housing stock is broken down into twelve home types and the parameters mentioned above reflect the characteristics of each home type.

This Working Paper is intended to provide additional information about the Residential Energy Use Model for Austria (REUMA) It is not a formal users guide, however it should allow potential users to assess the computer aspects of the model.[1] This report includes:

- A block diagram of the model, the principle flow diagrams;
- A description of the general approach used to specify fuel substitutions;
- Input-, output-, and workfiles; A list of all subroutines, their function, usage, and the other subroutines they call;

---

1 This Working Paper is intended to amplify the detailed description of REUMA provided in Poenitz, Erwin, Residential Energy Use Model for Austria (REUMA) IIASA RM-78-00, International Institut for Applied Systems Analysis.

- An alphabetical list of all parameters used in the model, and a comprehensive parameter description;

- A listing of the program, with subroutines in alphabetical order;

- An example of the data input file "rdatabase", and a sample run.

All subroutines of the model are written in FORTRAN IV. The model has been developed and is running at IIASA with a SIMulation CONtrol language, called SIMCON. It is difficult to transfer SIMCON to other computer facilities, however it can be replaced with some standard FORTRAN input/output software.

The model in its present state of development calculates residential end use energy for Austria and all its regions (Bundeslaender). Only one region can be handled at a time. No formal consistency between Austria and the Bundeslaender has been established. This problem can be solved by calculating energy demand for all regions first and then summing the results to obtain the Austrian total. However, this involves a great amount of computer time. On the other hand some degree of inconsistency (depending upon the quality and consistency of initial data input and scenario assumptions for the Bundeslaender and Austria as a whole) can be accepted taking the length of the simulation time frame into account.

2 Block Diagram and Principle Flow Diagrams of the Model

In Figure 1 a block diagram showing the relationship between the major subroutines of the model is presented. Figure 2 shows a simplified flow diagram for the subroutine umodel. The subroutine uinit contains just one call to subroutine rinit. In Figure 3 a simplified flow diagram of the subroutine rinit shows how the major subroutines of rinit interact. A flow diagram of the subroutine rmodel is shown in Figure 4.

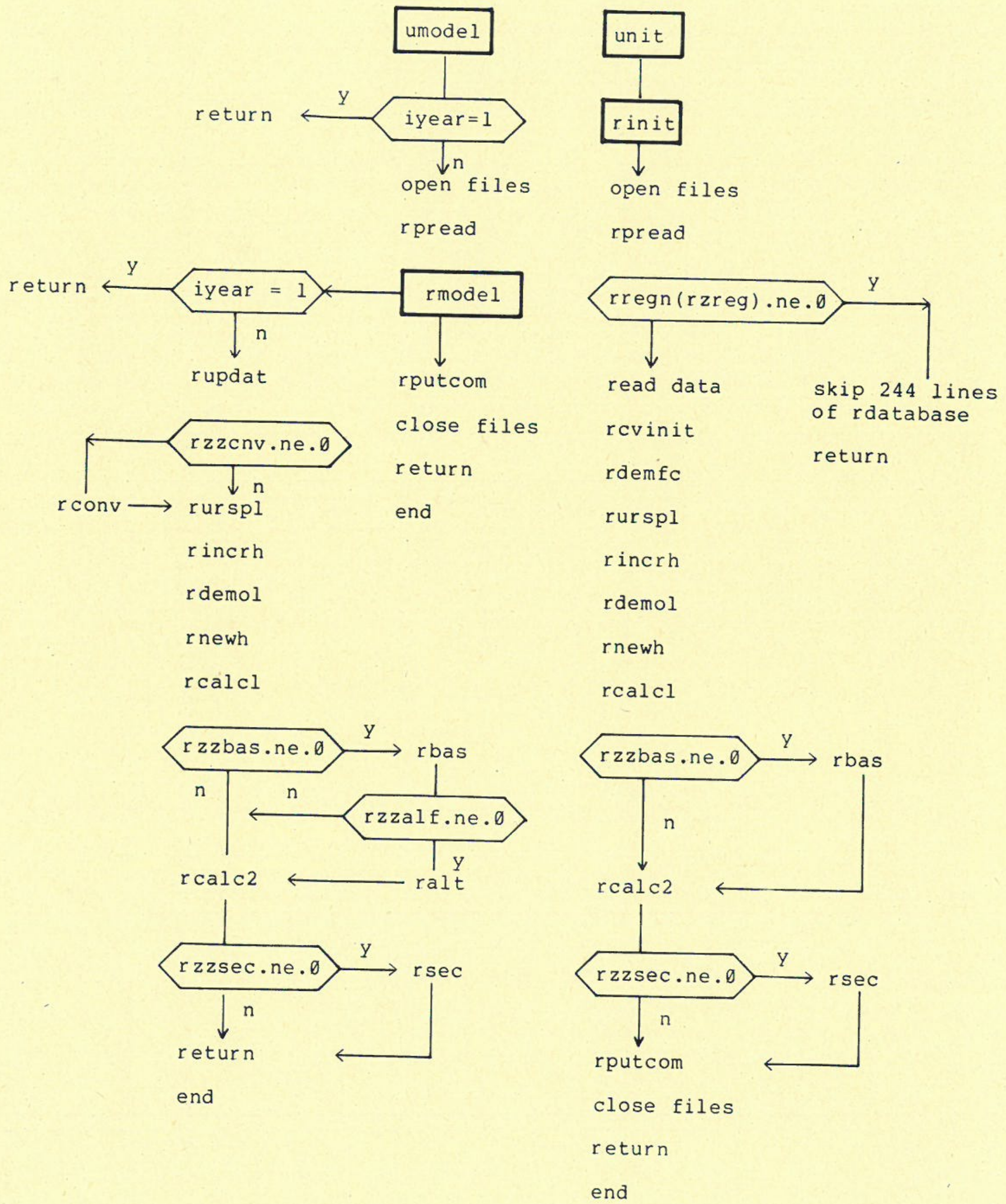

Figure 1 Block Diagram of REUMA

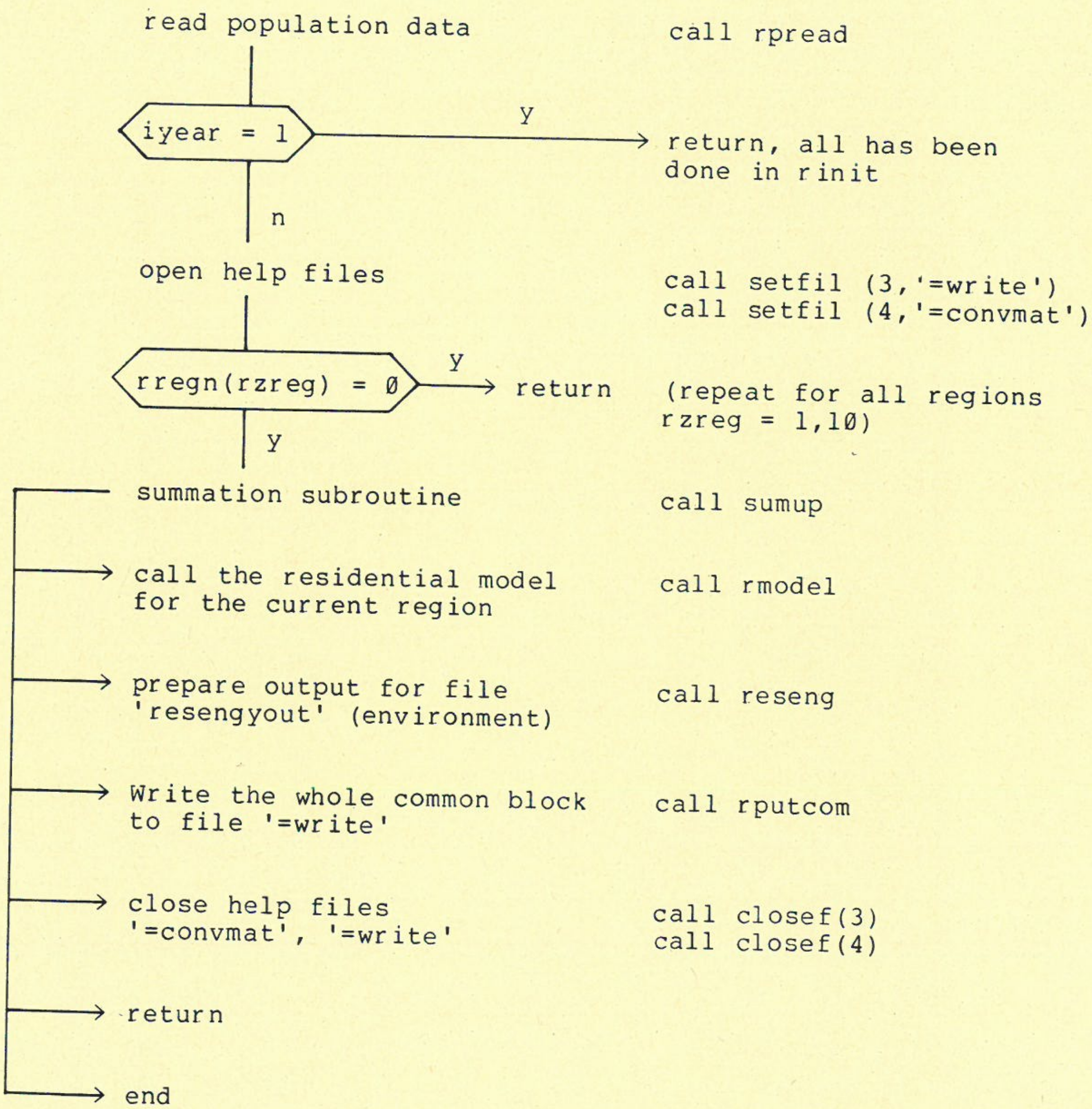

Figure 2   Flow Diagram Umodel

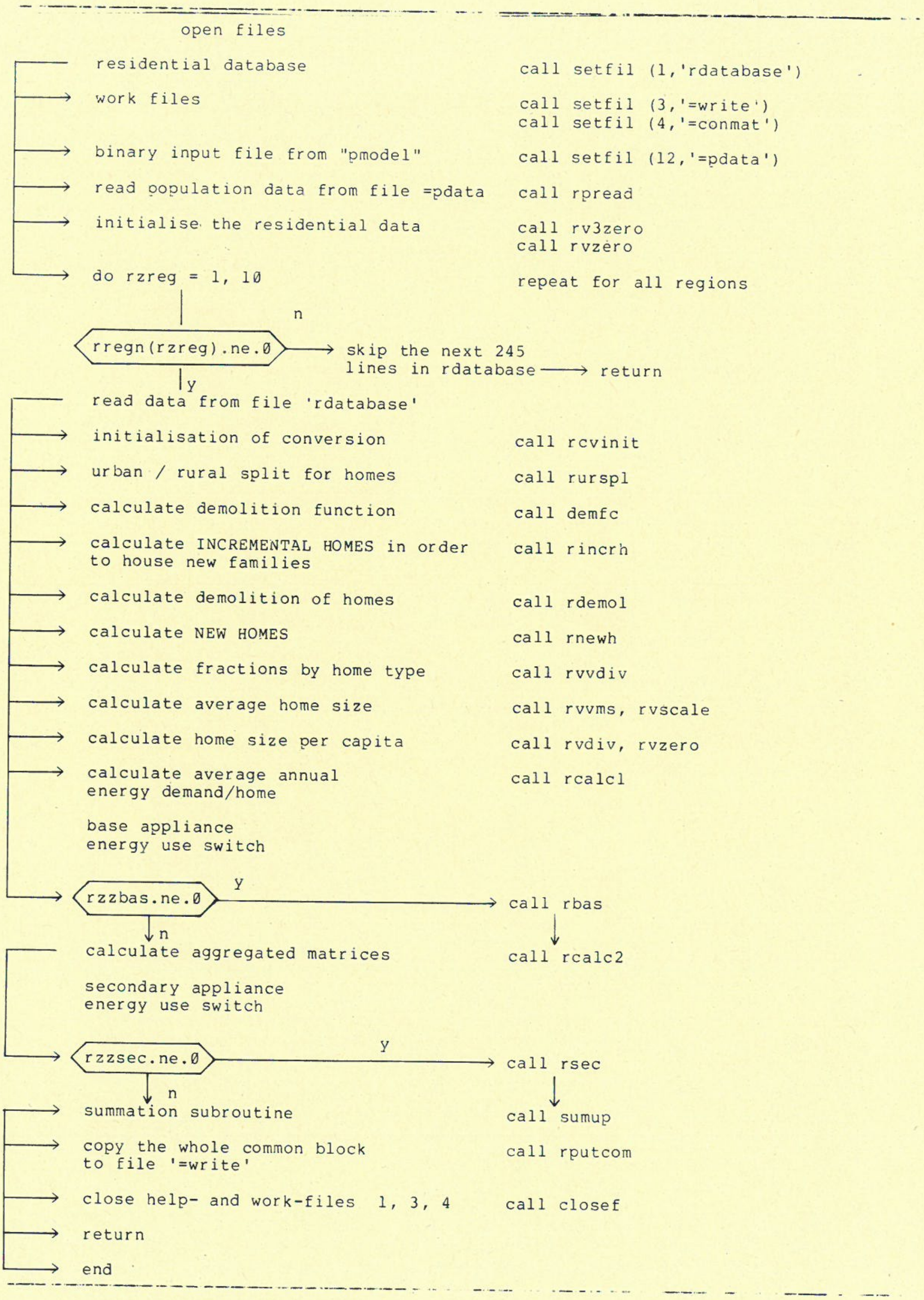

Figure 3 Flow Diagram of Subroutine Rinit

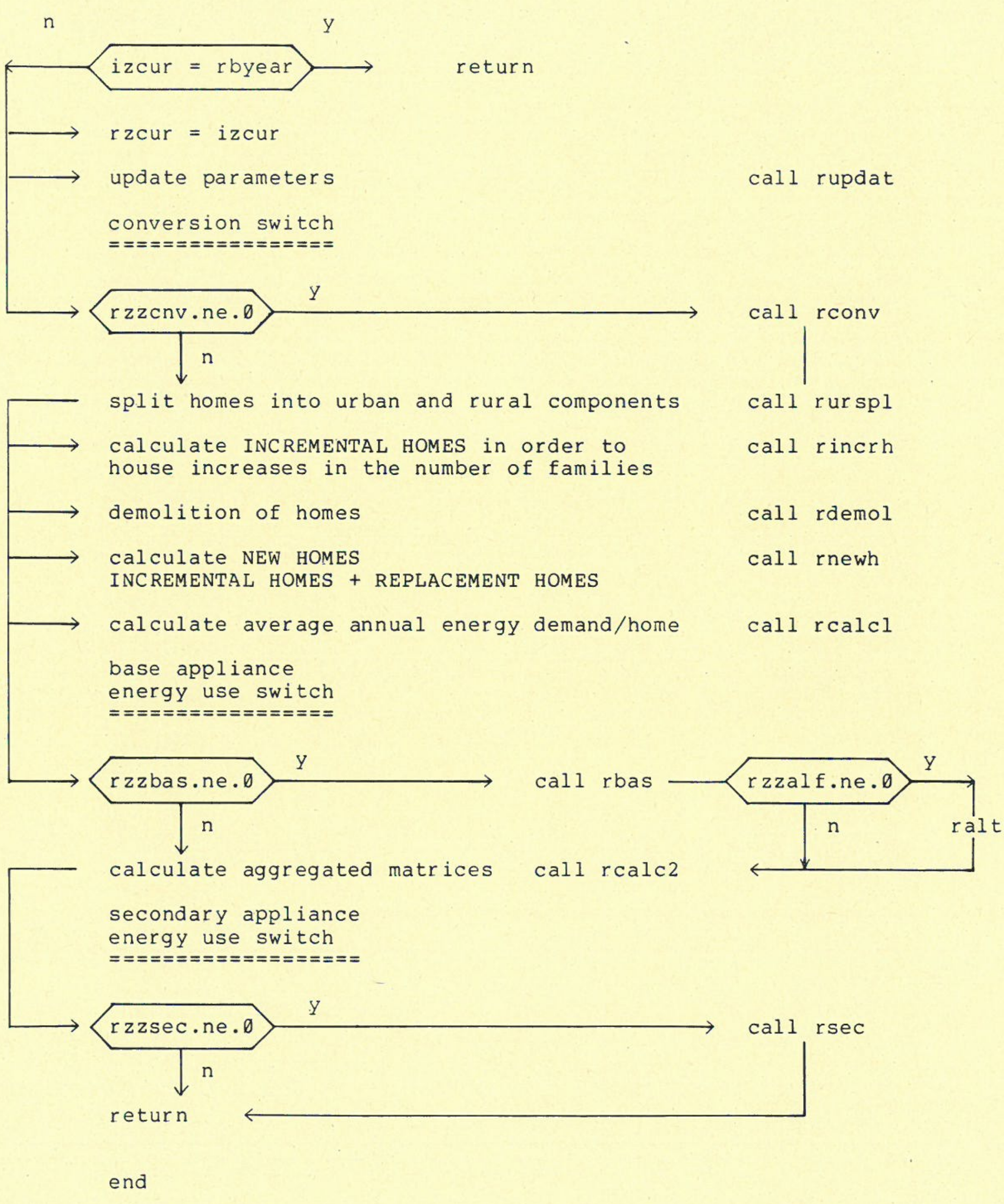

Figure 4 Flow Diagram of Subroutine Rmodel

3 Calculation of Transition Matrices for REUMA

This section describes the general approach for the calculation of the transition matrices which are used in REUMA in order to examine shifts in energy types and base appliances. The methodological approach was developed by A. Hoelzl at the International Institut for Applied Systems Analysis, Laxenburg.

As a first step the probability that a home of a given type at a given time will use a certain energy type and base appliance is determined. Four home types are considered. OLD SINGLE FAMILY HOMES and OLD APARTMENTS, which are constructed before the starting year 1971, and NEW SINGLE FAMILY HOMES and NEW APARTMENTS, which are the number of HOMES constructed in a given simulation year. For these four home types the probability distribution is specified for three points in time, thus producing twelve distinct matrices, which serve as data input for the calculation of the transition matrices.

In order to explain this approach, probability distributions for NEW SINGLE FAMILY HOMES have been used to provide consistent illustrations of computations.

The initial probability distribution for the starting year (1971) (RBXSA for OLD HOMES and RBNSA for NEW HOMES) was taken from census data, which serves also as data input for REUMA. A second matrix with the probabilities for 1975 was derived from census data and from trends of the recent past. This was done because of lack of data for the regions. When census results for more years become available, the estimation precedure should be based on time series analysis. A third probability distribution, for 2000 (approximately the last third of the scenario time frame), was constructed on the basis of scenario assumptions about energy type shifts and trends towards certain base appliances. These assumptions implicitly include future energy prices, availibility of energy types and environmental considerations. Seven energy types (electricity, gas, oil, coal, wood, district heat, and an alternative energy source) have been considered. The summation of the probabilities for space heating and for water heating for a given home type usually equals one. However, in the case of Austria, the sum of probabilities for water heating of OLD HOMES has been set to less than one. The value of one is gradually approached by 1990, which means all HOMES are fitted with a bath or shower.

These three probability distributions for a given home type are used to determine transition matrices corresponding to a Markov chain, with constant transition probabilities whose stable values are approximately equal to the hypothetical values in the year 2000. These transition matrices have been included in the data base (rdatabase) and produce in combination with the probability distributions for the starting year (RBXSA, RBNSA as included in rdatabase) for each simulation year the desired probability distribution for a given home type.

As input for the calculation of the transition matrices, only the data for one region and for two home types can be used at one time (for instance, NEW SINGLE FAMILY HOMES and NEW APARTMENTS).

3.1 Algorithmus

given:

$$d^{71},\ d^{75},\ d^{H}$$

where:

d^{71}, d^{75}, observed distributions of 1971 and 1975, and d^{H} denotes the hypothetical stationary distribution.

Step one: Initialization

a) $$\tilde{d}^{t} = \frac{t - 1971}{1975 - 1971} * (d^{75} - d^{71}) + d^{71}; \qquad t = 72, 73, 74$$

b) Least Square Solution of the following system of linear equations:

$$A\ d^{71} = \tilde{d}^{72}$$

$$A\ d^{72} = \tilde{d}^{73}$$

$$A\ d^{73} = \tilde{d}^{74}$$

$$A\ d^{74} = \tilde{d}^{75}$$

$$A\ d^{H} = \tilde{d}^{H}$$

where the coefficients of A must fulfill the following conditions:

$a_{i,j} \geq 0; \; i,j = 1,2, \ldots, n$

$\sum_{i=1}^{n} a_{i,j} = 1$

Step two: Iteration

The matrix A calculated in the first step produces a chain:

$d^{71}, \; \tilde{d}^{72} = A\, d^{71}, \; \ldots, \; \tilde{d}^{75} = A\, d^{74}$

$\text{stop}: \max_i \; | \tilde{d}_i^{75} - d_i^{75} | = e$

$$a)\; d^t = \frac{t - 1971}{1975 - 1971} * (\tilde{d}^{75} - d^{75}) + d^t \; ; \qquad t = 72, 73, 74$$

Least Square solution of the following system of linear equations:

$A\, d^{71} = \tilde{d}^{72}$

$A\, d^{72} = \tilde{d}^{73}$

$A\, d^{73} = \tilde{d}^{74}$

$A\, d^{74} = \tilde{d}^{75}$

The same conditions are valid as given for step one.

Remarks

It is probably sufficient to calculate in four year steps and to determine the intermediate distributions by linear interpolation. In this case no iteration would be required to calculate the transition matrix.

The least square solution found for the system would be:

$A\, d^{71} = d^{75}$

$A\, d^{H} = d^{H}$

together with the conditions mentioned above.

For the solution of least square problem, the matrix is regarded as a vector, consisting of the columns of the matrix A. The following system results:

$$\left.\begin{array}{lll} C\,X = d & & \text{(the elements of all colums must add up to one)} \\ E\,X = f & & \text{(transition conditions)} \\ G\,X > H & & \text{(nonnegativity conditions)} \end{array}\right\} \quad (1)$$

Since the system is underdetermined, a rank for C and E is estimated. In order to get only one solution, the solution with minimal Euclidian length is chosen, i.e.:

$$||x|| = \min \{\ ||\dot{x}||\ ;\quad \dot{x} \text{ is the solution of (1) } \}$$

3.2 Results

One resulting transition matrix is shown and explained in Table 1 below.

In Table 1 below:

$a_{i,j}$: is the probability, that someone using energy type j in time t will use energy type i in time t+1

where :

{ i : row index }
{ j : column index }

If: $d^{(t)} = \begin{Bmatrix} d_1^{(t)} \\ d_2^{(t)} \\ . \\ . \\ . \\ d_n^{(t)} \end{Bmatrix}$ is the distribution in time t

then:

$A\ d^{(t)} = d^{(t+1)} = \begin{Bmatrix} d_1^{(t+1)} \\ d_2^{(t+1)} \\ . \\ . \end{Bmatrix}$ is the distribution in time (t + 1)

Table 1   Example of transition matrix (NEW SINGLE FAMILY HOMES)

```
ier=0
estimated rank of c:   2
estimated rank of e:   2
0 components excluded eps=  1.0000e-03 iter=  1
 0.9233747 0.0134685
 0.0766253 0.9865315
ier=0
estimated rank of c:   5
estimated rank of e:  20
0 components excluded eps=  1.0000e-03 iter=  4
 0.9921875 0.0390625 0.0546875 0.0310078 0.0314961
 0.0078125 0.9609375 0.0000000 0.0077519 0.5433071
 0.0000000 0.0000000 0.8906250 0.4031008 0.0000000
 0.0000000 0.0000000 0.0546875 0.5503876 0.0000000
 0.0000000 0.0000000 0.0000000 0.0077519 0.4251969
ier=0
estimated rank of c:   6
estimated rank of e:  25
0 components excluded eps=  4.0000e-03 iter= 13
 0.9125977 0.0000000 0.0000000 0.3430786 0.0000000 0.0000000
 0.0012207 0.4589844 0.0078731 0.0384521 0.0980225 0.0256958
 0.0000000 0.0297241 0.9772963 0.1746216 0.0671997 0.0000000
 0.0213623 0.0126953 0.0121453 0.1672974 0.0000000 0.0000000
 0.0648193 0.4985962 0.0026854 0.1181641 0.7953491 0.1478882
 0.0000000 0.0000000 0.0000000 0.1583862 0.0394287 0.8264160
ier=0
estimated rank of c:   6
estimated rank of e:  25
2 components excluded eps=  4.0000e-03 iter= 13
 1.0000000 0.0000000
 0.0000000 1.0000000
ier=0
estimated rank of c:   5
estimated rank of e:  20
1 components excluded eps=  1.0000e-03 iter=  2
 0.9687500 0.0852713 0.0468750 0.0000000 0.0000000 0.0000000
 0.0078125 0.5891473 0.0546875 0.0000000 0.0000000 0.0000000
 0.0156250 0.3255814 0.8984375 0.0000000 0.0390625 0.0000000
 0.0000000 0.0000000 0.0000000 1.0000000 0.0000000 0.0000000
 0.0078125 0.0000000 0.0000000 0.0000000 0.9609375 0.0156250
 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.9843750
```

$$\left\{ \dot{d}_n(t+1) \right\}$$

A small subroutine calculates the resulting probability distributions if the transition matrix is applied to the probability distribution (RBNSA) for the starting year. In Table 2 an example of this table is shown.

The in Table 2 presented results are also shown in Figure 11 (Changes of Energy Type Mix and Base Appliance Mix by Home Type) on page 43 of the Reseach Memorandum, IIASA RM-78-ØØ.

Table 2 Example of Resulting Probability Distribution

ltype of home: sin
0
0

0year	oven						ctht						hotw					
	elec	gas	oil	coal	wood	dihe	elec	gas	oil	coal	wood	dihe	elec	gas	oil	coal	wood	dihe
1971	20.0	1.0	10.0	2.0	1.0	00.0	1.0	1.0	58.0	1.0	5.0	00.0	72.0	3.0	20.0	00.0	5.0	00.0
1972	19.5	1.6	9.2	1.6	0.4	00.0	1.3	1.5	58.7	0.9	4.9	0.4	70.9	3.4	20.3	00.0	5.4	00.0
1973	19.0	1.8	8.4	1.3	0.2	00.0	1.5	1.7	59.3	0.9	5.2	0.7	70.0	3.7	20.6	00.0	5.7	00.0
1974	18.5	1.9	7.6	1.1	0.1	00.0	1.8	1.9	59.7	0.9	5.6	0.9	69.1	3.8	21.1	00.0	6.0	00.0
1975	18.1	2.0	6.9	1.0	0.0	00.0	2.0	2.0	60.0	1.0	6.0	1.1	68.2	4.0	21.5	00.0	6.3	00.0
1976	17.7	2.0	6.3	0.9	0.0	00.0	2.2	2.1	60.2	1.0	6.4	1.4	67.4	4.0	21.9	00.0	6.6	00.0
1977	17.2	2.0	5.7	0.8	0.0	00.0	2.3	2.2	60.4	1.0	6.8	1.6	66.7	4.1	22.3	00.0	6.9	00.0
1978	16.9	2.0	5.2	0.7	0.0	00.0	2.5	2.2	60.5	1.0	7.3	1.7	66.0	4.2	22.7	00.0	7.1	00.0
1979	16.5	2.0	4.8	0.7	0.0	00.0	2.6	2.3	60.6	1.0	7.7	1.9	65.4	4.2	23.0	00.0	7.4	00.0
1980	16.2	1.9	4.4	0.6	0.0	00.0	2.8	2.4	60.6	1.0	8.1	2.0	64.8	4.2	23.4	00.0	7.6	00.0
1981	15.9	1.9	4.0	0.6	0.0	00.0	2.9	2.5	60.6	1.0	8.5	2.2	64.2	4.3	23.7	00.0	7.8	00.0
1982	15.6	1.9	3.7	0.5	0.0	00.0	3.0	2.6	60.5	1.0	8.9	2.3	63.7	4.3	24.0	00.0	8.0	00.0
1983	15.3	1.9	3.4	0.5	0.0	00.0	3.1	2.6	60.5	1.0	9.2	2.4	63.2	4.4	24.3	00.0	8.2	00.0
1984	15.0	1.9	3.1	0.4	0.0	00.0	3.2	2.7	60.4	1 0	9.6	2.6	62.7	4.4	24.5	00.0	8.4	00.0
1985	14.8	1.9	2.9	0.4	0.0	00.0	3.3	2.8	60.3	1.0	9.9	2.7	62.3	4.4	24.8	00.0	8.5	00.0
1986	14.6	1.9	2.7	0.4	0.0	00.0	3.4	2.9	60.2	1.0	10.2	2.8	61.9	4.4	25.0	00.0	8.7	00.0
1987	14.4	1.9	2.5	0.3	0.0	00.0	3.5	2.9	60.1	1.0	10.5	2.9	61.5	4.5	25.2	00.0	8.8	00.0
1988	14.2	1.9	2.3	0.3	0.0	00.0	3.5	3.0	60.0	1.0	10.8	3.0	61.1	4.5	25.4	00.0	9.0	00.0
1989	14.1	1.9	2.1	0.3	0.0	00.0	3.6	3.0	59.8	1.0	11.1	3.0	60.8	4.5	25.6	00.0	9.1	00.0
1990	13.9	2.0	2.0	0.3	0.0	00.0	3.6	3.1	59.7	1.0	11.3	3.1	60.5	4.5	25.8	00.0	9.2	00.0
1991	13.8	2.0	1.8	0.3	0.0	00.0	3.7	3.1	59.6	1.0	11.5	3.2	60.2	4.6	25.9	00.0	9.3	00.0
1992	13.7	2.0	1.7	0.2	0.0	00.0	3.7	3.2	59.5	1.0	11.8	3.3	59.9	4.6	26.1	00.0	9.4	00.0
1993	13.5	2.0	1.6	0.2	0.0	00.0	3.7	3.2	59.4	1.0	12.0	3.3	59.6	4.6	26.2	00.0	9.5	00.0
1994	13.4	2.0	1.5	0.2	0.0	00.0	3.8	3.3	59.3	1.0	12.2	3.4	59.4	4.6	26.4	00.0	9.6	00.0
1995	13.3	2.0	1.4	0.2	0.0	00.0	3.8	3.3	59.1	1.0	12.4	3.5	59.2	4.6	26.5	00.0	9.7	00.0
1996	13.3	2.0	1.3	0.2	0.0	00.0	3.8	3.3	59.0	1.0	12.5	3.5	59.0	4.6	26.6	00.0	9.8	00.0
1997	13.2	2.0	1.2	0.2	0.0	00.0	3.8	3.4	58.9	1.0	12.7	3.6	58.8	4.6	26.7	00.0	9.9	00.0
1998	13.1	2.0	1.1	0.2	0.0	00.0	3.9	3.4	58.8	1.0	12.9	3.6	58.6	4.7	26.8	00.0	9.9	00.0
1999	13.1	2.0	1.1	0.1	0.0	00.0	3.9	3.4	58.7	1.0	13.0	3.7	58.4	4.7	26.9	00.0	10.0	00.0
2000	13.0	2.0	1.0	0.1	0.0	00.0	3.9	3.5	58.6	1.0	13.1	3.7	58.2	4.7	27.0	00.0	10.1	00.0
2001	12.9	2.0	1.0	0.1	0.0	00.0	3.9	3.5	58.5	1.0	13.3	3.7	58.1	4.7	27.1	00.0	10.1	00.0
2002	12.9	2.1	0.9	0.1	0.0	00.0	3.9	3.5	58.4	1.0	13.4	3.8	57.9	4.7	27.2	00.0	10.2	00.0
2003	12.9	2.1	0.8	0.1	0.0	00.0	3.9	3.5	58.4	1.0	13.5	3.8	57.8	4.7	27.2	00.0	10.3	00.0
2004	12.8	2.1	0.8	0.1	0.0	00.0	3.9	3.5	58.3	1.0	13.6	3.8	57.7	4.7	27.3	00.0	10.3	00.0
2005	12.8	2.1	0.7	0.1	0.0	00.0	3.9	3.6	58.2	1.0	13.7	3.9	57.5	4.7	27.4	00.0	10.4	00.0
2006	12.7	2.1	0.7	0.1	0.0	00.0	3.9	3.6	58.1	1.0	13.8	3.9	57.4	4.7	27.4	00.0	10.4	00.0
2007	12.7	2.1	0.7	0.1	0.0	00.0	3.9	3.6	58.1	1.0	13.9	3.9	57.3	4.7	27.5	00.0	10.4	00.0
2008	12.7	2.1	0.6	0.1	0.0	00.0	3.9	3.6	58.0	1.0	13.9	4.0	57.2	4.7	27.5	00.0	10.5	00.0
2009	12.7	2.1	0.6	0.1	0.0	00.0	3.9	3.6	57.9	1.0	14.0	4.0	57.1	4.7	27.6	00.0	10.5	00.0
2010	12.7	2.1	0.6	0.1	0.0	00.0	3.9	3.6	57.9	1.0	14.1	4.0	57.1	4.8	27.6	00.0	10.6	00.0
2011	12.6	2.1	0.5	0.1	0.0	00.0	3.9	3.7	57.8	1.0	14.2	4.0	57.0	4.8	27.7	00.0	10.6	00.0
2012	12.6	2.2	0.5	0.1	0.0	00.0	4.0	3.7	57.8	1.0	14.2	4.0	56.9	4.8	27.7	00.0	10.6	00.0
2013	12.6	2.2	0.5	0.1	0.0	00.0	3.9	3.7	57.7	1.0	14.3	4.1	56.8	4.8	27.8	00.0	10.6	00.0
2014	12.6	2.2	0.4	0.1	0.0	00.0	3.9	3.7	57.7	1.0	14.3	4.1	56.8	4.8	27.8	00.0	10.7	00.0
2015	12.6	2.2	0.4	0.1	0.0	00.0	3.9	3.7	57.6	1.0	14.4	4.1	56.7	4.8	27.8	00.0	10.7	00.0
2016	12.6	2.2	0.4	0.1	0.0	00.0	3.9	3.7	57.6	1.0	14.4	4.1	56.6	4.8	27.9	00.0	10.7	00.0
2017	12.6	2.2	0.4	0.1	0.0	00.0	3.9	3.7	57.6	1.0	14.5	4.1	56.6	4.8	27.9	00.0	10.7	00.0
2018	12.6	2.2	0.4	0.0	0.0	00.0	3.9	3.7	57.5	1.0	14.5	4.1	56.5	4.8	27.9	00.0	10.8	00.0
2019	12.5	2.2	0.3	0.0	0.0	00.0	3.9	3.7	57.5	1.0	14.5	4.2	56.5	4.8	27.9	00.0	10.8	00.0
2020	12.5	2.2	0.3	0.0	0.0	00.0	3.9	3.7	57.5	1.0	14.6	4.2	56.4	4.8	28.0	00.0	10.8	00.0
2021	12.5	2.2	0.3	0.0	0.0	00.0	3.9	3.7	57.4	1.0	14.6	4.2	56.4	4.8	28.0	00.0	10.8	00.0

4 Files Used for the Residential Energy Use Model for Austria (REUMA)

This section describes the input-, output-, and work files of REUMA. Examples are given in the appendix C.

4.1 Input Files

rdatabase	data base for a scenario
=pdata	binary file, containing population data

4.2 Output file

All parameters included in the common block (as listed in the parameter discription) can be used to create output at each time step. One output file is always automatically prepared, i.e. resengyout: binary output file, containing information about the regions, but not about Austria as a whole. Interesting only for Environmental Models.

4.3 Work Files (binary)

=write }
} do not change these files until the end of a model run
=convmat }

4.4 Contents of the Input Data File "rdatabase"

One example of rdatabase for Austria is given in the appendix C. This file contains information for Austria and for every region (Bundeslaender) in alphabetical order:

1 Austria

2 Burgenland
3 Kaernten
4 Niederoesterreich
5 Oberoesterreich
6 Salzburg
7 Steiermark
8 Tirol

9 Vorarlberg
10 Wien

For each region input data are:

Line Number	Input Data	Data Format
1-2	Comment, Name of Region	12 (/10x,7f8.2)
3-4	rscfrc	12 (/10x,7f8.2)
5-6	rschyr	10x,7F8.2/10,7F10.6
7-8	rscsat	- " -
9-10	rpsk	10x,7F10.6/10x,7F10.6
11	rpkcal	10x,4F9.3
12	rpqcal	-"-
13	rpxhms	-"-
14	rpnhms	-"-
15	rphthy	-"-
16	rptemp	10x,4F9.5
17	rpqwat	-"-
18	rpxwat	-"-
19	rpnwat	-"-
20-26	rbxsa	6 (/10x,7F8.3)
27-39	rpxtbf	12 (/10x,7F8.3)
40-46	rbnsa	6 (/10x,7F8.3)
47-59	rpntbf	12 (/10x,7F8.3)
60	rhifrc	10x,4F8.2
61-73	rurfrc	12 (/10x,7F8.3)
74	rhzsiz,rhztyp (5)	10x,3F10.0
75	rdm	10x,2F10.5
76	rat	10x,2F10.5
77,80	ragval,ragunk	3 (/10x,8F8.0)

Transition (pre-1971) Matrices for OLD SINGLE FAMILY HOMES

81-84	comment	
85-86	rtsoch	///2F10.7/2F10.7
87-90	comment	
91-95	rtso	///4 (5F10.7/),5F10.7
96-99	comment	
100-105	rtch	///5 (6F10.7/),6F10.7
106-109	comment	
110-111	rthnhw	///2F10.7/,2F10.7

112-115	comment	
116-121	rthw	///5 (6F10.7/).6F10.7

Transition Matrices for OLD APARTMENTS (pre-1971)

122-125	comment	
126-127	rtsoch	///2F10.7/2F10.7
128-131	comment	
132-136	rtso	///4 (5F10.7/),5F10.7
137-140	comment	
141-146	rtch	///5 (6F10.7/).6F10.7
147-150	comment	
151-152	rthnhw	///2F10.7/,2F10.7
153-156	comment	
157-162	rthw	///5 (6F10.7/),6F10.7

Transition Matrices for NEW SINGLE FAMILY HOMES (= annually constructed)

163-166	comment	
167-168	rtsoch	///2F10.7/2F10.7
169-172	comment	
173-177	rtso	///4 (5F10.7/),5F10.7
178-181	comment	
182-187	rtch	///5 (6F10.7/),6F10.7
188-191	comment	
192-193	rthnhw	///2F10.7/,2F10.7
194-197	comment	
198-203	rthw	///5 (6F10.7/),6F10.7

Transition Matrices for NEW APARTMENTS (= annually constructed)

204-207	comment	
208-209	rtsoch	///2F10.7/2F10.7
210-213	comment	
214-218	rtso	///4 (5F10.7/),5F10.7
219-222	comment	
223-228	rtch	///5 (6F10.7/),6F10.7

229-232	comment	
233-234	rthnhw	///2F10.7/,2F10.7
235-238	comment	
239-244	rthw	///5 (6F10.7/),6F10.7

5 List of Subroutines Used in the Model

In this section a comprehensive list of all subroutines, their function, usage and calls to other subroutines is provided. The section number associated with the name of the subroutine indicates the importance of the subroutine, for example

- 5.2 umodel.f - these subroutines interact with directly with the SIMulation CONtrol language SIMCON and are not called by any other subroutine
- 5.2.1 rinit.f - level 1, called only by uint.f and umodel.f
- 5.2.2.1 rcvinit.f - level 2, called by rinit.f
- rciget.f, rcigtl.f - level 3, 4 or 5, subroutines without number belong to level 3, 4 or 5.

The subroutines of the model are interrelated in a complex fashion. The list of subroutines presented here, is meant to orient potential users in the system.

5.1 uinit.f

function : interface with SIMCON COMMAND LANGUAGE

usage : interface with command language simcon

subroutines used - call rinit

```
      call rinit
```

5.2 umodel.f

function: interface with SIMCON COMMAND LANGUAGE

usage : interface with simcon command language

subroutines used - call rinit, rpread, setfil 3,4, rvzero, rmodel, rsumup, reseng, rputcom, closef 3,4

```
      call rinit
      call rpread

      call setfil (3,'=write')
      call setfil (4,'=convmat')
      call rvzero (rctot,  6)
      call rvzero (rcdemr, 7)
      call rmodel (iyear + rbyear - 1)
      call rsumup
c     call reseng
      call rputcom (rzreg, 3)
      call closef (3)
      call closef (4)
```

5.2.1 rinit.f

function: initializes the values for a new run

usage : call rinit

subroutines used - call setfil 1,3,4,8,12, closef 8,12 rvzero, rpread, rv3zero, rcvinit, rurspl, rvscale, rvdiv, rdemfc, rincrh, rdemol, rnewh, rvvdiv, rvvms, rvscale, rcalc1, rbas, rcalc2, rsec, rsumup, rputcom, closef 1,3,4

```
      call setfil (1,'rdatabase')
      call setfil (3,'=write')
      call setfil (4,'=convmat')
c     call closef (8)
c        call setfil (8,'resengyout')
      call closef (12)
      call setfil (12,'=pdata')
      call closef (8)
      call setfil (8,'pdata ')
      call rpread
```

```
      call rv3zero (rbqtbf, 7, 3, 4)
      call rv3zero (rpqtbf, 7, 3, 4)

      call rvzero  (rhqtyp, 5)
      call rvzero  (rdxtyp, 5)
      call rvzero  (rdqtyp, 5)
      call rvzero  (rhityp, 5)
      call rvzero  (rhntyp, 5)

      call rcvinit
      call rurspl
      call rvscale (rhztyp, 5, 1., rhxtyp)
      call rvzero  (rhqtyp, 5)

      call rvdiv  (rhztyp, 4, rhztyp(5), rhzfrc)
      call rvdiv  (rhxtyp, 4, rhxtyp(5), rhxfrc)
      call rvzero (rhqfrc, 4)
      call rdemfc
      call rincrh
      call rdemol
      call rnewh
      call rvvdiv (rhxtyp, rhztyp, rxxfrc, 5)
      call rvvdiv (rhqtyp, rhztyp, rqqfrc, 5)
      call rvvms   (rpxhms, rhxfrc, 4, rpxhms(5))
      call rvvms   (rpqhms, rhqfrc, 4, rpqhms(5))
      call rvscale (rpxhms, 5, 1., rpzhms)
      call rvdiv   (rpxhms, 5, pfsize(rzreg), rpxlac)
      call rvdiv   (rpzhms, 5, pfsize(rzreg), rpzlac)
      call rvzero  (rpqlac, 5)
      call rvzero  (rpnlac, 5)
      call rvscale (rbntbf(1,j,i), 7, 1., rbqtbf(1,j,i))
      call rcalc1

      if (rzzbas.ne.0)  call rbas

      call rcalc2

      if (rzzsec.ne.0)  call rsec

      call rvzero (rctot,  6)
      call rvzero (rcdemr, 7)
500   call rsumup
      call rputcom (rzreg, 3)

      call closef ( 1)
      call closef ( 3)
      call closef ( 4)
      call closef ( 9)
```

5.2.1.1 rpread.f

function: reads population data from file =pdata

usage : call rpread

5.2.2 rmodel.f

function: level 1

usage : call rmodel

subroutines used - call rupdat, rdemol, rconv, rurspl, rincrh, rnewh, rcalcl, rbas, rcalc2, rsec

```
call rupdat
if(rzzcnv.ne.0)    call rconv
call rurspl
call rincrh
call rdemol
call rnewh
call rcalcl
if (rzzbas.ne.0)  call rbas
call rcalc2
if (rzzsec.ne.0)  call rsec
```

5.2.2.1 rcvinit.f

function: initialises the conversion matrices

usage : call rcvinit

subroutines used - call rciget, rciput, rcvinl, rsplit

```
call rciget (1)
call rcvinl (dxsoch, dxnso, dxnch, dxhnhw, dxnhw, rbxsa)
```

```
call rsplit (rbxsa, rbxtbf, rurfrc)
call rciput (4)
call rciget (1)
call rcvinl (dnsoch, dnnso, dnnch, dnhnhw, dnnhw, rbnsa)
call rsplit (rbnsa, rbntbf, rurfrc)
call rciput (4)
```

rciget.f

function: read conversion matrices from file 'iin'

usage : call rciget (iin)

subroutines used - call rcigtl.

```
call rcigtl (iin, rtsoch(1,1,k), 2)
call rcigtl (iin, rtso  (1,1,k), 5)
call rcigtl (iin, rtch  (1,1,k), 6)
call rcigtl (iin, rthnhw(1,1,k), 2)
call rcigtl (iin, rthw  (1,1,k), 6)
```

rcigtl.f

function: reads matrix 'x' from file 'iin'

usage : call rcigtl (iin, x, idim)

rcvinl.f

function: initialises a set of conversion matrices

usage : call rcvinl (dsoch, dnso, dnch, dhnhw, dnhw,x)

subroutines used - call rvdiv, rvnorm, rvsum.

```
call rvsum (x(1,1,k), 5, dsoch(1,k))
call rvsum (x(1,2,k), 6, dsoch(2,k))
call rvsum (x(1,3,k), 6, dhnhw(1,k))
call rvdiv (x(1,1,k), 5, dsoch(1,k), dnso (1,k))
call rvdiv (x(1,2,k), 6, dsoch(2,k), dnch (1,k))
call rvdiv (x(1,3,k), 6, dhnhw(1,k), dnhw (1,k))

call rvnorm (rtsoch(1,j,k), 2)
call rvnorm (rthnhw(1,j,k), 2)
call rvnorm (rtso  (1,j,k), 5)
```

```
call rvnorm (rtch  (l,j,k), 6)
call rvnorm (rthw  (l,j,k), 6)
```

rsplit.f

function: to split matrices into two identical ones

usage : call rsplit

subroutines used - call rvsum, rvdiv, rvscale

```
call rvsum   (x(l,3,m), 6, sum3)
call rvsum   (r(l,j,k), 6, sum(j))

call rvdiv   (r(l,l,k), 6, sum2,    r(l,l,k))
call rvdiv   (r(l,2,k), 6, sum2,    r(l,2,k))
call rvdiv   (r(l,3,k), 6, sum(3), r(l,3,k))

call rvscale (r(l,3,k), 6, sum3,    r(l,3,k))
```

rciput.f

function: writes the conversion matrices to file 'iout'

usage : call rciput (iout)

rciptl.f

function: writes matrix 'x' to file 'iout'

usage : call rciptl (iout, x, idim)

subroutines used - call rciptl

```
call rciptl (iout, rtsoch(l,l,k), 2)
call rciptl (iout, rtso  (l,l,k), 5)
call rciptl (iout, rtch  (l,l,k), 6)
call rciptl (iout, rthnhw(l,l,k), 2)
call rciptl (iout, rthw  (l,l,k), 6)
```

5.2.2.2 rupdat.f

function: updates variables

usage : call rupdat

subroutines used - call rvsum, rvdiv, rvvdiv, rvvms

```
call rvsum (rhztyp, 4, rhztyp(5))
call rvsum (rhqtyp, 4. rhqtyp(5))
call rvsum (rhxtyp, 4, rhxtyp(5))

call rvdiv (rhztyp, 4, rhztyp(5), rhzfrc)
call rvdiv (rhqtyp, 4, rhqtyp(5), rhqfrc)
call rvdiv (rhxtyp, 4, rhxtyp(5), rhxfrc)

call rvvdiv (rhxtyp, rhztyp, rxxfrc, 5)
call rvvdiv (rhqtyp, rhztyp, rqqfrc, 5)

call rv2sum (rhytyp,b,4,rhy,2)

call rvvms (rpqhms, rhqfrc, 4, rpqhms(5))
call rvvms (rpxhms, rhxfrc, 4, rpxhms(5))
call rvvms (rpzhms, rhzfrc, 4, rpzhms(5))
call rvvms (rpnhms, rhnfrc, 4, rpnhms(5))

call rvdiv (rpqhms, 5, pfsize(rzreg), rpqlac)
call rvdiv (rpxhms, 5, pfsize(rzreg), rpxlac)
call rvdiv (rpnhms, 5, pfsize(rzreg), rpnlac)
call rvdiv (rpzhms, 5, pfsize(rzreg), rpzlac)
```

5.2.2.3 rconv

function: calculates new distribution of probabilities for base appliances and energy types

usage: call rconv

subroutines used - call rcgetm, rconv2, rsplit

```
call rcgetm
call rconv2 (dxsoch, dxnso, dxnch, dxhnhw, dxnhw, rbxsa)
call rsplit (rbxsa. rbxtbf, rurfrc)
call rcgetm
call rconv2 (dnsoch, dnnso, dnnch, dnhnhw, dnnhw, rbnsa)
```

```
call rsplit (rbnsa,  rbntbf, rurfrc)
```

rcgetm

function: reads conversion matrices from file '4'; setfil (4,'=convmat')

usage : call rcgetm

subroutines used - rcget2

```
call rcget2 (4, rtsoch(1,1 m), 2)
call rcget2 (4, rtso  (1,1,m), 5)
call rcget2 (4, rtch  (1,1,m), 6)
call rcget2 (4, rthnhw(1,1,m), 2)
call rcget2 (4, rthw  (1,1,m), 6)
```

rcget2

function: reads an array from file 'iin'

usage : call rcget2 (iin, x, idim)

rconv2

function: calculates

usage : call rconv2 (dsoch, dnso, dnch, dhnhw, dnhw, x)

subroutines used - call rmvmul, rvnorm, rvscale.

```
call rmvmul (rtsoch(1 1,m), dsoch(1,m), 2)
call rmvmul (rtso  (1,1,m), dnso (1,m), 5)
call rmvmul (rtch  (1,1,m), dnch (1,m), 6)
call rmvmul (rthnhw(1,1,m), dhnhw(1,m), 2)
call rmvmul (rthw  (1,1,m), dnhw (1,m), 6)

call rvnorm  (dsoch(1,m), 2)
call rvnorm  (dnso (1,m), 5)
call rvnorm  (dnch (1,m), 6)
call rvnorm  (dhnhw(1,m), 2)
call rvnorm  (dnhw (1,m), 6)
call rvscale (dnso (1,m), 5, dsoch(1,m), x(1,1,m))
call rvscale (dnch (1,m), 6, dsoch(2,m), x(1,2,m))
call rvscale (dnhw (1,m), 6  dhnhw(1,m), x(1,3,m))
```

rsplit.f

function: to split matrices into two identical ones

usage : call rsplit

subroutines used - call rvsum, rvdiv, rvscale

```
call rvsum   (x(1,3,m), 6, sum3)
call rvsum   (r(1,j,k), 6, sum(j))

call rvdiv   (r(1,1,k), 6, sum2,   r(1,1,k))
call rvdiv   (r(1,2,k), 6, sum2,   r(1,2,k))
call rvdiv   (r(1,3,k), 6, sum(3), r(1,3,k))

call rvscale (r(1,3,k), 6, sum3,   r(1,3,k))
```

5.2.2.4 rurspl f

function: splits single family homes and apartments into an urban and rural component

usage : call rurspl

subroutines used - call rvdiv, rvscale

```
call rvdiv   (rhzsiz, 2, rzl, rhzsiz)
call rvscale (rhzsiz, 2, rhztyp(5), rhzsiz)
```

5.2.2.5 rdemfc.f

function: calculates values for the demolition function

usage : call rdemfc

subroutines used - call ragedtr, rvscale

```
call ragedtr

call rvscale (rhy,rmxage,rhztyp(5),rhy)
call rvscale (rhy,rmxage,rhzfrc(j),rhytyp(1,j))
```

ragedtr.f

function: calculates initial age distribution of homes according to census data

usage: call ragedtr

5.2.2.6 rincrh.f

function: calculates INCREMENTAL HOMES in order to house increase in the number of families

usage : call rincrh

subroutines used - call rvsum

```
call rvsum (rhityp, 4, rhityp(5))
```

5.2.2.7 rdemol.f

function: calculates number of annually demolished homes

usage : call rdemol

subroutines used - call rvzero, sum, rvdiv

```
call rvzero (rdxtyp, 5)
call rvzero (rdqtyp, 5)
call rvzero (rhdfrc, 4)

call rvsum (rdqtyp, 4, rdqtyp(5))
call rvsum (rdxtyp, 4, rdxtyp(5))
call rvsum (rhdtyp, 4, rhdtyp(5))

call rvdiv (rdqtyp, 4, rdqtyp(5), rdqfrc)
call rvdiv (rdxtyp, 4, rdxtyp(5), rdxfrc)
call rvdiv (rhdtyp, 4, rhdtyp(5), rhdfrc)
```

5.2.2.8 rnewh.f

function: calculates NEW HOMES

usage : call rnewh

subroutines used - call rvsum, rvdiv

```
        call rvsum (rhntyp, 4, rhntyp(5))
        call rvdiv (rhntyp, 4, rhntyp(5), rhnfrc)
```

5.2.2.9 rcalcl

function: calculates average energy demand per home type on the basis of heat losses, heating hours, and home size

usage: call rcalcl

5.2.2.10 rbas.f

function: calculates base appliance energy use (space and water heating)

usage: call rbas

subroutines used - call ralt, rvdiv, rvscale, rvsum, rv2sum, rvzero

```
        call rvscale (rpztbf(1,b,t), 6, rhztyp(t), rubtbf(1,b,t))
        call rv2mul  (rubtbf(1,b,t), rbztbf(1,b,t), 6, 1, rubtbf(1,b,t

        if  ((rzzalf.eq.1).and.(rzcur.gt.rbyear))  call ralt

        call rv2zero (rubbf, 3, 7)
        call rv2sum  (rubbf, 3, 7, rubf, 1)
        call rvdiv   (rubf, 7, pop(rzreg), ribf)
        call rv2sum  (rubbf, 3, 7, rubb, 2)
        call rvsum   (rubb,  3, rubtot)
```

ralt.f

function: calculates substitution processes of alternative energy type technology for conventional technologies for space and water heating

usage: call ralt

subroutines used - call rvdiv, rvscale, rvsum, rvvdiv, rvvsum, rvzero, rv2sum, rv2zero

```
call rvscale (rapl, 3, rafunc, rab)
call rvdiv   (rubtbf(l,b,t), 7, rhztyp(t), rpztbf(l,b,t))
call rvvdiv  (rpztbf(l,b,t), rbztbf(l,b,t), rpztbf(l,b,t), 7)
call rv2zero (rsbbf, 3, 7)

call rv2sum  (rsbbf, 3, 7, rsbf, 1)
call rv2sum  (rsbbf, 3, 7, rsbb, 2)

call rvsum   (rsbb, 3, rsbtot)

call rvzero  (rafbq, 4)
call rvzero  (rafbx, 4)
call rvzero  (rafbn, 4)
call rvzero  (rafb,  4)
call rvzero  (raftq, 3)
call rvzero  (raftx, 3)
call rvzero  (raftn, 3)
call rvzero  (raft,  3)

call rvsum   (rafbq, 3, rafbq(4))
call rvsum   (rafbx, 3, rafbx(4))
call rvsum   (rafbn, 3, rafbn(4))

call rvvsum  (rafbq, rafbx, rafb, 4)
call rvvsum  (rafb,  rafbn, rafb, 4)
call rvsum   (raftq, 2, raftq(3))
call rvsum   (raftx, 2, raftx(3))
call rvsum   (raftn, 2, raftn(3))

call rvvsum  (raftq, raftx, raft, 3)
call rvvsum  (raft,  raftn, raft, 3)
```

5.2.2.11 rcalc2

function: calculates all matrices related to per home consumption of energy type (f) as a function of base appliance (b) and home type (t)

usage: call rcalc2

subroutines used - call rvdiv, rvnorm, rvsum, rvzero, rv2sum, rv2zero

```
call rvzero (rpnt, 4)
call rvzero (rpxt, 4)
call rvzero (rpqt, 4)
call rvzero (rpzt, 4)

call rv2zero (rpntb, 4, 3)
call rv2zero (rpxtb, 4, 3)
call rv2zero (rpqtb, 4, 3)
call rv2zero (rpztb, 4, 3)

call rvzero (rpnf, 7)
call rvzero (rpxf, 7)
call rvzero (rpqf, 7)
call rvzero (rpzf, 7)
call rvzero (rbxf, 7)
call rvzero (rbqf, 7)
call rvzero (rbzf, 7)
call rvzero (rbnf, 7)

call rv2zero (rpntf, 4, 7)
call rv2zero (rpxtf, 4, 7)
call rv2zero (rpqtf, 4, 7)
call rv2zero (rpztf, 4, 7)

call rvzero (rpnb, 3)
call rvzero (rpxb, 3)
call rvzero (rpqb, 3)
call rvzero (rpzb, 3)

call rv2zero (rpnbf, 3, 7)
call rv2zero (rpxbf, 3, 7)
call rv2zero (rpqbf, 3, 7)
call rv2zero (rpzbf, 3, 7)

call rvdiv  (rbnf, 7, rhntyp(5), rbnf)
call rvdiv  (rbxf, 7, rhxtyp(5), rbxf)
call rvdiv  (rbqf, 7, rhqtyp(5), rbqf)
```

```
call rvdiv  (rbzf, 7, rhztyp(5), rbzf)

call rvnorm (rbnf, 7)
call rvnorm (rbxf, 7)
call rvnorm (rbqf, 7)
call rvnorm (rbzf, 7)

call rv2sum (rpntb, 4, 3, rpnt, 2)
call rv2sum (rpxtb, 4, 3, rpxt, 2)
call rv2sum (rpqtb, 4, 3, rpqt, 2)
call rv2sum (rpztb, 4  3, rpzt, 2)
call rv2sum (rpnbf, 3, 7, rpnf, 1)
call rv2sum (rpxbf, 3, 7, rpxf, 1)
call rv2sum (rpqbf, 3, 7, rpqf, 1)
call rv2sum (rpzbf, 3, 7, rpzf, 1)
call rv2sum (rpnbf, 3, 7, rpnb, 2)
call rv2sum (rpxbf, 3, 7, rpxb, 2)
call rv2sum (rpqbf, 3, 7, rpqb, 2)
call rv2sum (rpzbf, 3, 7, rpzb, 2)

call rvsum (rpnb, 3, rpn)
call rvsum (rpxb, 3, rpx)
call rvsum (rpqb, 3, rpq)
call rvsum (rpzb, 3, rpz)
```

5.2.2.12 rsec.f

function: calculates secondary appliance energy use and forms the overall totals

usage : call rsec

subroutines used - call rvzero, rvsum, rvscale, rvdiv

```
call rvzero (rusf,   7)
call rvzero (rustot, 1)

call rvsum  (rusk, 12, rscf(1))

call rvscale (rusk, 14, rhztyp(5), rusk)

call rvsum   (rscf, 2, rsc)

call rvzero  (rusf, 7)

call rvscale (rscf, 2, rhztyp(5), rusf)

call rvdiv (rusf, 2, pop(rzreg), risf)
```

```
call rvsum (rusf, 2, rustot)

call rvdiv (ruwf, 7, pop(rzreg), riwf)
call rvdiv (ruwf, 7, rhztyp(5),  rhiwf)
call rvdiv (ruwf, 7, 1., demr)
call rvdiv (ruwf, 7, 860., rkw)
```

5.2.2.13 rsumup.f

function: calculates summation values

usage : call rsumup

5.2.2.14 reseng.f

function: creates a binary file, which has been used for an environmental model

usage : call reseng

subroutines used - call rv2mul, rvscale

```
call rv2mul  (rpztf, rhztyp, 4, 7, hl, 1)
call rv2mul  (rpzt,  rhztyp, 4, 1, h2, 1)

call rvscale (rpnf,  7, rhztyp(5), h3)
```

5.2.2.15 rputcom

function: writes a copy of the common-block to file 'ifile'

usage: call rputcom (ireg, ifile)

A P P E N D I X   A

List of Parameters in Alphabetical Order

In order to find the description of a certain parameter, the parameter should be located in the alphabetical parameter list in Appendix A. The left hand column of this list gives the number of the parameter by which it can be found in the parameter description.

Parameter Description

Each parameter is described and numbered according to its appearence in the 'common block' (listed as 'model.com' in Appendix B). Integer parameters are identified. If no identification of the type is given, the parameter of in real type. If variables are required as initial input, it is noted with reference to the input data file name.

Param. Nr. List of Parameters in Alphabetical Order

2079	DEMR (F)	F=1,7		
1	INTEGR (32)			
63	PFAM (I)	I=1,10		
73	PFRUR (R)	R=1,10		
53	PFSIZE(I)	I=1,10		
83	PIFAM (I)	I=1,10		
93	PIFRUR(R)	R=1,10		
33	POP (I)	I=1,10		
43	PRURAL(I)	I=1,10		
3128	RAB (I)	I=1,3		
3123	RACEL (B)	B=1,3		
3197	RAFB (B)	B=1,4		
3193	RAFBN (B)	B=1,4		
3185	RAFBQ (B)	B=1,4		
3189	RAFBX (B)	B=1,4		
3167	RAFN (F,S,B)	F=1,3	S=1,2	B=1,3
3131	RAFQ (F,S,B)	F=1,3	S=1,2	B=1,3
3210	RAFT (S)	S=1,3		
3207	RAFTN (S)	S=1,3		
3201	RAFTQ (S)	S=1,3		
3204	RAFTX (S)	S=1,3		
3149	RAFX (F,S,B)	F=1,3	S=1,2	B=1,3
3106	RAGUNK			
3087	RAGVAL(I)	I=1,19		
3117	RAPL (B)	B=1,3		
3120	RAREPL(B)	B=1,3		
3126	RAT (I)	I=1,2		
2625	RBNF (F)	F=1 7		
2149	RBNSA (F,B,K)	F=1,7	B=1,3	K=1,2
2443	RBNTBF(F,B,T)	F=1,7	B=1,3	T=1,4
2618	RBQF (F)	F=1,7		
2359	RBQTBF(F,B,T)	F=1.7	B=1,3	T=1,4
2611	RBXF (F)	F=1,7		
2107	RBXSA (F,B,K)	F=1,7	B=1,3	K=1,2
2275	RBXTBF(F,B,T)	F=1,7	B=1,3	T=1,4
103	RBYEAR (integer)			
2632	RBZF (F)	F=1,7		
2527	RBZTBF(F,B,T)	F=1,7	B=1,3	T=1.4
2086	RCDEMR (F)	F=1,7		
553	RCOEF (J)	J=1,14		
201	RCTOT (6)			
1879	RDEM (N)	N=1,200		
199	RDM (2)			
193	RDQFRC(T)	T=1,4		
188	RDQTYP(T)	T=1,5		
184	RDXFRC(T)	T=1,4		
179	RDXTYP(T)	T=1,5		
175	RHDFRC(T)	T=1,4		
170	RHDTYP(T)	T=1,5		
157	RHIFRC(T)	T=1,4		
152	RHITYP(T)	T=1,5		

Param. Nr. List of Parameters in Alphabetical Order

866	RHIWF (F)	F=1,7		
873	RHIWTOT			
166	RHNFRC(T)	T=1,4		
161	RHNTYP(T)	T=1,5		
139	RHQFRC(T)	T=1,4		
134	RHQTYP(T)	T=1,5		
148	RHXFRC(T)	T=1,4		
143	RHXTYP(T)	T=1,5		
874	RHY (N)	N=1,2Ø1		
1Ø75	RHYTYP(N,T)	N=1,2Ø1	T=1,4	
128	RHZFRC(T)	T=1,4		
132	RHZSIZ(S)	S=1,2		
123	RHZTYP(T)	T=1,5		
827	RIBF (F)	F=1,7		
834	RIBTOT			
852	RISF (F)	F=1 2		
854	RISTOT			
855	RIWF (F)	F=1,7		
862	RIWTOT			
2Ø93	RKW (F)	F=1,7		
1Ø4	RMXAGE (integer)			
2979	RPHTHY(T)	T=1,4		
2975	RPKCAL(T)	T=1,4		
51Ø	RPN			
5ØØ	RPNB (B)	B=1,3		
43Ø	RPNBF (B,F)	B=1,3	F=1,7	
5Ø3	RPNF (F)	F=1,7		
2997	RPNHMS(T)	T=1,5		
3Ø35	RPNINS(T)	T=1,4		
3Ø17	RPNLAC(T)	T=1,5		
496	RPNT (T)	T=1,4		
418	RPNTB (T,B)	T=1,4	B=1,3	
28Ø7	RPNTBF(F,B,T)	F=1,7	B=1,3	T=1,4
39Ø	RPNTF (T,F)	T=1,4	F=1,7	
3Ø47	RPNWAT(T)	T=1,4		
3Ø75	RPNX (T,B)	T=1,4	B=1,3	
48Ø	RPQ			
47Ø	RPQB (B)	B=1,3		
3Ø8	RPQBF (B,F)	B=1,3	F=1,7	
473	RPQF (F)	F=1,7		
2987	RPQHMS(T)	T=1,5		
3Ø27	RPQINS(T)	T=1,4		
3ØØ7	RPQLAC(T)	T=1,5		
466	RPQT (T)	T=1,4		
296	RPQTB (T,B)	T=1,4	B=1,3	
2723	RPQTBF(F,B,T)	F=1,7	B=1,3	T=1,4
268	RPQTF (T,F)	T=1,4	F=1,7	
3Ø39	RPQWAT(T)	T=1,4		
3Ø63	RPQX (T,B)	T=1,4	B=1,3	
567	RPSK (J)	J=1,14		
2983	RPTEMP(T)	T=1,4		

Param. Nr.	List of Parameters in Alphabetical Order			
465	RPX			
455	RPXB (B)	B=1,3		
247	RPXBF (B,F)	B=1,3	F=1,7	
458	RPXF (F)	F=1,7		
2992	RPXHMS(T)	T=1,5		
3031	RPXINS(T)	T=1,4		
3012	RPXLAC(T)	T=1,5		
451	RPXT (T)	T=1,4		
235	RPXTB (T,B)	T=1,4	B=1,3	
2639	RPXTBF(F,B,T)	F=1,7	B=1,3	T=1,4
207	RPXTF (T,F)	T=1,4	F=1,7	
3043	RPXWAT(T)	T=1,4		
3051	RPXX (T,B)	T=1,4	B=1,3	
495	RPZ			
485	RPZB (B)	B=1,3		
369	RPZBF (B,F)	B=1,3	F=1,7	
488	RPZF (F)	F=1,7		
3002	RPZHMS(T)	T=1,5		
3022	RPZLAC(T)	T=1,5		
481	RPZT (T)	T=1,4		
357	RPZTB (T,B)	T=1,4	B=1,3	
2891	RPZTBF(F,B,T)	F=1,7	B=1,3	T=1,4
329	RPZTF (T,F)	T=1,4	F=1,7	
3107	RQQFRC(T)	T=1,5		
198	RRBHOM			
105	RREGN(D)	D=1,10	(integer)	
197	RRUHOM			
816	RSBB (B)	B=1,3		
795	RSBBF (B,F)	B=1,3	F=1,7	
819	RSBF (F)	F=1,7		
679	RSBTBF(F,B,T)	F=1,7	B=1,3	T=1,4
826	RSBTOT			
865	RSC			
863	RSCF (F)	F=1,2		
511	RSCFRC(J)	J=1,14		
525	RSCHYR(J)	J=1,14		
539	RSCSAT(J)	J=1,14		
2100	RSUM (F)	F=1,7		
784	RUBB (B)	B=1,3		
763	RUBBF (B,F)	B=1 3	F=1,7	
787	RUBF (F)	F=1,7		
595	RUBTBF(F,B,T)	F=1,7	B=1,3	T=1,4
794	RUBTOT			
2191	RURFRC(F,B,T)	F=1,7	B=1,3	T=1,4
835	RUSF (F)	F=1,7		
581	RUSK (J)	J=1,14		
842	RUSTOT			
851	RUWCUM			
843	RUWF (F)	F=1,7		
850	RUWTOT			
3112	RXXFRC(T)	T=1,5		

Param. Nr. List of Parameters in Alphabetical Order

12Ø	RZCUR (integer)
122	RZFILE (integer)
121	RZREG (integer)
118	RZZALF (integer)
115	RZZBAS (integer)
119	RZZCNV (integer)
117	RZZRNW (integer)
116	RZZSEC (integer)

PARAMETER DESCRIPTION

Parameters referring to energy use, energy consumption or energy substitution are expressed in 10^{12} calories, if not otherwise stated.

Simcon Variables

1	INTEGR (32)	Simcon integer variables

Population Variables

A 'P' at the beginning of a parameter name indicates a population variable

33	POP (I)	I=1,10 Population Values are read annually from file =pdata. I region 1 to region 10
43	PRURAL(I)	I=1,10 Rural population. Values are read annually from file =pdata.
53	PFSIZE(I)	I=1,10 Family size Values are read annually from file =pdata.
63	PFAM (I)	I=1,10 Number of families. Values are read annually from file =pdata.
73	PFRUR (R)	R=1,10 Number of rural families. Values are read annually from file =pdata. R region 1 to region 10
83	PIFAM (I)	I=1,10 Number of new families. Values are read

annually from file =pdata.

93	PIFRUR(R)		R=1,10 Number of new rural families. Values are read annually from file =pdata.

Starting Year of Model Simulations

1Ø3	RBYEAR (integer)	1971	Starting year for all calculations Initial values from blkdat (blockdata). (For Austria in this paper 1971) Homes constructed before the starting year are referred to as OLD HOMES (X), all homes constructed after the starting year are referred to as TOTAL NEW HOMES (Q). Homes built in the current simulation are called NEW HOMES (N). The number of NEW HOMES has two components: a) INCREMENTAL HOMES (I) in order to house new families and b) REPLACEMENT HOMES needed to substitute for demolished homes (D)
1Ø4	RMXAGE (integer)	13Ø	Maximum age of homes for demolition function. in 'REUMA', homes exceeding RMXAGE years are not demolished. For Austria 13Ø years in this paper. Initial values from blkdat (blockdata).

Switches and Policies

1Ø5	RREGN(D)	1,9*Ø	D=1,1Ø (integer) Region bypass switch Initial values from blkdat (blockdata). "Ø" bypass calculation for region; "1" execute calculation for region.

115 RZZBAS (integer)
1 Base appliance bypass switch Initial values from blkdat (blockdata).
"Ø" bypass base appliance routine;
"1" base appliance energy use is recalculated based on new base appliance fractions.

116 RZZSEC (integer)
1 Secondary appliance bypass switch Initial values from blkdat (blockdata).
"Ø" bypass secondary appliance routine;
"1" secondary appliance energy use is recalculated based on new secondary appliance fractions.

117 RZZRNW (integer)
1 Homes renewal policy switch . Initial values from blkdat (blockdata).
Ø" bypass demolition subroutine;
"1" demolished homes are substituted with REPLACEMENT HOMES of the same type; (i.e. a demolished single family home is replaced by single family home; this policy option tends to preseve the ratio of single family homes to apartments)
"2" the fraction of urban (rural) INCREMENTAL HOMES constructed as apartments is determined by the user over the scenario time frame. See RHNFRC(T).
RZZRNW should not be changed once simulation has started.

118 RZZALF (integer)
1 Switch for alternative energy type technology. Initial values from blkdat (blockdata).
"Ø" bypass alternative energy type technology;
"1" alternative energy type technology.

119 RZZCNV (integer)
1 Conversion (energy type substitution) routine bypass switch. Initial values from blkdat (blockdata).
Ø" bypass conversion routine;

"1" execute conversion routine.
Conversion routine: OLD and NEW HOMES change annually their energy type and base appliance mix as specified by the conversion matrices in "rdatabase".

120 RZCUR (integer)
The current simulation year RZCUR = RBYEAR + number of simulation years.

121 RZREG (integer)
Current region

122 RZFILE (integer)
Ø Switch set by umodel (not used at the present stage of development).

Housingstock

123 RHZTYP(T)
T=1,5
Number of homes:
1 urban single family homes;
2 urban apartments;
3 rural single family homes;
4 rural apartments;
5 total homes.
RHZTYP(5) Initial values from rdatabase.

128 RHZFRC(T)
T=1,4
Fraction of homes according to type.

132 RHZSIZ(S)
S=1,2
Number of homes according to size:
1 = single homes = rhztyp(1) + rhztyp(3);
2 = apartments = rhztyp(2) + rhztyp(4). Initial values from rdatabase.

134 RHQTYP(T)
T=1,5
Number of TOTAL NEW HOMES:
1-4 according to type;

		5 total.
139	RHQFRC(T)	T=1,4 Fraction of TOTAL NEW HOMES (constructed after the starting year) according to type. RHQTYP(5) = 1.
143	RHXTYP(T)	T=1,5 Number of OLD HOMES, (constructed before the starting year): 1-4 according to type; 5 total. (RHXTYP(5) is in the starting year identical with RHZTYP(5))
148	RHXFRC(T)	T=1,4 Fraction of OLD HOMES according to type. RHXTYP(5) = 1.
152	RHITYP(T)	T=1,5 Number of INCREMENTAL HOMES, which are the annually constructed homes in order to satisfy the needs of new families (PIFAM, PIFRUR) 1-4 according to type; 5 total.
157	RHIFRC(T)	T=1,4 Fraction of INCREMENTAL HOMES according to type. RHITYP(5) = 1.
161	RHNTYP(T)	T=1,5 Number of NEW HOMES (= all homes constructed in a given simulation year; INCREMENTAL HOMES plus REPLACEMENT HOMES): 1-4 according to type; 5 total.
166	RHNFRC(T)	T=1,4 Fraction of NEW HOMES according to type. RHNTYP(5) = 1.

170	RHDTYP(T)	T=1.5 Number of homes demolished in a given simulation year, which are annually substituted by REPLACEMENT HOMES: 1-4 according to type; 5 total. RHDTYP(T) = RDXTYP(T) + RDQTYP(T)
175	RHDFRC(T)	T=1,4 Fraction of demolished homes according to type. RHDTYP(5) = 1.
179	RDXTYP(T)	T=1,5 Number of demolished OLD HOMES: 1-4 according to type; 5 total.
184	RDXFRC(T)	T=1,4 Fraction of demolished OLD HOMES according to type. RDXTYP(5) = 1.
188	RDQTYP(T)	T=1,5 Number of demolished TOTAL NEW HOMES: 1-4 according to type; 5 total.
193	RDQFRC(T)	T=1,4 Fraction of demolished TOTAL NEW HOMES according to type. RHDTYP(5) = 1.
197	RRUHOM	Number of rural homes.
198	RRBHOM	Number of urban homes.
199	RDM (2)	0.02,0.0145 Constants used in the demolition function. RDEM(U)=RDM(1)*EXP(RDM(2)*(K-RMXAGE))

201 RCTOT (6)

Flows of homes summed up over time 1 = INCREMENTAL 2 = REPLACEMENT 3 = TOTAL NEW (1 + 2)
4,5, and 6 can be used to sum up some other annual flows over the simulation time frame.

Energy Use

207 RPXTF (T,F) T=1,4 F=1,7
Base appliance energy use per OLD HOME according to home type and base energy type.

235 RPXTB (T,B) T=1,4 B=1,3
Base appliance energy use per OLD HOME according to home type and base appliance.

247 RPXBF (B,F) B=1,3 F=1,7
Base appliance energy use per OLD HOME according to appliance and energy type type.

268 RPQTF (T,F) T=1,4 F=1,7
Base appliance energy use per TOTAL NEW HOME home according to home and energy type type.

296 RPQTB (T,B) T=1,4 B=1,3
Base appliance energy use per TOTAL NEW HOME according to home type and base appliance type.

308 RPQBF (B,F) B=1,3 F=1,7
Base appliance energy use per TOTAL NEW HOME according to appliance and energy type.

329	RPZTF (T,F)	T=1,4 F=1,7 Base appliance energy use per home according to home and energy type.
357	RPZTB (T,B)	T=1,4 B=1,3 Base appliance energy use per home according to home type and base appliance.
369	RPZBF (B,F)	B=1,3 F=1,7 Base appliance energy use per home according to base appliance and energy type type.
390	RPNTF (T,F)	T=1,4 F=1,7 Base appliance energy use per NEW HOME (=all annually constructed homes) according to home type and energy type.
418	RPNTB (T,B)	T=1,4 B=1,3 Base appliance energy use per NEW HOME according to home type and base appliance.
430	RPNBF (B,F)	B=1,3 F=1,7 Base appliance energy use per NEW HOME according to base appliance and energy type type.
451	RPXT (T)	T=1,4 Base appliance energy use per OLD HOME according to home type.
455	RPXB (B)	B=1,3 Base appliance energy use per OLD HOME according to base appliance type.
458	RPXF (F)	F=1,7 Base appliance energy use per OLD HOME according to energy type.

465	RPX	Energy use for all base appliances per OLD HOME.
466	RPQT (T)	T=1,4 Base appliance energy use per TOTAL NEW HOME according to home type.
470	RPQB (B)	B=1,3 Base appliance energy use per TOTAL NEW HOME according to base appliance type.
473	RPQF (F)	F=1,7 Base appliance energy use per TOTAL NEW HOME according to energy type.
480	RPQ	Energy use for all base appliances per TOTAL NEW HOME.
481	RPZT (T)	T=1,4 Base appliance energy use per home according to home type.
485	RPZB (B)	B=1,3 Base appliance energy use per home according to base appliance. This is a hypothetical value, which allows to calculate the fractions of energy used for space heating and water heating.
488	RPZF (F)	F=1,7 Base appliance energy use per home according to energy type.
495	RPZ	Energy use for all base appliances per home.
496	RPNT (T)	T=1,4 Base appliance energy use per NEW HOMES (=all annually constructed homes) ac-

		cording to home type.
500	RPNB (B)	B=1,3 Base appliance energy use per NEW HOME according to base appliance.
503	RPNF (F)	F=1,7 Base appliance energy use per NEW HOME according to energy type.
510	RPN	Energy use for all base appliances per NEW HOME.

Secondary (Household) Appliances

511	RSCFRC(J)	J=1,14 Fraction of homes owning the particular appliance. Initial values from rdatabase.
525	RSCHYR(J)	J=1,14 The "half-time" for the saturation curve for secondary appliances, i.e. the number of years for the ownership fraction to reach a value halfway between its present value and its specified saturation value. Initial values from rdatabase.
539	RSCSAT(J)	J=1,14 Saturation level at which the ownership fraction becomes approximately constant. Initial values from rdatabase.
553	RCOEF (J)	J=1,14 Calculated by the model RCOEFF(J) = 1.-EXP(-.693/RSCHYR(J))

567	RPSK (J)	J=1,14 Average yearly energy consumption of one secondary appliance of type j. Initial values from rdatabase.
581	RUSK (J)	J=1,14 Secondary appliance energy use according to appliance type.

Base Appliances

595	RUBTBF(F,B,T)	F=1,7 B=1,3 T=1,4 Total base appliance energy use according to home type, base appliance type and energy type.
679	RSBTBF(F,B,T)	F=1,7 B=1,3 T=1,4 Amount of unconventional energy source substituted for conventional energy type if alternative energy technology is applied, according to base appliance and home type.
763	RUBBF (B,F)	B=1,3 F=1,7 Total base appliance energy use according to base appliance and energy type.
784	RUBB (B)	B=1,3 Total base appliance energy according to base appliance type.
787	RUBF (F)	F=1,7 Total base appliance energy according to energy type.
794	RUBTOT	Total energy used according to all base appliances in all homes.

Substituted Energy by Unconventional Energy Technologies

795	RSBBF (B,F)	B=1,3 F=1,7 Substituted alternative energy source according to base appliance and energy type.
816	RSBB (B)	B=1,3 Substituted alternative energy source according to base appliance .
819	RSBF (F)	F=1,7 Substituted alternative energy source according to energy type.
826	RSBTOT	Total substituted alternative energy source

Values per Capita, per Home and Overall Totals

827	RIBF (F)	F=1,7 Base appliance energy per capita according to energy type used in all homes. (Energy/capita)
834	RIBTOT	Base appliance energy use per capita.
835	RUSF (F)	F=1,7 Secondary appliance energy use according to energy type.
842	RUSTOT	Total energy used for all secondary appliances in all homes.
843	RUWF (F)	F=1,7 Total consumption of each energy type.

850	RUWTOT	Total energy consumption.
851	RUWCUM	Total energy consumption for a region over the whole simulation period.
852	RISF (F)	F=1,2 Secondary appliance energy use per capita according to energy type (energy / capita).
854	RISTOT	Secondary appliance energy use per capita (energy / capita).
855	RIWF (F)	F=1,7 Total energy used per capita according to energy type.
862	RIWTOT	Total energy used per capita
863	RSCF (F)	F=1,2 Energy used by secondary appliance per home according to energy type. (only elec. and gas)
865	RSC	Energy used by all secondary appliances per home.
866	RHIWF (F)	F=1,7 Total energy use per home according to energy type.
873	RHIWTOT	Total energy used per home.

Housing Stock and Demolition Function

874	RHY (N)	N=1,2Ø1 Number of homes as a function of age over a period of RMXAGE years.
1Ø75	RHYTYP(N,T)	N=1,2Ø1 T=1,4 Number of homes N-1 years old of type T.
1879	RDEM (N)	N=1,2ØØ Probability of homes being demolished or removed as a function of age over a period of RMXAGE years.
2Ø79	DEMR (F)	F=1,7 Demand according to energy type from the residential sector.
2Ø86	RCDEMR (F)	F=1,7 Demand according to energy type from the residential sector for all calculated regions.
2Ø93	RKW (F)	F=1,7 Demand of electricity (RKW(1)), in $1Ø^6$ kwh.
21ØØ	RSUM (F)	F=1,7 rsum(j) = sum over all demr(i), i=1,j

Base Appliance Mix

21Ø7	RBXSA (F,B,K)	F=1,7 B=1,3 K=1,2 Probability of an OLD SINGLE FAMILY HOME/APARTMENT having base appliance B of energy type F. Initial values from rdatabase.

2149	RBNSA (F,B,K)	F=1,7 B=1,3 K=1,2 Probability of a NEW SINGLE FAMILY HOMES/APARTMENT having base appliance B of energy type F. Initial values from rdatabase.
2191	RURFRC(F,B,T)	F=1,7 B=1,3 T=1,4 Split fraction between urban and rural. Initial values from rdatabase.
2275	RBXTBF(F,B,T)	F=1,7 B=1,3 T=1,4 Probability of OLD HOMES of type T having base appliance B of energy type F (for Austria).
2359	RBQTBF(F,B,T) Ø.Ø	F=1,7 B=1,3 T=1,4 Probability of a TOTAL NEW HOME of type T having base appliance B and energy type F; Weighted average of RBNTBF
2443	RBNTBF(F,B,T)	F=1,7 B=1,3 T=1,4 Probability of a NEW HOME (i.e. constructed in the current simulation year) of type T having base appliance B and energy type F (for Austria). Split with RURFRC, basis RBNSA
2527	RBZTBF(F,B,T)	F=1,7 B=1,3 T=1,4 Probability of a HOME of type T having base appliance B of energy type F.
2611	RBXF (F)	F=1,7 Probability of an OLD HOME using energy type F.
2618	RBQF (F)	F=1,7 Probability of a TOTAL NEW HOME using energy type F.
2625	RBNF (F)	F=1,7 Probability of a NEW HOME using energy type F.

2632	RBZF (F)	F=1,7 Probability of a HOME using energy type F.

Energy Use Related Parameters

2639	RPXTBF(F,B,T)	F=1,7 B=1,3 T=1,4 In this matrix several energy type F, base appliance type B, and home type T related variables can be combined, such as efficiencies of heating appliances and, the fraction of the floorspace heated etc.; for an OLD HOME of type T. Initial values from rdatabase.
2723	RPQTBF(F,B,T) Ø.Ø	F=1,7 B=1,3 T=1,4 Variables related to energy use of type F for a TOTAL NEW HOME of type T, using base appliance B.
28Ø7	RPNTBF(F,B,T)	F=1,7 B=1,3 T=1,4 Variables related to energy use of type F for a NEW HOME of type T, using base appliance B. Initial values from rdatabase.
2891	RPZTBF(F,B,T)	F=1,7 B=1,3 T=1,4 Energy use of a home of type T, on base appliance B with energy type F.
2975	RPKCAL(T)	T=1,4 cal $1Ø^3/m^2/h$; Initial values from rdatabase.
2979	RPHTHY(T)	T=1,4 heating hours/year. Initial values from rdatabase.
2983	RPTEMP(T)	T=1,4 Factor to account for different climat in regions. Initial values from rdatabase.

Size of Homes, Floor Space per Capita

2987	RPQHMS(T)	T=1,5 Home-size of TOTAL NEW HOMES (m^2/home). Weighted average of RPNHMS
2992	RPXHMS(T)	T=1,5 Home-size of OLD HOMES (m^2/home). Initial values from rdatabase.
2997	RPNHMS(T)	T=1,5 Home-size of NEW HOMES (m^2/home). Initial values from rdatabase.
3002	RPZHMS(T)	T=1,5 Home-size of all homes (m^2/home).
3007	RPQLAC(T)	T=1,5 Floor space per capita for TOTAL NEW HOMES: 1-4 according to type; 5 total.
3012	RPXLAC(T)	T=1,5 Floor space per capita for OLD HOMES: 1-4 according to type; 5 total.
3017	RPNLAC(T)	T=1,5 Floor space per capita for NEW HOMES: 1-4 according to type; 5 total.
3022	RPZLAC(T)	T=1,5 Average floor space per capita for all homes: 1-4 according to type; 5 total.

Insulation - Heatlosses

3027	RPQINS(T)	T=1,4 Save factor due to improved insulation standards by construction of TOTAL NEW HOMES. If this values are not changed during the simulation period, they give the average insulation standards of NEW HOMES constructed between the starting year and a given simulation year. Initial values in subroutine 'rinit'
3031	RPXINS(T)	T=1,4 Save factor due to gradual improvement and retrofitting of OLD HOMES. Initial values in subroutine 'rinit'
3035	RPNINS(T)	T=1,4 Save factor due to improved insulation standards by construction of NEW HOMES. Initial values in subroutine 'rinit'

Water Heating

3039	RPQWAT(T)	T=1,4 water-use of TOTAL NEW HOMES. Initial values from rdatabase.
3043	RPXWAT(T)	T=1,4 water-use of OLD HOMES. Initial values from rdatabase.
3047	RPNWAT(T)	T=1,4 water-use of NEW HOMES. Initial values from rdatabase.
3051	RPXX (T,B)	T=1,4 B=1,3 internal factor calculated from the factors listed above.

3Ø63 RPQX (T,B) T=1,4 B=1,3
internal factor calculated from the factors listed above.

3Ø75 RPNX (T,B) T=1,4 B=1,3
internal factor calculated from the factors listed above.

Age of Homes in the Starting Year

3Ø87 RAGVAL(I) I=1,19
Age-distribution of homes. Up to 19 agegroups can be distinguished plus one group of homes of unknown age. Initial values from rdatabase.

31Ø6 RAGUNK
Homes of unknown age. Initial values from rdatabase.

31Ø7 RQQFRC(T) T=1,5
Fraction of TOTAL NEW HOMES:
1-4 according to type;
5 total.

3112 RXXFRC(T) T=1,5
Fraction of OLD HOMES:
1-4 according to type;
5 total.

Alternative Energy Technologies

3117 RAPL (B) B=1,3
Ø,1,1 Penetration limit for solar or heat-pumps. Initial values from blkdat (blockdata).
1-3 for single ovens, central heating, hot water

312Ø	RAREPL(B)	Ø,.5,.7	B=1,3 Fraction of energy-use, which gets replaced. Initial values from <u>blkdat</u> (<u>blockdata</u>).
3123	RACEL (B)	Ø,Ø,Ø	B=1,3 Electricity consumption of the alternative system in percent of the replaced energy (e.g. a heat pump provides 1ØØØ kwh, however needs a third of this provided energy as initial input in the form of electricity) Initial values from <u>blkdat</u> (<u>blockdata</u>).
3126	RAT (I)		I=1,2 Constants used for s-shaped functions. Initial values from <u>rdatabase</u>.
3128	RAB (I)		I=1,3 Probability that a base appliance is combined with an alternative energy appliance
3131	RAFQ (F,S,B)		F=1,3 S=1,2 B=1,3 Number of TOTAL NEW HOMES using alternative energy source, according to the energy type of the back up system, the base appliance, and the type of home. type of home 1 TOTAL NEW urban SINGLE FAMILY HOMES 2 TOTAL NEW rural SINGLE FAMILY HOMES
3149	RAFX (F,S,B)		F=1,3 S=1,2 B=1,3 Number of OLD HOMES using alternative energy source, according to the energy type of the back up system, the base appliance, and the type of home. S = type of home 1 OLD urban SINGLE FAMILY HOMES 2 OLD rural SINGLE FAMILY HOMES
3167	RAFN (F,S,B)		F=1,3 S=1,2 B=1,3 Number of NEW HOMES using alternative energy source, according to the energy type of the back up system, the base ap-

pliance, and the type of home.
type of home
1 NEW urban SINGLE FAMILY HOMES
2 NEW rural SINGLE FAMILY HOMES

3185 RAFBQ (B) B=1,4
Number of TOTAL NEW HOMES using alternative energy technology according to base appliance
1 single ovens
2 central heating
3 hot water
4 total

3189 RAFBX (B) B=1,4
Number of OLD HOMES using alternative energy technology according to base appliance
1 single ovens
2 central heating
3 hot water
4 total

3193 RAFBN (B) B=1,4
Number of NEW HOMES using alternative energy technologie according to base appliance
1 single ovens
2 central heating
3 hot water
4 total

3197 RAFB (B) B=1,4
Homes using alternative energy type technology according to base appliance.

3201 RAFTQ (S) S=1,3
Number of TOTAL NEW HOMES using alternative energy technologies, according to home type
1 TOTAL NEW urban SINGLE FAMILY HOMES
2 TOTAL NEW rural - " -
3 total (1 + 2)

3204 RAFTX (S) S=1,3

		Number of OLD HOMES (=constructed before the starting year) using alternative energy technology, according to home type 1 OLD urban SINGLE FAMILY HOMES 2 OLD rural - " - 3 total (1 + 2)
3207	RAFTN (S)	S=1,3 Number of NEW HOMES using alternative energy technology, according to home type 1 NEW urban SINGLE FAMILY HOMES 2 NEW rural - " - 3 total (1 + 2)
3210	RAFT (S)	S=1,3 Number of homes using alternative energy technology according to home type 1 urban single family homes; 2 rural single family homes; 3 total

3212 elements

12808 bytes

Variables not in rcmn

' 1	RTSOCH(2,2,2)	conversion table single oven - central heating.
' 9	RTSO (5,5,2)	conversion table single oven.
' 59	RTCH (6,6,2)	conversion table central heating.
' 131	RTHNHW(2,2,2)	conversion table hot water - non hot water
' 139	RTHW (6,6,2)	conversion table hot water
	210	Elements
	840	bytes

A P P E N D I X   B

Model Listing

The subroutines generally appear in alphabetical order. As an exception to this rule the common blocks used (model.com, conv.cmn, conv2.cmn, and equiv.cmn) are listed in the beginning.

```
c--------------------------------------------------------
c       model.com
c--------------------------------------------------------
c
c--------  pcmn
c
c
c--------  rcmn
c
      integer rbyear,  rmxage,  rregn
      integer rzzbas,  rzzsec,  rzzrnw,  rzzalf,  rzzcnv
      integer rzcur,   rzreg,   rzfile
c
c--------  simcon
c
      common integr(32)
c
c--------  pcmn
c
      common pop   (1Ø), prural(1Ø)
      common pfsize(1Ø)
      common pfam  (1Ø), pfrur (1Ø)
      common pifam (1Ø), pifrur(1Ø)
c
c--------  rcmn
c
      common  rbyear,      rmxage,      rregn (1Ø)
      common  rzzbas,      rzzsec,      rzzrnw,      rzzalf,      rzzcnv
      common  rzcur,       rzreg,       rzfile
      common  rhztyp(5),   rhzfrc(4),   rhzsiz(2)
      common  rhqtyp(5),   rhqfrc(4),   rhxtyp(5),   rhxfrc(4)
      common  rhityp(5),   rhifrc(4),   rhntyp(5),   rhnfrc(4)
      common  rhdtyp(5),   rhdfrc(4)
      common  rdxtyp(5),   rdxfrc(4),   rdqtyp(5),   rdqfrc(4)
      common  rruhom,      rrbhom
      common  rdm   (2),   rctot(6)
      common  rpxtf (4,7),  rpxtb (4,3),  rpxbf (3,7)
      common  rpqtf (4,7),  rpqtb (4,3),  rpqbf (3,7)
      common  rpztf (4,7),  rpztb (4,3),  rpzbf (3,7)
      common  rpntf (4,7),  rpntb (4,3),  rpnbf (3,7)
      common  rpxt  (4),  rpxb  (3),  rpxf  (7),  rpx
      common  rpqt  (4),  rpqb  (3),  rpqf  (7),  rpq
      common  rpzt  (4),  rpzb  (3),  rpzf  (7),  rpz
      common  rpnt  (4),  rpnb  (3),  rpnf  (7),  rpn
      common  rscfrc(14), rschyr(14), rscsat(14), rcoef(14)
      common  rpsk  (14), rusk  (14)
      common  rubtbf(7,3,4),            rsbtbf(7,3,4)
      common  rubbf(3,7), rubb  (3),  rubf  (7),     rubtot
      common  rsbbf(3,7), rsbb  (3),  rsbf  (7),     rsbtot
      common  ribf  (7),  ribtot
      common  rusf  (7),  rustot,     ruwf  (7),     ruwtot,      ruwcum
      common  risf  (2),  ristot,     riwf  (7),     riwtot
      common  rscf  (2),  rsc,        rhiwf (7),     rhiwtot
      common  rhy (2Ø1),  rhytyp(2Ø1,4),  rdem(2ØØ)
      common  demr  (7),  rcdemr(7),  rkw   (7),     rsum   (7)
```

```
      common  rbxsa (7,3,2), rbnsa (7,3,2), rurfrc(7,3,4)
      common  rbxtbf(7,3,4), rbqtbf(7,3,4), rbntbf(7,3,4), rbztbf(7,3,4)
      common  rbxf  (7),     rbqf  (7),     rbnf  (7),     rbzf  (7)
      common  rpxtbf(7,3,4), rpqtbf(7,3,4), rpntbf(7,3,4), rpztbf(7,3,4)
      common  rpkcal(4),     rphthy(4),     rptemp(4)
      common  rpqhms(5),     rpxhms(5),     rpnhms(5),     rpzhms(5)
      common  rpqlac(5),     rpxlac(5),     rpnlac(5),     rpzlac(5)
      common  rpqins(4),     rpxins(4),     rpnins(4)
      common  rpqwat(4),     rpxwat(4),     rpnwat(4)
      common  rpxx  (4,3),   rpqx  (4,3),   rpnx(4,3)
      common  ragval(19),    ragunk
      common  rqqfrc(5),     rxxfrc(5)
      common  rapl  (3),     rarepl(3),     racel (3)
      common  rat   (2),     rab  (3)
      common  rafq  (3,2,3), rafx (3,2,3),  rafn (3,2,3)
      common  rafbq (4),     rafbx (4),     rafbn (4),     rafb (4)
      common  raftq (3),     raftx (3),     raftn (3),     raft (3)
c
c----------------
%
```

```
c-----------------------------------------------------------
c     conv.cmn
c-----------------------------------------------------------
        common /conv/ dxsoch(2,2), dxnso(5,2), dxnch(6,2),
     1                dxhnhw(2,2), dxnhw(6 2),
     2                dnsoch(2,2), dnnso(5,2), dnnch(6,2),
     3                dnhnhw(2,2), dnnhw(6,2)
```

```
c-----------------------------------------------------------------
c     conv2.cmn
c-----------------------------------------------------------------
        common /conv2/  rtsoch(2,2,2), rtso(5,5,2),   rtch(6,6,2),
     1                  rthnhw(2,2,2), rthw(6,6,2)
```

```
c ------------------------------------
c       equiv.cmn
c-------------------------------------
        data ilen /65/
        dimension idumy(100,65)
        equivalence (idumy(1,1),pop(1))
c
```

```
#
c------ b l k d a t ------------------------------------------------
c
c  function                - initialization of variables used in REUMA
c                            (Residential Energy Use Model for Austria)
c
c
c-------------------------------------------------------------------
c  author                  - anton toifelhardt
c  latest revision         - 77/10/27
c
c
      block data
c
c
#include "model.com"
c
      data rzzrnw /1/,       rzzbas /1/,       rzzsec /1/
      data rzzcnv /1/,       rzzalf /0/,       rzfile /1/
      data rregn  /1,0,0,0,0,0,0,0,0,0/
c
      data rbyear /1971/,    rzcur /1971/,     rmxage /130/
c
      data rbqtbf /84*0./,   rpqtbf /84*0./
      data rhqtyp /5*0.0/,   rdxtyp /5*0.0/,   rdqtyp /5*0.0/
      data rhityp /5*0.0/,   rhntyp /5*0.0/
      data ruwcum /0.0/
c
      data rapl   /0.,1.,1./
      data rarepl /0.,.5,.7/
      data racel  /0.,.1,.1/
c
      end
```

```
#
c------ r a g e d t r ---------------------------------------------
c
c   function              - calculates age-distribution of homes;
c                           the age groups are organised
c                           according to Austrian census data of 1971
c
c   usage                 - call ragedtr
c
c   parameters            - none
c
c   subroutine used       - none
c
c....................
c
c   note                  - the result on field 'rhy' is normalized
c
c-----------------------------------------------------------------
c   author                - anton toifelhardt
c   latest revision       - 78/08/19, erwin poenitz
c
c
        subroutine ragedtr
c
c
#include "model.com"
c
        equivalence (n,rmxage)
c.....
        if (n.lt.100) goto 1000
c.....
        i = n - 91
        if (i.le.0) goto 200
c
c               compute values before 1880
c
        h = ragval(1)/float(i)
        do  100  j = 0,i-1
        rhy(n-j) = h
100     continue
c
c               values from 1880 until 1918
c
200     continue
        h = ragval(2) / 39.
        do  220  j = i,i+38
        rhy(n-j) = h
220     continue
        i = i + 39
c
c               add unknown value to the years until 1918
c
        h = ragunk/float(i)
        do  230  j = 0,i-1
        rhy(n-j) = rhy(n-j) + h
230     continue
```

```
c
c                  values from 1919 until 1944
c
         h = ragval (3) / 26.
         do  240  j = i,i+25
         rhy(n-j) = h
240      continue
         i = i + 26
c
c                  values from 1945 until 1960
c
         h = ragval (4) / 16.
         do  260  j = i,i+15
         rhy(n-j) = h
260      continue
         i = i + 16
c
c                  values from 1961 until 1971
c
         h = ragval (5) / 10.
         do  280  j = i,i+9
         rhy(n-j) = h
280      continue
         i = i + 10
c
c                  normalisation
c
         sum = 0.
         do  400  j = 1,n
         sum = sum + rhy(j)
400      continue
c
         do  440  j = 1,n
         rhy(j) = rhy(j) / sum
440      continue
c
1000     continue
         return
         end
```

```
#
c------ r a l t ------------------------------------------------------
c
c  function                - calculates substitution of alternative energy
c                            type technologies for conventional technologies
c                            for space and water heating
c
c  usage                   - call ralt
c
c  parameters              - none
c
c  subroutines used        - call rvdiv, rvscale, rvsum, rvvdiv,
c                            call rvvsum, rvzero, rv2sum, rv2zero
c
c  ..................
c
c  note:                     this subroutine is only executed if policy
c                            option switch rzzalf is set equal one
c
c
c--------------------------------------------------------------------
c  author                  - anton toifelhardt
c  latest revision         - 77/11/11
c
c
        subroutine ralt
c
c
#include "model.com"
c
        integer t,b,f,s
c
c
        ralpha = 4.394449155 / rat(2)
        rzl = exp (ralpha * (rzcur - rbyear - rat(1) - 10))
        rafunc = rzl / (1. + rzl)
c
c               probability that a home is using alternative
c               energy type technologies
c
        call rvscale (rapl, 3, rafunc, rab)
c
c               calculate energy saved using
c               alternative energy type technologies
c
        do  100  t = 1,4,2
        do  100  b = 1,3
        do  100  f = 1,3
        rsbtbf(f,b,t) = rab(b) *  rarepl(b) * rubtbf(f,b,t)
100     continue
c
c               subtract saved energy and sum up energy replaced
c               by alternative energy type technologies
c
        do  200  t = 1,4,2
        do  200  b = 1,3
        rubtbf(7,b,t) = 0.
        do  200  f = 1,3
```

```
      rubtbf(f,b,t) = rubtbf(f,b,t) - rsbtbf(f,b,t)
      rubtbf(7,b,t) = rubtbf(7,b,t) + rsbtbf(f,b,t)
200   continue
c
c             additional electricity used for alternative
c             energy type technologies, if any
c
      do  300  t = 1,4,2
      do  300  b = 1,3
      rzl = rubtbf(7,b,t) * racel(b)
      rsbtbf(l,b,t) = rsbtbf(l,b,t) - rzl
      rubtbf(l,b,t) = rubtbf(l,b,t) + rzl
300   continue
c
c             probabilities for alternative energy type technologies
c
      do  400  t = 1,4,2
      do  400  b = 1,3
      rbqtbf(7,b,t)   = 0.
      rbxtbf(7,b,t)   = 0.
      rbntbf(7,b,t)   = 0.
      rbztbf(7,b,t)   = 0.
c
      rbqtbf(7,b,t+1) = 0.
      rbxtbf(7,b,t+1) = 0.
      rbntbf(7,b,t+1) = 0.
      rbztbf(7,b,t+1) = 0.
c
      do  400  f = 1,3
      rbqtbf(7,b,t) = rbqtbf(7,b,t) + rab(b) * rbqtbf(f,b,t)
      rbxtbf(7,b,t) = rbxtbf(7,b,t) + rab(b) * rbxtbf(f,b,t)
      rbntbf(7,b,t) = rbntbf(7,b,t) + rab(b) * rbntbf(f,b,t)
      rbztbf(7,b,t) = rbztbf(7,b,t) + rab(b) * rbztbf(f,b,t)
400   continue
c
c             recalculate energy used per home according to
c             home type, base appliance and energy type.
c
      do  500  t = 1,4,2
      do  500  b = 1,3
      call rvdiv  (rubtbf(l,b,t), 7, rhztyp(t), rpztbf(l,b,t))
      call rvvdiv (rpztbf(l,b,t), rbztbf(l,b,t), rpztbf(l,b,t), 7)
500   continue
c
c             summation for saved energy
c
      call rv2zero (rsbbf, 3, 7)
c
      do  600  b = 1,3
      do  600  f = 1,7
      do  600  t = 1,4
      rsbbf(b,f) = rsbbf(b,f) + rsbtbf(f,b,t)
600   continue
c
      call rv2sum (rsbbf, 3, 7, rsbf, 1)
c
```

```
      call rv2sum (rsbbf, 3, 7, rsbb, 2)
c
      call rvsum  (rsbb, 3, rsbtot)
c
c
      call rvzero (rafbq, 4)
      call rvzero (rafbx, 4)
      call rvzero (rafbn, 4)
      call rvzero (rafb,  4)
      call rvzero (raftq, 3)
      call rvzero (raftx, 3)
      call rvzero (raftn, 3)
      call rvzero (raft,  3)
c
      do  3100  s = 1,2
      t = s * 2 - 1
      do  3100  b = 1,3
      do  3030  f = 1,3
      rafq(f,s,b) = rhqtyp(t) * rab(b) * rbqtbf(f,b,s)
      rafx(f,s,b) = rhxtyp(t) * rab(b) * rbxtbf(f,b,s)
      rafn(f,s,b) = rhntyp(t) * rab(b) * rbntbf(f,b,s)
3030  continue
c
      do  3100  f = 1,3
      rafbq(b) = rafbq(b) + rafq(f,s,b)
      rafbx(b) = rafbx(b) + rafx(f,s,b)
      rafbn(b) = rafbn(b) + rafn(f,s,b)
c
      raftq(s) = raftq(s) + rafq(f,s,b)
      raftx(s) = raftx(s) + rafx(f,s,b)
      raftn(s) = raftn(s) + rafn(f,s,b)
3100  continue
c
c
      call rvsum  (rafbq, 3, rafbq(4))
      call rvsum  (rafbx, 3, rafbx(4))
      call rvsum  (rafbn, 3, rafbn(4))
      call rvvsum (rafbq, rafbx, rafb, 4)
      call rvvsum (rafb,  rafbn, rafb, 4)
      call rvsum  (raftq, 2, raftq(3))
      call rvsum  (raftx, 2, raftx(3))
      call rvsum  (raftn, 2, raftn(3))
      call rvvsum (raftq, raftx, raft, 3)
      call rvvsum (raft,  raftn, raft, 3)
c
c
      return
      end
```

```
#
c------ r b a s ----------------------------------------------------
c
c  function              - calculates base appliance energy use
c                          (space and water heating)
c
c  usage                 - call rbas
c
c  parameters            - none
c
c  subroutines used      - call ralt, rvdiv, rvscale, rvsum, rv2sum, rvzero
c
c-------------------------------------------------------------------
c  author                - anton toifelhardt
c  latest revision       - 77/11/10
c
c
      subroutine rbas
c
c
#include "model.com"
c
      integer f, b, t
c
c             total base appliance energy according to home type,
c             appliance and energy type
c
      do 100  t = 1,4
      do 100  b = 1,3
      call rvscale (rpztbf(1,b,t), 6, rhztyp(t), rubtbf(1,b,t))
      call rv2mul (rubtbf(1,b,t), rbztbf(1,b,t), 6,1, rubtbf(1,b,t),1)
100   continue
c
c             calculate energy use by alternative energy type technologies
c
c     ------------------------------------------
      if  ((rzzalf.eq.1).and.(rzcur.gt.rbyear))  call ralt
c     ------------------------------------------
c
c             base appliance energy according to appliance
c             and energy type
c
      call rv2zero (rubbf, 3, 7)
c
      do  200  b = 1,3
      do  200  f = 1,7
      do  200  t = 1,4
      rubbf(b,f) = rubbf(b,f) + rubtbf(f,b,t)
200   continue
c
c             base appliance energy according to energy type
c
      call rv2sum (rubbf, 3, 7, rubf, 1)
c
c             base appliance energy use per capita according to energy type
c
```

```
      call rvdiv   (rubf, 7, pop(rzreg), ribf)
c
c             total base appliance energy according to base appliance type
c
      call rv2sum (rubbf, 3, 7, rubb, 2)
c
c             total base appliance energy use
c
      call rvsum (rubb, 3, rubtot)
c
c             total base appliance energy use per capita
c
      ribtot = rubtot / pop(rzreg)
c
      return
      end
```

```
#
c------ r c a l c l --------------------------------------------------
c
c  function                  - calculates average energy demand per home type
c                              on the basis of heat losses, heating hours,
c                              and home size
c
c  usage                     - call rcalcl
c
c  parameters                - none
c
c  subroutines used          - none
c
c--------------------------------------------------------------------
c  author                    - anton toifelhardt
c  latest revision           - 77/11/10
c
c
        subroutine rcalcl
c
c
#include "model.com"
c
        integer t, b, f
c
c
        do  100  t = 1,4
        rpxx(t,1) = rpkcal(t) * rphthy(t) * rptemp(t) * 1.e-9
c
        rpnx(t,1) = rpxx(t,1) * rpnhms(t) * rpnins(t)
        rpqx(t,1) = rpxx(t,1) * rpqhms(t) * rpqins(t)
        rpxx(t,1) = rpxx(t,1) * rpxhms(t) * rpxins(t)
c
        rpnx(t,2) = rpnx(t,1)
        rpxx(t,2) = rpxx(t,1)
        rpqx(t,2) = rpqx(t,1)
c
        rpnx(t,3) = rpnwat(t)
        rpxx(t,3) = rpxwat(t)
        rpqx(t,3) = rpqwat(t)
100     continue
c
        do  200  t = 1,4
        do  200  b = 1,3
        rbztbf(7,b,t) = 0.
        rpztbf(7,b,t) = 0.
        do  200  f = 1,6
        rbztbf(f,b,t) = rxxfrc(t) * rbxtbf(f,b,t)
     1                  + rqqfrc(t) * rbqtbf(f,b,t)
        rpztbf(f,b,t) = rxxfrc(t) * rpxtbf(f,b,t) * rpxx(t,b)
     1                  + rqqfrc(t) * rpqtbf(f,b,t) * rpqx(t,b)
200     continue
c
        return
        end
c
```

```
c
c------ r c a l c 2 -------------------------------------------
c
c   function              - subroutine to calculate all submatrices
c                           related to per home consumption of fuel(f)
c                           as a function of appliance(b) in home type(t)
c                           for x (OLD), q (TOTAL NEW), z (TOTAL),
c                           and n (NEW) HOMES.
c
c   usage                 - call rcalc2
c
c   parameters            - none
c
c   subroutines used      - call rvdiv, rvnorm, rvsum, rvzero,
c                           call rv2sum, rv2zero
c
c--------------------------------------------------------------
c   author                - anton toifelhardt
c   latest revision       - 77/11/10
c
c
        subroutine rcalc2
c
#include "model.com"
c
        integer t, b, f
c
        call rvzero (rpnt, 4)
        call rvzero (rpxt, 4)
        call rvzero (rpqt, 4)
        call rvzero (rpzt, 4)
c
        call rv2zero (rpntb, 4, 3)
        call rv2zero (rpxtb, 4, 3)
        call rv2zero (rpqtb, 4, 3)
        call rv2zero (rpztb, 4, 3)
c
        call rvzero (rpnf, 7)
        call rvzero (rpxf, 7)
        call rvzero (rpqf, 7)
        call rvzero (rpzf, 7)
c
        call rvzero (rbxf, 7)
        call rvzero (rbqf, 7)
        call rvzero (rbzf, 7)
        call rvzero (rbnf, 7)
c
        call rv2zero (rpntf, 4, 7)
        call rv2zero (rpxtf, 4, 7)
        call rv2zero (rpqtf, 4, 7)
        call rv2zero (rpztf, 4, 7)
c
        call rvzero (rpnb, 3)
        call rvzero (rpxb, 3)
        call rvzero (rpqb, 3)
        call rvzero (rpzb, 3)
```

```
c
      call rv2zero (rpnbf, 3, 7)
      call rv2zero (rpxbf, 3, 7)
      call rv2zero (rpqbf, 3, 7)
      call rv2zero (rpzbf, 3, 7)
c
      do  200  f = 1,7
      do  200  t = 1,4
      do  200  b = 1,2
      rbnf(f) = rbnf(f) + rbntbf(f,b,t) * rhntyp(t)
      rbxf(f) = rbxf(f) + rbxtbf(f,b,t) * rhxtyp(t)
      rbqf(f) = rbqf(f) + rbqtbf(f,b,t) * rhqtyp(t)
200   rbzf(f) = rbzf(f) + rbztbf(f,b,t) * rhztyp(t)
c
      call rvdiv  (rbnf, 7, rhntyp(5), rbnf)
      call rvdiv  (rbxf, 7, rhxtyp(5), rbxf)
      call rvdiv  (rbqf, 7, rhqtyp(5), rbqf)
      call rvdiv  (rbzf, 7, rhztyp(5), rbzf)
c
      call rvnorm (rbnf, 7)
      call rvnorm (rbxf, 7)
      call rvnorm (rbqf, 7)
      call rvnorm (rbzf, 7)
c
c           summation subroutines discussed above.
c
      do  320  t = 1,4
      do  320  b = 1,3
      do  300  f = 1,6
      rpqtb(t,b)=rpqtb(t,b) + rpqtbf(f,b,t) * rpqx(t,b) *rbqtbf(f,b,t)
      rpntb(t,b)=rpntb(t,b) + rpntbf(f,b,t) * rpnx(t,b) *rbntbf(f,b,t)
      rpxtb(t,b)=rpxtb(t,b) + rpxtbf(f,b,t) * rpxx(t,b) *rbxtbf(f,b,t)
300   rpztb(t,b)=rpztb(t,b) + rpztbf(f,b,t) * rbztbf(f,b,t)
c
      do  320  f = 1,7
      rpqtf(t,f)=rpqtf(t,f) + rpqtbf(f,b,t) * rpqx(t,b) *rbqtbf(f,b,t)
      rpntf(t,f)=rpntf(t,f) + rpntbf(f,b,t) * rpnx(t,b) *rbntbf(f,b,t)
      rpxtf(t,f)=rpxtf(t,f) + rpxtbf(f,b,t) * rpxx(t,b) *rbxtbf(f,b,t)
      rpztf(t,f)=rpztf(t,f) + rpztbf(f,b,t) * rbztbf(f,b,t)
c
      rpqbf(b,f) = rpqbf(b,f)
     1        + rpqtbf(f,b,t) * rpqx(t,b) * rbqtbf(f,b,t) * rhqfrc(t)
      rpnbf(b,f) = rpnbf(b,f)
     1        + rpntbf(f,b,t) * rpnx(t,b) * rbntbf(f,b,t) * rhnfrc(t)
      rpxbf(b,f) = rpxbf(b,f)
     1        + rpxtbf(f,b,t) * rpxx(t,b) * rbxtbf(f,b,t) * rhxfrc(t)
      rpzbf(b,f) = rpzbf(b,f)
     1        + rpztbf(f,b,t) * rbztbf(f,b,t) * rhzfrc(t)
320   continue
c
c           now sum over these to create the one dimensional matrices
c
      call rv2sum (rpntb, 4, 3, rpnt, 2)
      call rv2sum (rpxtb, 4, 3, rpxt, 2)
      call rv2sum (rpqtb, 4, 3, rpqt, 2)
      call rv2sum (rpztb, 4, 3, rpzt, 2)
```

```
c
      call rv2sum (rpnbf, 3, 7, rpnf, 1)
      call rv2sum (rpxbf, 3, 7, rpxf, 1)
      call rv2sum (rpqbf, 3, 7, rpqf, 1)
      call rv2sum (rpzbf, 3, 7, rpzf, 1)
c
      call rv2sum (rpnbf, 3, 7, rpnb, 2)
      call rv2sum (rpxbf, 3, 7, rpxb, 2)
      call rv2sum (rpqbf, 3, 7, rpqb, 2)
      call rv2sum (rpzbf, 3, 7, rpzb, 2)
c
c           last but not least sum over one set of the above columns
c
      call rvsum (rpnb, 3, rpn)
      call rvsum (rpxb, 3, rpx)
      call rvsum (rpqb, 3, rpq)
      call rvsum (rpzb, 3, rpz)
c
      return
      end
```

```
#
c------ r g e t c o m ---------------------------------------------------
c
c  function                  - reads the next copy of the common-block from
c                              file 'ifile'
c
c  usage                     - call rgetcom (ifile, ireg, iend)
c
c  parameters    ifile       - input.  number of the input-file.
c                ireg        - output. region-number of the common-block.
c                iend        - output. error message
c                                      = 0: ok
c                                      > 0: end of input-file reached
c
c  subroutines used          - none
c
c------------------------------------------------------------------------
c  author                    - anton toifelhardt
c  latest revision           - 77/10/27
c
c
        subroutine rgetcom (ifile, ireg, iend)
c
c
#include "model.com"
c
#include "equiv.cmn"
c
        iend = 0
        read (ifile, end = 9000) ireg
c
        do  100  i = 1,ilen
100     read (ifile) (idumy(j,i), j = 1,100)
c
        go to 9999
9000    iend = 1
c
c
9999    return
        end
c
c
c
c------ r p u t c o m ---------------------------------------------------
c
c  function                  - writes a copy of the common-block to file
c                              'ifile'.
c
c  usage                     - call rputcom (ireg, ifile)
c
c  parameters    ireg        - input. number of the region.
c                ifile       - input. number of the output-file.
c
c  subroutines used          - none
c
c------------------------------------------------------------------------
```

```
c  author                  - anton toifelhardt
c  latest revision         - 77/10/27
c
c
        subroutine rputcom (ireg, ifile)
c
c
#include "model.com"
c
#include "equiv.cmn"
c
        write (ifile) ireg
        do  100  i = 1,ilen
100     write (ifile) (idumy(j,i), j = 1,100)
c
        return
        end
```

```
#
c------ r c o n v -----------------------------------------------------
c
c   function              - calculates new distribution of probabilities
c                           for base appliances and energy types
c                           (OLD and NEW HOMES)
c
c   usage                 - call rconv
c
c   parameters            - none
c
c   subroutines used      - call rcgetm, rconv2, rsplit
c
c----------------------------------------------------------------------
c   author                - anton toifelhardt
c   latest revision       - 77/11/07
c
c
          subroutine rconv
c
c
c                   calculates new distributions of probabilities for
c                   base appliances and energy types
c                   (OLD and NEW HOMES)
c
c
#include "model.com"
c
#include "conv.cmn"
c
          call rcgetm
          call rconv2 (dxsoch, dxnso, dxnch, dxhnhw, dxnhw, rbxsa)
          call rsplit (rbxsa, rbxtbf, rurfrc)
c
          call rcgetm
          call rconv2 (dnsoch, dnnso, dnnch, dnhnhw, dnnhw, rbnsa)
          call rsplit (rbnsa, rbntbf, rurfrc)
c
          return
          end
c
c
c------ r c o n v 2 ---------------------------------------------------
c
c   function              - calculates base appliance and energy type
c                           substitution for OLD HOMES (X) and changing
c                           construction policies in this respect for
c                           NEW HOMES (N).
c
c   usage                 - call rconv2 (dsoch, dnso, dnch, dhnhw, dnhw, x)
c
c   parameters  dsoch     - i/o.
c               dnso      - i/o.
c               dnch      - i/o.
c               dhnhw     - i/o.
c               dnhw      - i/o.
```

```
c                  x        - output. new distribution of probabilities.
c
c   subroutines used        - call rmvmul, rvnorm, rvscale.
c
c-----------------------------------------------------------------------
c   author                  - anton toifelhardt
c   latest revision         - 77/11/07
c
c
      subroutine  rconv2 (dsoch, dnso, dnch, dhnhw, dnhw, x)
c
c
#include "conv2.cmn"
c
      dimension dsoch(2,2), dnso(5,2), dnch(6,2)
      dimension dhnhw(2,2), dnhw(6,2), x(7,3,2)
      dimension hh(6)
c
      do  100  m = 1,2
c
      call rmvmul (rtsoch(1,1,m), dsoch(1,m), 2)
      call rmvmul (rtso   (1,1,m), dnso (1,m), 5)
      call rmvmul (rtch   (1,1,m), dnch (1,m), 6)
      call rmvmul (rthnhw(1,1,m), dhnhw(1,m), 2)
      call rmvmul (rthw   (1,1,m), dnhw (1,m), 6)
c
      call rvnorm  (dsoch(1,m), 2)
      call rvnorm  (dnso (1,m), 5)
      call rvnorm  (dnch (1,m), 6)
      call rvnorm  (dhnhw(1,m), 2)
      call rvnorm  (dnhw (1,m), 6)
c
      call rvscale (dnso (1,m), 5, dsoch(1,m), x(1,1,m))
      call rvscale (dnch (1,m), 6, dsoch(2,m), x(1,2,m))
      call rvscale (dnhw (1,m), 6, dhnhw(1,m), x(1,3,m))
c
100   continue
c
      return
      end
c
c
c
c------ r c g e t m ----------------------------------------------------
c
c  function                 - reads conversion matrices from file '4'
c                             setfil (4,'=convmat')
c
c  usage                    - call rcgetm
c
c  parameters               - none
c
c  subroutines used         - none
c
c-----------------------------------------------------------------------
c  author                   - anton toifelhardt
```

```
c   latest revision        - 77/11/07
c
c
c
        subroutine rcgetm
c
c
#include "conv2.cmn"
c
        do  100  m = 1,2
        call rcget2 (4, rtsoch(1,1,m), 2)
        call rcget2 (4, rtso  (1,1,m), 5)
        call rcget2 (4, rtch  (1,1,m), 6)
        call rcget2 (4, rthnhw(1,1,m), 2)
        call rcget2 (4, rthw  (1,1,m), 6)
100     continue
c
        return
        end
c
c
c
c------ r c g e t 2 ---------------------------------------------------
c
c   function                   - reads an array from file 'iin'
c
c   usage                      - call rcget2 (iin,x,idim)
c
c   parameters    iin          - input.  number of input-file.
c                 x            - output. array of dimension (idim,idim).
c                 idim         - input.  dimension of x.
c
c   subroutines used           - none
c
c----------------------------------------------------------------------
c   author                     - anton toifelhardt
c   latest revision            - 77/11/07
c
c
c
        subroutine  rcget2 (iin, x, idim)
c
c
        dimension  x(idim,idim)
c
        read  (iin)  x
c
c
        return
        end
```

```
#
c------ r c v i n i t ---------------------------------------------
c
c   function                 - initializes the conversion matrices
c
c   usage                    - call rcvinit
c
c   parameters               - none
c
c   subroutines used         - call rciget, rciput, rcvinl, rsplit
c ...................
c
c   note                     - this subroutine is called only from rinit
c
c------------------------------------------------------------------
c   author                   - anton toifelhardt
c   latest revision          - 77/10/27
c
c
        subroutine rcvinit
c
c
#include "model.com"
c
#include "conv.cmn"
#include "conv2.cmn"
c
c
        call rciget (1)
        call rcvinl (dxsoch, dxnso, dxnch, dxhnhw, dxnhw, rbxsa)
        call rsplit (rbxsa, rbxtbf, rurfrc)
        call rciput (4)
c
        call rciget (1)
        call rcvinl (dnsoch, dnnso, dnnch, dnhnhw, dnnhw, rbnsa)
        call rsplit (rbnsa, rbntbf, rurfrc)
        call rciput (4)
c
        return
        end
c
c
c------ r c v i n l -----------------------------------------------
c
c   function                 - initializes a set of conversion matrices
c
c   usage                    - call rcvinl (dsoch, dnso, dnch, dhnhw, dnhw, x)
c
c   parameters    dsoch      - output. single oven - central heating
c                 dnso       - output. single oven.
c                 dnch       - output. central heating.
c                 dhnhw      - output. hot water - non hot water.
c                 dnhw       - output. hot water.
c                 x          - output. new distribution of probabilities.
c
c   subroutines used         - call rvdiv, rvnorm, rvsum.
```

```
c
c------------------------------------------------------------------------
c   author                     - anton toifelhardt
c   latest revision            - 77/10/27
c
c
        subroutine  rcvinl (dsoch, dnso, dnch, dhnhw, dnhw, x)
c
c
#include "conv2.cmn"
c
        dimension  dsoch(2,2), dnso(5,2), dnch(6,2)
        dimension  dhnhw(2,2), dnhw(6,2), x(7,3,2)
c
        do  1000  k = 1,2
c
        call rvsum (x(1,1,k), 5, dsoch(1,k))
        call rvsum (x(1,2,k), 6, dsoch(2,k))
        call rvsum (x(1,3,k), 6, dhnhw(1,k))
        dhnhw(2,k) = 1. - dhnhw(1,k)
c
        call rvdiv (x(1,1,k), 5, dsoch(1,k), dnso (1,k))
        call rvdiv (x(1,2,k), 6, dsoch(2,k), dnch (1,k))
        call rvdiv (x(1,3,k), 6, dhnhw(1,k), dnhw (1,k))
c
        do  500  j = 1,2
        call rvnorm (rtsoch(1,j,k), 2)
        call rvnorm (rthnhw(1,j,k), 2)
500     continue
c
        do  600  j = 1,5
        call rvnorm (rtso  (1,j,k), 5)
600     continue
c
        do  700  j = 1,6
        call rvnorm (rtch  (1,j,k), 6)
        call rvnorm (rthw  (1,j,k), 6)
700     continue
1000    continue
c
        return
        end
c
c
c------ r c i g e t -----------------------------------------------------
c
c  function                   - read conversion matrices from file 'iin'
c
c  usage                      - call rciget (iin)
c
c  parameters    iin          - input. number of input-file.
c
c  subroutines used           - call rcigtl.
c
c------------------------------------------------------------------------
c  author                     - anton toifelhardt
```

```
c  latest revision      - 77/10/27
c
c
         subroutine rciget (iin)
c
c
#include "conv2.cmn"
c
         do  100  k = 1,2
         call rcigtl (iin, rtsoch(1,1,k), 2)
         call rcigtl (iin, rtso  (1,1,k), 5)
         call rcigtl (iin, rtch  (1,1,k), 6)
         call rcigtl (iin, rthnhw(1,1,k), 2)
         call rcigtl (iin, rthw  (1,1,k), 6)
c
100      continue
         return
         end
c
c
c------ r c i g t l -----------------------------------------------
c
c  function                 - reads matrix 'x' from file 'iin'
c
c  usage                    - call rcigtl (iin, x, idim)
c
c  parameters    iin        - input.  number of input-file.
c                x          - output. matrix of dimension (idim,idim)
c                idim       - input.  dimensions of matrix x.
c
c  subroutines used         - none
c
c-----------------------------------------------------------------------
c  author                   - anton toifelhardt
c  latest revision          - 77/10/27
c
c
         subroutine rcigtl (iin, x, idim)
c
c
         dimension  x(idim, idim)
c
         read  (iin,5000)
c
         do  100  i = 1,idim
         read  (iin,5100)  (x(i,j), j=1,idim)
100      continue
         return
c
5000     format (1x/1x/1x/1x)
5100     format (6f10.7)
c
         end
c
c
c
```

```
c------ r c i p u t -----------------------------------------------
c
c   function                  - writes the conversion matrices to file 'iout'
c
c   usage                     - call rciput (iout)
c
c   parameters    iout        - input. number of output-file.
c
c   subroutines used          - call rciptl
c
c------------------------------------------------------------------
c   author                    - anton toifelhardt
c   latest revision           - 77/10/27
c
c
        subroutine  rciput (iout)
c
c
#include "conv2.cmn"
c
        do  100  k = 1,2
        call rciptl (iout, rtsoch(1,1,k), 2)
        call rciptl (iout, rtso  (1,1,k), 5)
        call rciptl (iout, rtch  (1,1,k), 6)
        call rciptl (iout, rthnhw(1,1,k), 2)
        call rciptl (iout, rthw  (1,1,k), 6)
100     continue
c
        return
        end
c
c------ r c i p t l -----------------------------------------------
c
c   function                  - writes matrix 'x' to file 'iout'
c
c   usage                     - call rciptl (iout,x,idim)
c
c   parameters    iout        - input. number of output-file.
c                 x           - input. matrix of dimension idim by idim.
c                 idim        - input. dimensions of matrix x
c
c   subroutines used          - none.
c
c------------------------------------------------------------------
c   author                    - anton toifelhardt
c   latest revision           - 77/10/27
c
c
        subroutine rciptl (iout, x, idim)
c
c
        dimension  x (idim, idim)
c
        write (iout) x
        return
        end
```

```
#
c------ r d e m f c ----------------------------------------------------
c
c   function                - This subroutine calculates values for the
c                             demolition function dependent on the values
c                             of 'rdm'
c
c   usage                   - call rdemfc
c
c   parameters              - none
c
c   subroutines used        - call ragedtr, rvscale
c
c....................
c
c   note                    -
c
c-----------------------------------------------------------------------
c   author                  - anton toifelhardt
c   latest revision         - 78/08/23 erwin poenitz
c
c
        subroutine rdemfc
c
c
#include "model.com"
c
        do  100  i = 1,rmxage
100     rdem(i) = rdm(1) * exp (rdm(2) * float (i-rmxage))
c
c               initial distribution of homes
c
        call ragedtr
c
c               the distribution produced is normalized, so
c               multiply by 'rhztyp(5)'
c
        call rvscale (rhy, rmxage, rhztyp(5), rhy)
c
        do  200  j = 1,4
        call rvscale (rhy, rmxage, rhzfrc(j), rhytyp(1,j))
200     continue
c
        return
        end
c
c
c
#
c------ r d e m o l -----------------------------------------------------
c
c   function                - demolition subroutine
c
c   usage                   - call rdemol
c
c   parameters              - none
```

```
c
c  subroutines used      - call rvzero, sum, rvdiv
c
c....................
c
c  note                  - NEW HOMES (n)
c                          not yet added to TOTAL NEW HOMES (q)
c
c-----------------------------------------------------------------------
c  author                - anton toifelhardt
c  latest revision       - 78/08/23, erwin poenitz
c
c
        subroutine rdemol
c
c
#include "model.com"
c
        call rvzero (rdxtyp, 5)
        call rvzero (rdqtyp, 5)
        call rvzero (rhdfrc, 4)
c
c       -----------------
        if  (rzzrnw.eq.0)  go to 1000
c       -----------------
c
c               demolished homes by type for OLD and TOTAL NEW HOMES
c
        do 600 j = 1,4
        n = rzcur - rbyear
c
        do 510 l = 1,n
        rdqtyp(j) = rdqtyp(j) + rhytyp(l,j) * rdem(l)
510     continue
c
        npl = n + 1
c
c               homes older than rmxage years will be not demolished
c
        do 520 l = npl,rmxage
        rdxtyp(j) = rdxtyp(j) + rhytyp(l,j) * rdem(l)
        rhdtyp(j) = rdxtyp(j) + rdqtyp(j)
520     continue
600     continue
c
c
c               total demolished homes
c
        call rvsum (rdqtyp, 4, rdqtyp(5))
        call rvsum (rdxtyp, 4, rdxtyp(5))
        call rvsum (rhdtyp, 4, rhdtyp(5))
c
c               demolition fraction by type
c
        call rvdiv (rdqtyp, 4, rdqtyp(5), rdqfrc)
        call rvdiv (rdxtyp, 4, rdxtyp(5), rdxfrc)
```

```
      call rvdiv (rhdtyp, 4, rhdtyp(5), rhdfrc)
c
1ØØØ  continue
c
c
      return
      end
```

```
#
c------ r e s e n g ------------------------------------------------
c
c  function               - prepares and writes output to file 'resengyout'
c
c  usage                  - call reseng
c
c  parameters             - none
c
c  subroutines used       - call rv2mul, rvscale
c
c....................
c
c  note                   -
c
c-------------------------------------------------------------------
c  author                 - anton toifelhardt
c  latest revision        - 77/00/00
c
c
        subroutine  reseng
c
c
#include "model.com"
c
        dimension hl(4,7), h2(4), h3(7)
        integer t,f
c
c       ---------------
        if (rzreg.eq.1)  go to 1000
c       ---------------
c
        call rv2mul  (rpztf, rhztyp, 4, 7, hl, 1)
        call rv2mul  (rpzt,  rhztyp, 4, 1, h2, 1)
        call rvscale (ronf,  7, rhztyp(5), h3)
        h4  = rpn * rhztyp(5)
c
1000    continue
        return
        end
```

```
#
c------ r i n c r h -------------------------------------------
c
c  function              - calculates  INCREMENTAL HOMES  in  order  to
c                          house new families
c
c  usage                 - call rincrh
c
c  parameters            - none
c
c  subroutines used      - call rvsum
c
c....................
c
c  note                  -
c
c---------------------------------------------------------------------
c  author                - anton toifelhardt
c  latest revision       - 77/00/00
c
c
        subroutine  rincrh
c
c
#include "model.com"
c
        do  300  i = 1,4
        if (i.ge.3)  go to 100
        rzl = pifam (rzreg) - pifrur (rzreg)
        rz2 = pfam  (rzreg) - pfrur  (rzreg) - rrbhom
c
        go to 200
100     rzl = pifrur (rzreg)
        rz2 = pfrur  (rzreg) - rruhom
200     continue
c
c               add unsatisfied demand
c
        if (rz2.gt.0.)  rzl = rzl + rz2 * 0.05
        if (rzl.lt.0.)  rzl = 0.
300     rhityp(i) = rzl * rhifrc(i)
c
        call rvsum (rhityp, 4, rhityp(5))
c
        return
        end
c
c
c
c------ r n e w h ------------------------------------------
c
c  function              - calculates NEW HOMES of the
c                          current simulation year
c
c  usage                 - call rnewh
c
```

```
c  parameters              - none
c
c  subroutines used        - call rvsum, rvdiv
c
c....................
c
c  note                    - NEW HOMES =  construction to
c                            house new families plus REPLACEMENT HOMES
c                            to  substitute for demolished  homes
c
c----------------------------------------------------------------------
c  author                  - anton toifelhardt
c  latest revision         - 77/00/00
c
c
        subroutine  rnewh
c
c
#include "model.com"
c
        do  100  i = 1,4
        rhntyp(i) = rhityp(i)
c       ----------------
        if (rzzrnw.eq.1) rhntyp(i) = rhntyp(i) + rhdtyp(i)
        if (rzzrnw.eq.2) rhntyp(i) = rhntyp(i) + rhdtyp(5)  * rhifrc(i)
c       ----------------
100     continue
c
        call rvsum (rhntyp, 4, rhntyp(5))
        call rvdiv (rhntyp, 4, rhntyp(5), rhnfrc)
c
        return
        end
```

```
#
c------ r i n i t ----------------------------------------------------
c
c  function                 - initializes the values for a new run
c
c  usage                    - call rinit
c
c  parameters               - none
c
c  subroutines used         - call setfil 1,3,4,8,12, closef 8,12
c                             rvzero,  rpread, rv3zero, rcvinit, rurspl,
c                             rvscale, rvdiv,  rdemfc,  rincrh,  rdemol, rnewh,
c                             rvvdiv,  rvvms,  rvscale, rcalcl,  rbas,   rcalc2
c                             rsec,    rsumup, rputcom, closef 1,3,4
c....................
c
c  note                     - rinit is only called in the first simulation year
c
c-----------------------------------------------------------------------
c  author                   - anton toifelhardt
c  latest revision          - 78/08/12, erwin poenitz
c
c
        subroutine rinit
c
c
#include "model.com"
c
c                residential data base
c
        call setfil (1,'rdatabase')
c
c                work files
c
        call setfil (3,'=write')
        call setfil (4,'=convmat')
c
c                binary output file for other submodels
c
c       call closef (8)
c       call setfil (8,'resengyout')
c
c                binary input file from "pmodel"
c
        call closef (12)
        call setfil (12,'=pdata')
c
c                initialize the population data
c
        call closef (8)
        call setfil(8,'pdata ')
        call rpread
c
c                initialize the residential data
c
        call rv3zero (rbqtbf, 7, 3, 4)
```

```
      call rv3zero (rpqtbf, 7, 3, 4)
c
      call rvzero  (rhqtyp, 5)
      call rvzero  (rdxtyp, 5)
      call rvzero  (rdqtyp, 5)
      call rvzero  (rhityp, 5)
      call rvzero  (rhntyp, 5)
c
      do   50  i = 1,4
      rpqins(i) = 1.
      rpxins(i) = 1.
50    rpnins(i) = 1.
c
      do  1000  rzreg = 1,10
c
c     ----------------------
      if (rregn(rzreg).ne.0)  goto 200
c     ----------------------
c
c             skip the next region
c             file 'rdatabase' (244 lines)

      do  70  i = 1,80
      read (1,2222)
70    continue
c
      do  80  i = 1,82
      read (1,2222)
80    continue
c
      do  90  i = 1,82
      read (1,2222)
90    continue
c
      go to 1000
c
200   read (1,2110)   rscfrc, rschyr, rscsat
      read (1,2120)   rpsk
      read (1,2130)   rpkcal
      read (1,2130)   (rpqhms(i), i=1,4)
      read (1,2130)   (rpxhms(i), i=1,4)
      read (1,2130)   (rpnhms(i), i=1,4)
      read (1,2130)   rphthy
      read (1,2140)   rptemp
      read (1,2140)   rpqwat
      read (1,2140)   rpxwat
      read (1,2140)   rpnwat
      read (1,2150)   rbxsa
      read (1,2160)   rpxtbf
      read (1,2150)   rbnsa
      read (1,2160)   rpntbf
      read (1,2170)   rhifrc
      read (1,2160)   rurfrc
      read (1,2180)   rhzsiz, rhztyp(5)
      read (1,2190)   rdm
      read (1,2190)   rat
```

```
      read (1,2200)   ragval, ragunk
c
c
c            initialization of conversion; shifts in
c            energy type and base  appliance  mix
c
      call rcvinit
c
c            urban - rural  split for homes
c
      call rurspl
c
      call rvscale (rhztyp, 5, 1., rhxtyp)
      call rvzero  (rhqtyp, 5)
c
      call rvdiv  (rhztyp, 4, rhztyp(5), rhzfrc)
      call rvdiv  (rhxtyp, 4, rhxtyp(5), rhxfrc)
      call rvzero (rhqfrc, 4)
c
c            calculate demolition function
c
      call rdemfc
c
c            INCREMENTAL HOMES in order to house new families
c
      call rincrh
c
c            demolition of homes in a given year
c
      call rdemol
c
c            NEW HOMES in a given year =
c            INCREMENTAL HOMES + REPLACEMENT HOMES
c
      call rnewh
c
c            fraction of OLD and NEW HOMES
c
      call rvvdiv (rhxtyp, rhztyp, rxxfrc, 5)
      call rvvdiv (rhqtyp, rhztyp, rqqfrc, 5)
c
c            calculate average home-size
c
      call rvvms   (rpxhms, rhxfrc, 4, rpxhms(5))
      call rvvms   (rpqhms, rhqfrc, 4, rpqhms(5))
      call rvscale (rpxhms, 5, 1., rpzhms)
      rpnhms(5) = 0.
c
c            calculate home-size per capita
c
      call rvdiv   (rpxhms, 5, pfsize(rzreg), rpxlac)
      call rvdiv   (rpzhms, 5, pfsize(rzreg), rpzlac)
      call rvzero  (rpqlac, 5)
      call rvzero  (rpnlac, 5)
c
c            no alternative energy types in the first year
```

```
c
c               summation
c
      do  300  i = 1,4
      do  300  j = 1,3
      call rvscale (rbntbf(1,j,i), 7, 1., rbqtbf(1,j,i))
300   continue
c
      call rcalcl
c
c               calculate base appliance energy use
c
c     ----------------
      if (rzzbas.ne.0)  call rbas
c     ----------------
c
      call rcalc2
c
c               calculate secondary appliance energy use
c
c     ----------------
      if (rzzsec.ne.0)  call rsec
c     ----------------
c
c               sum up over regions
c
c
c               initialise the summation values
c
      call rvzero (rctot,  6)
      call rvzero (rcdemr, 7)
c
      call rsumup
c
c               copy the whole common-block to file '=write'
c
      call rputcom (rzreg, 3)
c
1000  continue
c
      call closef ( 1)
      call closef ( 3)
      call closef ( 4)
c
c
2110  format (/6(/10x,7f8.2))
2120  format (10x,7f10.6,/,10x,7f10.6)
2130  format (10x,4f9.3)
2140  format (10x,4f9.5)
2150  format (6(/10x,7f8.3))
2160  format (12(/10x,7f8.3))
2170  format (10x,4f8.2)
2180  format (10x,3f10.0)
2190  format (10x,2f10.5)
2200  format (3(/10x,8f8.0))
2222  format (1x)
c
      return
      end
```

```
#
c------ r m o d e l ------------------------------------------
c
c  function                  - level 1, calls all other subroutines
c
c  usage                     - call rmodel (izcur)
c
c  parameters   izcur        - input. the current simulation year
c
c  subroutines used          - call   rupdat, rdemol, rconv,  rurspl, rvsum
c                              rvdiv, rincrh, rnewh,  rcalcl, rbas,   rcalc2
c                              rsec
c....................
c  note                      - at least one of the policy switches rzzbas or
c                              rzzsec must be set to 1, otherwise no energy
c                              use is calculated
c----------------------------------------------------------------------
c  author                    - anton toifelhardt
c  latest revision           - 77/Ø8/26
c
c
        subroutine  rmodel (izcur)
c
c
#include "model.com"
c
        integer t, b, f, yrblt
c
c       --------------------
        if (izcur.eq.rbyear)   return
c       --------------------
c
        rzcur = izcur
c
c               update parameters, substract demolished homes,
c               add NEW HOMES, calculate new averages
c
        call rupdat
c
c               conversion; shifts in energy
c               type and base  appliance mix
c
c       ---------------
        if(rzzcnv.ne.Ø)   call rconv
c       ---------------
c
c               split homes into rural and urban components
c
        call rurspl
c
c               INCREMENTAL HOMES in order to house new families
c
        call rincrh
c
c               demolition of homes subroutine
c
```

```
      call rdemol
c
c             calculate NEW HOMES = INCREMENTAL + REPLACEMENT
c
      call rnewh
c
c             calculate average annual energy demand per home
c
      call rcalcl
c
c             calculate base appliance energy use
c
c     ----------------
      if (rzzbas.ne.0)  call rbas
c     ----------------
c
c             calculate aggregated matrices
c
      call rcalc2
c
c             calculate sec.appliance energy use
c
c     ----------------
      if (rzzsec.ne.0)  call rsec
c     ----------------
c
      return
      end
```

```
#
c------ r p r e a d ----------------------------------------
c
c   function                - reads population data from file =pdata
c
c   usage                   - call rpread
c
c   parameters              - none
c
c   subroutines used        - none
c
c....................
c
c   note                    -
c
c-----------------------------------------------------------------
c   author                  - anton toifelhardt
c   latest revision         - 77/00/00
c
c
        subroutine rpread
c
c
#include "model.com"
c
        read (12)  pop,     prural
        read (12)  pfam,    pfrur
        read (12)  pifam,   pifrur
        read (12)  pfsize
        write (8,6500) pop, prural, pfam, pfrur, pifam, pifrur
6500    format(10F8.0)
        write (8,6600) pfsize
6600    format(10F8.6)
c
c
        return
        end
```

```
#
c------ r s e c ---------------------------------------------
c
c   function              - calculates secondary appliance values
c                           (refrigerators, freezers, tv., etc.)
c                           plus calculates the overall totals
c
c
c   usage                 - call rsec
c
c   parameters            - none
c
c   subroutines used      - call rvzero, rvsum, rvscale, rvdiv
c
c....................
c
c   note                  -
c
c-------------------------------------------------------------------
c   author                - anton toifelhardt
c   latest revision       - 77/00/00
c
c
        subroutine rsec
c
c
#include "model.com"
c
        call rvzero (rusf,   7)
        call rvzero (rustot, 1)
c
        if (rzzsec.ne.1)   goto 500
        if (rzcur.eq.rbyear) goto 250
c
c               secondary appliance fractions
c
        do  200  j = 1,14
c
        if (rschyr(j).eq.0.)  go to 100
c
        rcoef(j) = 1. - exp(-.693 / rschyr(j))
        go to 150
100     rcoef(j)  =  0.
150     rscfrc(j) = rscfrc(j) + (rscsat(j) - rscfrc(j)) * rcoef(j)
200     continue
250     continue
c
c               secondary appliance energy use.
c
        do  300  j = 1,14
300     rusk(j) = rscfrc(j) * rpsk(j)
c
c               fraction of homes owning this particular appliance
c
        call rvsum (rusk, 12, rscf(1))
        rscf(2) = rusk(13) + rusk(14)
```

```
c
c               total s.a. energy according to appliance type
c
      call rvscale (rusk, 14, rhztyp(5), rusk)
c
c               energy used for all secondary appliances
c
      call rvsum (rscf, 2, rsc)
c
c               total s.a. energy according to energy type
c
      call rvzero  (rusf, 7)
      call rvscale (rscf, 2, rhztyp(5), rusf)
c
c               sec. appl.  energy use per capita according to energy type
c
      call rvdiv (rusf, 2, pop(rzreg), risf)
c
c               total secondary appliance energy use
c
      call rvsum (rusf, 2, rustot)
c
c               total secondary appliance energy use per capita
c
      ristot  = rustot / pop(rzreg)
c
c               ++++++++++++++++
c               the grand totals
c               ++++++++++++++++
c
500   do  700  j = 1,7
c
c               total energy consumed by energy type
c
700   ruwf(j) = rubf(j) + rusf(j)
c
c               intensiveness for all energy used in all homes
c
      call rvdiv (ruwf, 7, pop(rzreg), riwf)
      call rvdiv (ruwf, 7, rhztyp(5).  rhiwf)
c
c               total energy consumed
c
      ruwtot = rubtot + rustot
c
c               intensiveness of all energy used in all homes
c
      riwtot  = ruwtot / pop(rzreg)
      rhiwtot = ruwtot / rhztyp(5)
c
c               total cumulative energy
c
      ruwcum = ruwcum + ruwtot
c
c               demands
c
```

```
      call rvdiv (ruwf, 7, 1., demr)
c
c             convert into kilowatt
c
      call rvdiv (ruwf, 7, 860., rkw)
c
c             sum up
c
      rsum(1) = demr(1)
c
      do 800 j = 2,7
800   rsum(j) = rsum(j-1) + demr(j)
c
c
      return
      end
```

```
#
c------ r s u b r ---------------------------------------------
c
c   function               - collection of minor subroutines
c
c   usage                  - call subroutine
c
c   parameters             - yes
c
c   subroutines used       - only in the case of subroutine rvnorm, which
c                            calls rvsum and rvdiv
c....................
c
c   note                   -
c
c---------------------------------------------------------------------
c   author                 - anton toifelhardt
c   latest revision        - 77/00/00
c
c
c
c          ===================
           subroutine  rvdiv (x,n,c,y)
c          ===================
c
c          function:        y(i) = x(i) / c    (c <>0.)
c          --------      or y(i) = x(i)        (c = 0.)    i = 1..n
c
c
           dimension  x(n), y(n)
c
           z = c
           if (z.eq.0.) z = 1.
           do  100  i = 1,n
100        y(i) = x(i) / z
           return
           end
c
c
c          ====================
           subroutine  rvvdiv (x,y,z,n)
c          ====================
c
c          function:        z(i) = x(i) / y(i)    (y(i) > 1.)
c          --------      or z(i) = x(i)           (y(i) = 0.)    i = 1..n
c
c
           dimension  x(n), y(n), z(n)
           dimension  w(10)
c
           do   50  i = 1,n
           w(i) = y(i)
           if  (w(i).eq.0.)   w(i) = 1.
50         continue
c
           do  100  i = 1,n
```

```
100     z(i) = x(i) / w(i)
        return
        end
c
c
c       ====================
        subroutine  rmvmul (a,b,n)
c       ====================
c
c       function:       c(i) = b(i)                          i = 1..n
c       --------        b(i) = sum  a(i,j) * c(j)            i = 1..n
c                              j=1,n
c
        dimension  a(n,n), b(n)
        dimension  c(6)
        double precision  s
c
        do  100  i=1,n
100     c(i) = b(i)
c
        do  300  i=1,n
        s = 0.d0
        do  200  j=1,n
200     s = s + dble(a(i,j)) * dble(c(j))
300     b(i) = sngl(s)
        return
        end
c
c
c       ====================
        subroutine  rv2mul (x,y,n1,n2,z,ind)
c       ====================
c
c       function:       z(i,j) = x(i,j) * y(i)  (ind = 1)
c       --------        z(i,j) = x(i,j) * y(j)  (ind > 1)  i = 1..n1
c                                                          j = 1..n2
c
c
        dimension  x(n1,n2), y(1), z(n1,n2)
c
        if (ind.gt.1)  go to 200
        do  100  i = 1,n1
        do  100  j = 1,n2
100     z(i,j) = x(i,j) * y(i)
        return
200     do  300  i = 1,n1
        do  300  j = 1,n2
300     z(i,j) = x(i,j) * y(j)
        return
        end
c
c
c       ====================
        subroutine  rvnorm (v,n)
c       ====================
c
```

```
c       function:       normalizes a vector 'v' of length 'n'
c       --------
c
        dimension  v(n)
c
        call rvsum (v,n,s)
        call rvdiv (v,n,s,v)
        return
        end
c
c
c       =====================
        subroutine  rvscale (a,n,s,b)
c       =====================
c
c       function:       b(i) = a(i) * s     i = 1..n
c       --------
c
c
        dimension a(n),b(n)
c
        do  100  i = 1,n
        b(i) = a(i)  * s
100     continue
        return
        end
c
c
c       ===================
        subroutine  rvsum (x,n,y)
c       ===================
c
c       function:       y =  sum  x(i)
c       --------            i=1.n
c
c
        dimension  x(n)
c
        y = 0.
        do  100  i =1,n
100     y = y + x(i)
c
        return
        end
c
c
c       ====================
        subroutine  rv2sum (x,n1,n2,s,ind)
c       ====================
c
c       function:       ind = 1:  s(i) =  sum   x(j,i)   i = 1..n2
c       --------                         j=1,n1
c
c                       ind > 1:  s(i) =  sum   x(i,j)   i = 1..n1
c                                        j=1,n2
c
```

```
c
      dimension  x(n1,n2), s(1)
c
      if (ind.gt.1)  go to  200
      do  100  i = 1,n2
      s(i) = 0.
      do  100  j = 1,n1
100   s(i) = s(i) + x(j,i)
      return
200   do  300  i = 1,n1
      s(i) = 0.
      do  300  j = 1,n2
300   s(i) = s(i) + x(i,j)
      return
      end
c
c
c     ====================
      subroutine  rvzero (x,n)
c     ====================
c
c     function:       x(i) = 0.        i = 1..n
c     --------
c
c
      dimension  x(n)
c
      do  100  i = 1,n
100   x(i) = 0.
c
      return
      end
c
c
c     =====================
      subroutine  rv2zero (x,n1,n2)
c     =====================
c
c     function:       x(i,j) = 0.      i = 1..n1;  j = 1..n2
c
c
      dimension  x(n1,n2)
c
      do  100  i = 1,n2
      do  100  j = 1,n1
100   x(j,i) = 0.0
c
      return
      end
c
c
c     =====================
      subroutine  rv3zero (x,n1,n2,n3)
c     =====================
c
c     function:       x(i,j,k) = 0.     i = 1..n1;  j = 1..n2
```

```
c       --------                                       k = 1..n3
c
        dimension  x(n1,n2,n3)
c
        do  100  i = 1,n2
        do  100  j = 1,n1
        do  100  k = 1,n3
100     x(j,i,k) = 0.0
c
        return
        end
c
c
c
c
c       ==================
        subroutine rvvms (x, y, n, z)
c       ==================
c
c       function:       z =  sum  x(i) * y(i)
c       --------             i=1,n
c
        dimension x(n), y(n)
        double precision  s
c
        s = 0.d0
        do  100  i = 1,n
        s = s + dble(x(i)) * dble(y(i))
100     continue
        z = sngl(s)
c
        return
        end
c
c
c
c       ===================
        subroutine rvvsum (x, y, z, n)
c       ===================
c
c       function:       z(i) = x(i) + y(i)     i = 1..n
c
c
        dimension x(n), y(n), z(n)
c
        do  100  i = 1,n
        z(i) = x(i) + y(i)
100     continue
c
        return
        end
```

```
#
c------ r s p l i t ---------------------------------------------------
c
c   function                - to split matrices in two identical ones
c
c   usage                   - call rsplit (x, r, f)
c
c   parameters              - x  input.
c                             r  output.
c                             f  input. fraction
c
c   subroutines used        - call rvsum, rvdiv, rvscale
c
c....................
c
c   note                    -
c
c----------------------------------------------------------------------
c   author                  - anton toifelhardt
c   latest revision         - 77/00/00
c
c
        subroutine rsplit (x, r, f)
c
c
        dimension x(7,3,2), r(7,3,4), f(7,3,4), sum(3)
c
        do  500  m = 1,2
        m2 = m + 2
c
        call rvsum (x(1,3,m), 6, sum3)
c
        do  200  j = 1,3
        do  200  i = 1,6
        r(i,j,m)  = x(i,j,m) * f(i,j,m)
200     r(i,j,m2) = x(i,j,m) * f(i,j,m2)
c
c               normalize the matrix
c
        do  400  k = m,4,2
        do  300  j = 1,3
c
        call rvsum (r(1,j,k), 6, sum(j))
300     continue
c
        sum2 = sum(1) + sum(2)
        call rvdiv   (r(1,1,k), 6, sum2,   r(1,1,k))
        call rvdiv   (r(1,2,k), 6, sum2,   r(1,2,k))
        call rvdiv   (r(1,3,k), 6, sum(3), r(1,3,k))
        call rvscale (r(1,3,k), 6, sum3,   r(1,3,k))
400     continue
500     continue
c
        return
        end
c
```

```
c
c
c------ r s u m u p ------------------------------------------------
c
c  function              - sums up some variables over time
c
c  usage                 - call rsumup
c
c  parameters            - none
c
c  subroutines used      - none
c
c....................
c
c  note                  - this is useful to calculate cumulative flows
c                          over time, e. g. the consumption of
c                          non-renewable energy sources over the simulation
c                          period for one region
c--------------------------------------------------------------------
c  author                - anton toifelhardt
c  latest revision       - 77/Ø8/26 erwin poenitz
c
c
      subroutine rsumup
c
c
#include "model.com"
c
      rctot(1) = rctot(1) + rhityp(5)
      rctot(2) = rctot(2) + rhdtyp(5)
      rctot(3) = rctot(3) + rhntyp(5)
      rctot(4) = rctot(4) + 1.
      rctot(5) = rctot(5) + 1.
      rctot(6) = rctot(6) + 1.

c
      do 1ØØ  i = 1,7
1ØØ   rcdemr(i) = rcdemr(i) + demr(i)
c
      return
      end
c
c
c
c------ r u r s p l ------------------------------------------------
c
c  function              - to split single family homes and apartments
c                          into an urban and rural fraction
c
c  usage                 - call rurspl
c
c  parameters            - none
c
c  subroutines used      - call rvdiv, rvscale
c
c....................
```

```
c
c  note                    -
c
c-------------------------------------------------------------------------
c  author                  - anton toifelhardt
c  latest revision         - 77/00/00
c
c
        subroutine rurspl
c
c
#include "model.com"
c
c
c       ----------------------------
        if (pfam(rzreg).ge.rhztyp(5))  go to 100
c       ----------------------------
c
c               total homes

        rhztyp(5) = pfam(rzreg)
        rzl  = rhzsiz(1) + rhzsiz(2)
c
        call rvdiv   (rhzsiz, 2, rzl, rhzsiz)
        call rvscale (rhzsiz, 2, rhztyp(5), rhzsiz)
c
c       ----------------
100     if (rzreg.eq.10)  go to 200
c       ----------------
c
c               homes per  family
c
        yk = rhztyp(5) / pfam(rzreg)
c
c               number of rural and urban homes
c
        rruhom = pfrur(rzreg)  * yk
        rrbhom = (pfam(rzreg) - pfrur(rzreg))  * yk
c
c               urban apartments
c
        rhztyp(2) = rrbhom * 0.9
        rzl    =  rhzsiz(2) * 0.9
        if (rhztyp(2).gt.rzl)  rhztyp(2) = rzl
c
c               rural apartments
c
        rhztyp(4) = rhzsiz(2) - rhztyp(2)
c
c               urban single homes
c
        rhztyp(1) = rrbhom - rhztyp(2)
c
c               rural single homes
c
        rhztyp(3) = rhzsiz(1) - rhztyp(1)
```

```
        go to 800
c
200     continue
c
c                for   vienna   only urban homes
c
        do  400 i = 1,2
400     rhztyp(i) = rhzsiz(i)
c
        do  600 i = 3,4
600     rhztyp(i) = 0.
c
        rruhom    = 0.
        rrbhom = rhztyp(5)
c
800     continue
c
c
        return
        end
```

```
#
c------ r u p d a t ---------------------------------------------
c
c  function             - update of rhz..., rhq..., rhx... to get new
c                         values for the 1.jan. of the current year.
c
c  usage                - call rupdat
c
c  parameters           - none
c
c  subroutines used     - call rvsum, rvdiv, rvvdiv, rvvms,
c
c....................
c
c  note                 - it is necessary to substract each simulation
c                         year the losses of homes and to add NEW HOMES
c                         homes
c-----------------------------------------------------------------
c  author               - anton toifelhardt
c  latest revision      - 78/08/26, erwin poenitz
c
c
        subroutine rupdat
c
c
#include "model.com"
c
        integer t,b,f
c
        do  200  t = 1,4
        rhztyp(t) = rhztyp(t) + rhntyp(t)
        rhqtyp(t) = rhqtyp(t) + rhntyp(t) - rdqtyp(t)
        rhxtyp(t) = rhxtyp(t) - rdxtyp(t)
c
c               remove 'negative' homes
c
        if (rhxtyp(t).lt.0.) rhxtyp(t) = 0.
        if (rhqtyp(t).lt.0.) rhqtyp(t) = 0.
        if (rhztyp(t).lt.0.) rhztyp(t) = 0.
c
c               set  rhxtyp + rhqtyp  =  rhztyp
c
        rzl = rhxtyp(t) + rhqtyp(t)
        rzl = rhztyp(t) / amax1 (rzl,1.)
        rhxtyp(t) = rhxtyp(t) * rzl
        rhqtyp(t) = rhqtyp(t) * rzl
200     continue
c
c               add home types, type 1 + 2 + 3 + 4 = 5
c
        call rvsum (rhztyp, 4, rhztyp(5))
        call rvsum (rhqtyp, 4, rhqtyp(5))
        call rvsum (rhxtyp, 4, rhxtyp(5))
c
c               fraction of homes by type
c
```

```
      call rvdiv (rhztyp, 4, rhztyp(5), rhzfrc)
      call rvdiv (rhqtyp, 4, rhqtyp(5), rhqfrc)
      call rvdiv (rhxtyp, 4, rhxtyp(5), rhxfrc)
c
c             fraction of OLD/TOTAL NEW HOMES of type t
c             related to all homes of type t
c
      call rvvdiv (rhxtyp, rhztyp, rxxfrc, 5)
      call rvvdiv (rhqtyp, rhztyp, rqqfrc, 5)
c
c             number of homes by size
c             1,3 single family homes, 2,4 apartments
c
      rhzsiz(1) = rhztyp(1) + rhztyp(3)
      rhzsiz(2) = rhztyp(2) + rhztyp(4)
c
c             calculate number of homes according to age
c
      do 300 n = 1,rmxage
      do 300 t = 1,4
      rhytyp(n,t) = rhytyp(n,t) * (1.-rdem(n))
300   continue
c
c
      k = rzcur - rbyear + rmxage - 1
      do 430 t = 1,4
      do 430 l = k,1,-1
      rhytyp(l+1,t) = rhytyp(l,t)
      rhytyp(l,t)   = rhntyp(t)
430   continue
c
c             remove 'negative' homes
c
      do  456  t = 1,4
      do  456  l = 1, rzcur -rbyear + rmxage + 1
      if (rhytyp(l,t).lt.0.) rhytyp(l,t) = 0.
456   continue
c
      do 480 l = 1, rzcur -rbyear + rmxage + 1
      rhy(l) = 0.
      do 480 t = 1,4
      rhy(l) = rhy(l) + rhytyp(l,t)
480   continue
c
c     call rv2sum (rhytyp,1,4,rhy,2)
c
c             adjust base appliance fractions and yearly use
c             and also home-size, insulation-factor and water-use
c             q = TOTAL NEW, n = NEW (=current simulation year)
c
      do  520  t = 1,4
      rfq = rhqtyp(t) / amax1((rhqtyp(t) + rhntyp(t)),1.)
      rfn = 1. - rfq
c
      do  515  i = 1,3
      do  514  j = 1,6
```

```
      rbqtbf(j,i,t) = rbqtbf(j,i,t) * rfq + rbntbf(j,i,t) * rfn
514   continue
      do  515  j = 1,7
      rpqtbf(j,i,t) = rpqtbf(j,i,t) * rfq + rpntbf(j,i,t) * rfn
515   continue
520   continue
c
      if (iyear.eq.2) goto 665
c
665   do 666 t=1,4
      do 666 i=1,3
      do 666 j=1,7
      rpqtbf(j,i,t) = rpntbf(j,i,t)
666   continue
c
      call rvvms (rpqhms, rhqfrc, 4, rpqhms(5))
      call rvvms (rpxhms, rhxfrc, 4, rpxhms(5))
      call rvvms (rpzhms, rhzfrc, 4, rpzhms(5))
      call rvvms (rpnhms, rhnfrc, 4, rpnhms(5))
c
      do  550  t = 1,5
      rfq = rhqtyp(t) / amaxl((rhqtyp(t) + rhntyp(t)),1.)
      rfn = 1. - rfq
      rpqhms(t) = rfq * rpqhms(t) + rfn * rpnhms(t)
      rpzhms(t) = rpqhms(t) * rqqfrc(t) + rpxhms(t) * rxxfrc(t)
550   continue
c
      do  560  t = 1,4
      rfq = rhqtyp(t) / amaxl((rhqtyp(t) + rhntyp(t)),1.)
      rfn = 1. - rfq
      rpqins(t) = rfq * rpqins(t) + rfn * rpnins(t)
      rpqwat(t) = rfq * rpqwat(t) + rfn * rpnwat(t)
560   continue
c
      call rvdiv (rpqhms, 5, pfsize(rzreg), rpqlac)
      call rvdiv (rpxhms, 5, pfsize(rzreg), rpxlac)
      call rvdiv (rpnhms, 5, pfsize(rzreg), rpnlac)
      call rvdiv (rpzhms, 5, pfsize(rzreg), rpzlac)
c
      return
      end
```

```
#
c------ u m o d e l ----------------------------------------------------
c
c  function              - interface with simcon command language
c
c  usage                 - interface with simcon command language
c
c  parameters            - iyear. current simulation year
c
c  subroutines used      - call    rinit, rpread, setfil 3,4, rvzero
c                          rmodel, rsumup, reseng, rputcom, closef 3,4
c....................
c
c  note                  -
c
c-----------------------------------------------------------------------
c  author                - anton toifelhardt
c  latest revision       - 78/Ø8/26, erwin poenitz
c
c
      subroutine umodel (iyear)
c
c
#include "model.com"
c
      integer f
c
      integr(8) = Ø
c
c           read population data
c
      call rpread
c
c           do nothing in the initial year (all has been done in 'rinit')
c
c
      ---------------
      if (iyear.eq.1)  return
      ---------------
c
c
c           open help-files
c
      call setfil (3,'=write')
      call setfil (4,'=convmat')
c
c
      do  25ØØ  rzreg = 1,1Ø
c
c           calculate only if switch for this region is set to '1'
c
c
      ----------------------
      if (rregn(rzreg).eq.Ø)  go to 25ØØ
      ----------------------
c
c
c           sum up over time for some variables
c
      call rsumup
```

```
c
        call rmodel (iyear + rbyear - 1)
c
c               prepare output for file 'resengyout' (environment)
c
c       call reseng
c
c               write the whole common-block to help file '=write'
c
        call rputcom (rzreg, 3)
c
c
2500    continue
c
c               close files '=convmat', '=write'
c
        call closef (3)
        call closef (4)
c
        return
        end
c
c
c------ u i n i t -----------------------------------------------
c
c  function               - initialize the residential model for all regions
c
c  usage                  - interface with command language simcon
c
c  parameters             - none
c
c  subroutines used       - call rinit
c
c....................
c
c  note                   -
c
c-------------------------------------------------------------------
c  author                 - anton toifelhardt
c  latest revision        - 77/00/00
c
c
        subroutine uinit
c
c
        call rinit
c
c
        return
        end
```

A P P E N D I X C

Input Data and Sample Run

This appendix provides the input data needed to run REUMA and presents one sample run.

```
=====  oesterreich  ==========  for scenarios  3 and 4 ==========
=====================================================================
-(rscfrc)-    0.84      0.22      0.03      0.5       0.5       0.05      0.65
              0.06      0.75      0.99      0.        0.        0.34      0.03
-(rschyr)-    10.       18.       50.       6.        15.       16.       15.
              35.       15.       15.       15.       15.       10.       10.
-(rscsat)-    1.        0.8       0.6       0.9       0.7       0.3       0.8
              1.        1.        1.        0.7       0.5       0.3       0.17
-(rpsk)-      0.000215  0.000430  0.000645  0.000344  0.000430  0.000601  0.0000
              0.000120  0.000258  0.000130  0.000171  0.000172  0.0011    0.0007
-(rpkcal)-    120.      90.       120.      90.
-(rpqhms)-    105.      67.       105.      67.
-(rpxhms)-    83.       53.       83.       53.
-(rpnhms)-    105.      67.       105.      67.
-(rphthy)-    1500.     1500.     1300.     1300.
-(rptemp)-    1.        1.        1.        1.
-(rpqwat)-    0.00130   0.00130   0.00130   0.0013
-(rpxwat)-    0.00130   0.00130   0.00130   0.0013
-(rpnwat)-    0.00130   0.00130   0.00130   0.0013
-(rbxsa)-
s oven        0.01      0.01      0.10      0.33      0.35                      0.8     1.
i ctht                  0.01      0.08      0.09      0.01      0.01            0.2
n hotw        0.36      0.02      0.06      0.04      0.01      0.01            0.5
a oven        0.05      0.09      0.20      0.50      0.04                      0.88    1.
p ctht                            0.07      0.02                0.03            0.12
t hotw        0.46      0.1       0.05      0.01                0.03            0.65
-(rpxtbf)-
u s si        1.        1.38      1.54      0.83      1.        1.        1.
r i ch        1.        1.38      1.54      1.66      2.        1.        1.
b n hw        1.        1.42      2.58      4.        2.        1.        1.
a a si        1.        1.38      1.54      0.83      1.        1.        1.
n p ch        1.        1.33      1.42      1.66      2.        1.        1.
  m hw        1.        1.42      1.42      3.        2.        1.        1.
r s si        1.        1.38      1.54      0.83      1.        1.        1.
u i ch        1.        1.38      1.54      1.66      2.        1.        1.
r n hw        1.        1.42      2.85      4.        2.        1.        1.
a a si        1.        1.38      1.54      0.83      1.        1.        1.
l p ch        1.        1.33      1.42      1.66      2.        1.        1.
  m hw        1.        1.42      1.42      3.        2.        1.        1.
-(rbnsa)-
s oven        0.20      0.01      0.10      0.02      0.01
i ctht        0.01      0.01      0.58      0.01      0.05
n hotw        0.72      0.03      0.20                0.05
a oven        0.10      0.10
p ctht        0.01      0.19      0.40                          0.20
t hotw        0.20      0.40      0.20                          0.20
-(rpntbf)-
u s si        1.        1.35      1.55
r i ch        1.        1.35      1.55      1.45      1.5       1.        1.
b n hw        1.        1.42      2.85      4.        2.        1.        1.
a a si        1.        1.35      1.55                          1.        1.
n p ch        1.        1.35      1.65      1.45      1.5       1.        1.
  m hw        1.        1.42      2.00      3.        2.        1.        1.
r s si        1.        1.35      1.55
u i ch        1.        1.35      1.55      1.45      1.5       1.        1.
r n hw        1.        1.42      2.85      4.        2.        1.        1.
```

```
a a si         1.       1.35     1.55                        1.       1.
l p ch         1.       1.35     1.65     1.45     1.5       1.       1.
  m hw         1.       1.42     2.00     3.       2.        1.       1.
-(rhifrc)-      0.25     0.75     0.75     0.25
-(rurfrc)-
u s so         0.50     0.50     0.50     0.50     0.50     0.50     0.5
r i ch         0.50     0.50     0.50     0.50     0.50     0.50     0.5
b n hw         0.50     0.50     0.50     0.50     0.50     0.50     0.5
a a so         0.50     0.50     0.50     0.50     0.50     0.50     0.5
n p ch         0.50     0.50     0.50     0.50     0.50     0.50     0.5
  t hw         0.50     0.50     0.50     0.50     0.50     0.50     0.5
r s so         0.50     0.50     0.50     0.50     0.50     0.50     0.5
u i ch         0.50     0.50     0.50     0.50     0.50     0.50     0.5
r n hw         0.50     0.50     0.50     0.50     0.50     0.50     0.5
a a so         0.50     0.50     0.50     0.50     0.50     0.50     0.5
l p ch         0.50     0.50     0.50     0.50     0.50     0.50     0.5
  t hw         0.50     0.50     0.50     0.50     0.50     0.50     0.5
-(rhzsiz)-  1076783.  1355119.  2431902.                                    -(rhztot)-
-(rdm)-        0.02       0.0145                                               -(rdm)-
-(rat)-           30.        30.
-(ragval,ragunk)-
              523084. 592084. 374316. 513343. 494421.       0.       0.       0.
                   0.      0.      0.      0.      0.       0.       0.       0.
                   0.      0.      0. 107894.
====================         oesterreich         ====================
estimated rank of c:   2
estimated rank of e:   2
0 components excluded eps=  1.0000e-03 iter=  1
 0.9579275 0.0291847
 0.0420725 0.9708153
ier=0
estimated rank of c:   5
estimated rank of e:  20
0 components excluded eps=  1.0000e-03 iter=  4
 0.7890625 0.5078125 0.0078125 0.0077519 0.0000000
 0.1640625 0.4843750 0.0390625 0.0155039 0.0000000
 0.0468750 0.0078125 0.9140625 0.0465116 0.0000000
 0.0000000 0.0000000 0.0390625 0.9147287 0.0000000
 0.0000000 0.0000000 0.0000000 0.0155039 1.0000000
ier=0
estimated rank of c:   6
estimated rank of e:  25
0 components excluded eps=  1.0000e-03 iter=  5
 0.2547322 0.0537415 0.0000000 0.0000000 0.0680395 0.2452964
 0.3772154 0.5225677 0.0173644 0.0303116 0.0275423 0.0472190
 0.1620572 0.0432968 0.8932503 0.0951080 0.0000000 0.3600824
 0.0000000 0.1276245 0.0138167 0.8267517 0.0000000 0.0840009
 0.0980232 0.2527695 0.0000000 0.0000000 0.9044182 0.0520409
 0.1079720 0.0000000 0.0755686 0.0478287 0.0000000 0.2113603
ier=0
estimated rank of c:   2
estimated rank of e:   2
0 components excluded eps=  1.0000e-03 iter=  1
 1.0000000 0.0918338
 0.0000000 0.9081662
ier=0
```

```
estimated rank of c:   6
estimated rank of e:  25
0 components excluded eps=  4.0000e-03 iter= 13
 0.8906250 0.0156250 0.3437500 0.3984375 0.0708661 0.0625000
 0.0000000 0.4921875 0.0000000 0.0000000 0.3779528 0.3359375
 0.0703125 0.0000000 0.5781250 0.0000000 0.1732284 0.0156250
 0.0234375 0.0000000 0.0000000 0.5625000 0.0000000 0.0000000
 0.0000000 0.4921875 0.0000000 0.0000000 0.3779528 0.3359375
 0.0156250 0.0000000 0.0781250 0.0390625 0.0000000 0.2500000
ier=0
estimated rank of c:   2
estimated rank of e:   2
0 components excluded eps=  1.0000e-03 iter=  0
 0.9819465 0.0628424
 0.0180535 0.9371576
ier=0
estimated rank of c:   5
estimated rank of e:  20
0 components excluded eps=  1.0000e-03 iter=  2
 0.9501953 0.0195503 0.0000000 0.0000000 0.3544922
 0.0498047 0.8445748 0.0996094 0.0000000 0.0000000
 0.0000000 0.1309873 0.8535156 0.0498047 0.0488281
 0.0000000 0.0019550 0.0000000 0.9277344 0.2646484
 0.0000000 0.0029326 0.0468750 0.0224609 0.3320312
ier=0
estimated rank of c:   5
estimated rank of e:  16
1 components excluded eps=  1.0000e-03 iter=  5
 0.3958935 0.3959103 0.0329828 0.0097494 0.0000000 0.0130348
 0.3959860 0.3960056 0.0329418 0.0098944 0.0000000 0.0130339
 0.0442229 0.0441456 0.9321136 0.0905743 0.0000000 0.0480003
 0.0000000 0.0000505 0.0009813 0.7890081 0.0000000 0.0000000
 0.0000000 0.0000000 0.0000000 0.0000000 1.0000000 0.0000000
 0.1638977 0.1638880 0.0009804 0.1007738 0.0000000 0.9259310
ier=0
estimated rank of c:   2
estimated rank of e:   2
0 components excluded eps=  1.0000e-03 iter=  1
 1.0000000 0.0806354
 0.0000000 0.9193646
ier=0
estimated rank of c:   5
estimated rank of e:  20
1 components excluded eps=  2.0000e-03 iter=  6
 0.9375000 0.0000000 0.0588235 0.3125000 0.0000000 0.5000000
 0.0000000 0.7333333 0.5294118 0.1250000 0.0000000 0.0625000
 0.0000000 0.2666667 0.4117647 0.0625000 0.0000000 0.0000000
 0.0000000 0.0000000 0.0000000 0.5000000 0.0000000 0.0000000
 0.0000000 0.0000000 0.0000000 0.0000000 1.0000000 0.0000000
 0.0625000 0.0000000 0.0000000 0.0000000 0.0000000 0.4375000
ier=0
estimated rank of c:   2
estimated rank of e:   2
0 components excluded eps=  1.0000e-03 iter=  1
 0.9233747 0.0134685
 0.0766253 0.9865315
```

```
ier=0
estimated rank of c:   5
estimated rank of e:  20
0 components excluded eps=  1.0000e-03 iter=  4
 0.9921875 0.0390625 0.0546875 0.0310078 0.0314961
 0.0078125 0.9609375 0.0000000 0.0077519 0.5433071
 0.0000000 0.0000000 0.8906250 0.4031008 0.0000000
 0.0000000 0.0000000 0.0546875 0.5503876 0.0000000
 0.0000000 0.0000000 0.0000000 0.0077519 0.4251969
ier=0
estimated rank of c:   6
estimated rank of e:  25
0 components excluded eps=  4.0000e-03 iter= 13
 0.9125977 0.0000000 0.0000000 0.3430786 0.0000000 0.0000000
 0.0012207 0.4589844 0.0078731 0.0384521 0.0980225 0.0256958
 0.0000000 0.0297241 0.9772963 0.1746216 0.0671997 0.0000000
 0.0213623 0.0126953 0.0121453 0.1672974 0.0000000 0.0000000
 0.0648193 0.4985962 0.0026854 0.1181641 0.7953491 0.1478882
 0.0000000 0.0000000 0.0000000 0.1583862 0.0394287 0.8264160
ier=0
estimated rank of c:   6
estimated rank of e:  25
2 components excluded eps=  4.0000e-03 iter= 13
 1.0000000 0.0000000
 0.0000000 1.0000000
ier=0
estimated rank of c:   5
estimated rank of e:  20
1 components excluded eps=  1.0000e-03 iter=  2
 0.9687500 0.0852713 0.0468750 0.0000000 0.0000000 0.0000000
 0.0078125 0.5891473 0.0546875 0.0000000 0.0000000 0.0000000
 0.0156250 0.3255814 0.8984375 0.0000000 0.0390625 0.0000000
 0.0000000 0.0000000 0.0000000 1.0000000 0.0000000 0.0000000
 0.0078125 0.0000000 0.0000000 0.0000000 0.9609375 0.0156250
 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.9843750
ier=0
estimated rank of c:   2
estimated rank of e:   2
0 components excluded eps=  1.0000e-03 iter=  1
 0.8792595 0.0075240
 0.1207405 0.9924760
ier=0
estimated rank of c:   2
estimated rank of e:   2
3 components excluded eps=  1.0000e-03 iter=  1
 0.9574497 0.0857142 0.0000000 0.0000000 0.0000000
 0.0425503 0.9142858 0.0000000 0.0000000 0.0000000
 0.0000000 0.0000000 1.0000000 0.0000000 0.0000000
 0.0000000 0.0000000 0.0000000 1.0000000 0.0000000
 0.0000000 0.0000000 0.0000000 0.0000000 1.0000000
ier=0
estimated rank of c:   4
estimated rank of e:  12
2 components excluded eps=  1.0000e-03 iter=  1
 0.3537978 0.0000000 0.0033123 0.0000000 0.0000000 0.0486213
 0.1397707 0.8081746 0.0363439 0.0000000 0.0000000 0.0884279
```

```
 0.3062337 0.0077052 0.9603437 0.0000000 0.0000000 0.0000000
 0.0000000 0.0000000 0.0000000 1.0000000 0.0000000 0.0000000
 0.0000000 0.0000000 0.0000000 0.0000000 1.0000000 0.0000000
 0.2001978 0.1841202 0.0000000 0.0000000 0.0000000 0.8629508
ier=0
estimated rank of c:    4
estimated rank of e:   12
2 components excluded eps=  1.0000e-03 iter=  1
 1.0000000 0.0000000
 0.0000000 1.0000000
ier=0
estimated rank of c:    4
estimated rank of e:   12
2 components excluded eps=  1.0000e-03 iter=  1
 0.4692875 0.1400586 0.2092186 0.0000000 0.0000000 0.0000000
 0.1546613 0.8599414 0.1048943 0.0000000 0.0000000 0.0000000
 0.3760512 0.0000000 0.5440139 0.0000000 0.0000000 0.0693042
 0.0000000 0.0000000 0.0000000 1.0000000 0.0000000 0.0000000
 0.0000000 0.0000000 0.0000000 0.0000000 1.0000000 0.0000000
 0.0000000 0.0000000 0.1418732 0.0000000 0.0000000 0.9306958
=====  burgenland  ==========================================
============================================================
-(rscfrc)-    0.72     0.4      0.01     0.58     0.56     0.06     0.63
              0.03     0.75     0.99     0.01     0.       0.1      0.01
-(rschyr)-   15.      15.      20.      13.      15.      13.      15.
             15.      15.      15.      15.      15.       5.       5.
-(rscsat)-    1.       0.9      0.9      1.       0.85     0.1      0.2
              1.       1.       1.       1.       0.       0.15     0.07
-(rpsk)-     0.000258  0.000515  0.000516  0.000344  0.000860  0.000602  0.00006
             0.000120  0.000602  0.000170  0.000171  0.000172  0.0011    0.0007
-(rpkcal)-   120.      80.      120.      80.
-(rpqhms)-   101.      70.      101.      70.
-(rpxhms)-    80.      51.       80.      51.
-(rpnhms)-   101.      70.      101.      70.
-(rphthy)-  1500.    1500.     1300.    1300.
-(rptemp)-     0.93     0.93      0.93     0.93
-(rpqwat)-  0.00130  0.00130  0.00130  0.0013
-(rpxwat)-  0.00130  0.00130  0.00130  0.0013
-(rpnwat)-  0.00130  0.00130  0.00130  0.0013
-(rbxsa)-
s oven        0.03              0.08     0.28     0.5               0.89    1.
i ctht                 0.01     0.06     0.03     0.01              0.11
n hotw        0.31     0.01     0.03     0.02     0.01              0.38
a oven        0.05              0.18     0.42     0.12              0.77    1.
p ctht                 0.03     0.12     0.02           0.06        0.23
t hotw        0.4      0.03     0.09     0.02           0.05        0.59
-(rpxtbf)-
p s si        1.       1.38     1.54     0.83     1.     1.     1.
r i ch        1.       1.38     1.54     1.66     2.     1.     1.
b n hw        1.       1.42     2.58     4.       2.     1.     1.
a a si        1.       1.38     1.54     0.83     1.     1.     1.
n p ch        1.       1.33     1.42     1.66     2.     1.     1.
  m hw        1.       1.42     1.42     3.       2.     1.     1.
r s si        1.       1.38     1.54     0.83     1.     1.     1.
u i ch        1.       1.38     1.54     1.66     2.     1.     1.
r n hw        1.       1.42     2.85     4.       2.     1.     1.
```

```
           Region:
           1        2        3        4        5        6        7        8        9        10
           Austria Burgenl.Kntn.    N.Oe.    O.Oe.    Salzbg. Stmk.    Tirol    Vlbg.    Wien

year 1

pop        7456403. 272119. 525728.1414161.1223444. 401766.1192100. 540771. 271473.1614841.
prural     2883460. 175742. 223194. 787301. 565082. 170818. 558014. 288523. 114786.      0.
pfam       2538723.  78075. 154718. 466740. 381284. 125544. 362488. 157575.  79149. 733150.
pfrur       823751.  50165.  56534. 248700. 158227.  46962. 153590.  76352.  33220.      0.
pifam         7694.    -15.    634.    289.   2087.   1358.   1147.   1568.   1036.  -1874.
pifrur        3329.     -9.    233.     11.    866.    492.    497.    833.    404.      0.
pfsize    2.9370683.4853543.3979763.0298693.2087473.2002013.2886613.4318333.4298982.202607

year 2     7456403. 272119. 525728.1414161.1223444. 401766.1192100. 540771. 271473.1614841.
  .        2883460. 175742. 223194. 787301. 565082. 170818. 558014. 288523. 114786.      0.
  .        2538723.  78075. 154718. 466740. 381284. 125544. 362488. 157575.  79149. 733150.
  .         823751.  50165.  56534. 248700. 158227.  46962. 153590.  76352.  33220.      0.
  .           7694.    -15.    634.    289.   2087.   1358.   1147.   1568.   1036.  -1874.
              3329.     -9.    233.     11.    866.    492.    497.    833.    404.      0.
          2.9370683.4853543.3979763.0298693.2087473.2002013.2886613.4318333.4298982.202607
           7457196. 271262. 526342.1410886.1226595. 404958.1192361. 544609. 274258.1605923.
           2876085. 174881. 222477. 783083. 564646. 170818. 556872. 288523. 114786.      0.
           2546633.  78063. 155365. 467060. 383416. 126922. 363658. 159171.  80202. 731295.
            824004.  50077.  56512. 248097. 158598.  47104. 153713.  76582.  33320.      0.
              7910.    -12.    647.    320.   2132.   1378.   1170.   1596.   1053.  -1855.
               253.    -88.    -22.   -603.    372.    141.    123.    230.    100.      0.
          2.9282573.4748983.3877823.0207793.1991213.1906003.2787953.4215373.4196082.195999
           7459093. 270434. 527037.1407833.1229991. 408216.1192774. 548567. 277098.1597144.
           2869014. 174045. 221796. 778995. 564252. 170818. 555799. 288523. 114786.      0.
           2554946.  78059. 156038. 467452. 385635. 128328. 364879. 160810.  81276. 729486.
            824337.  49995.  56499. 247530. 158981.  47245. 153854.  76812.  33421.      0.
              8313.     -4.    673.    392.   2218.   1406.   1221.   1639.   1074.  -1809.
               333.    -82.    -13.   -568.    383.    142.    141.    230.    100.      0.
          2.9194723.4644733.3776183.0117173.1895233.1810283.2689583.4112723.4093492.189411
           7462105. 269634. 527810.1404999.1233633. 411539.1193340. 552647. 279995.1588506.
           2862229. 173234. 221145. 775035. 563893. 170818. 554796. 288523. 114786.      0.
           2563668.  78062. 156737. 467915. 387940. 129762. 366151. 162494.  82373. 727724.
            824746.  49919.  56493. 246996. 159373.  47387. 154012.  77044.  33521.      0.
              8723.      3.    699.    463.   2306.   1434.   1272.   1683.   1097.  -1762.
               409.    -76.     -6.   -534.    392.    142.    158.    231.    101.      0.
          2.9107143.4540793.3674853.0026823.1799553.1714853.2591513.4010383.3991212.182843
           7466219. 268861. 528662.1402382.1237524. 414930.1194059. 556850. 282949.1580001.
           2855721. 172448. 220519. 771198. 563567. 170818. 553861. 288523. 114786.      0.
           2572800.  78073. 157463. 468449. 390335. 131225. 367473. 164222.  83492. 726005.
            825228.  49849.  56494. 246496. 159774.  47530. 154188.  77275.  33622.      0.
              9132.     10.    725.    534.   2394.   1463.   1323.   1729.   1120.  -1718.
               482.    -70.      0.   -500.    401.    143.    176.    232.    101.      0.
          2.9019823.4437173.3573832.9936743.1704153.1619713.2493743.3908353.3889242.176294
           7471447. 268115. 529595.1399982.1241664. 418388.1194928. 561181. 285962.1571632.
           2849469. 171686. 219919. 767484. 563276. 170818. 552976. 288523. 114786.      0.
           2582349.  78091. 158215. 469054. 392819. 132717. 368847. 165997.  84635. 724333.
            825780.  49785.  56501. 246028. 160184.  47673. 154378.  77508.  33723.      0.
              9549.     18.    752.    606.   2484.   1492.   1374.   1775.   1143.  -1673.
               552.    -64.      7.   -467.    410.    143.    190.    233.    101.      0.
          2.8932753.4333863.3473112.9846933.1609043.1524853.2396253.3806633.3787572.169765
```

```
7477794. 267397. 530608.1397796.1246058. 421917.1195950. 565639. 289035.1563394.
2843464. 170949. 219344. 763883. 563018. 170818. 552143. 288523. 114786.      0.
2592320.  78116. 158995. 469731. 395396. 134239. 370274. 167820.  85802. 722704.
 826398.  49727.  56514. 245591. 160603.  47816. 154581.  77741.  33825.      0.
   9971.     25.    780.    677.   2577.   1522.   1427.   1822.   1167.  -1629.
    618.    -58.     13.   -437.    419.    143.    203.    233.    101.      0.
2.8845963.4230863.3372692.9757393.1514213.1430273.2299073.3705213.3686212.163256
7485246. 266704. 531699.1395821.1250704. 425515.1197123. 570227. 292169.1555284.
2837700. 170235. 218794. 760390. 562794. 170818. 551360. 288523. 114786.      0.
2602711.  78148. 159801. 470478. 398064. 135791. 371752. 169690.  86993. 721118.
 827082.  49675.  56533. 245183. 161031.  47960. 154798.  77975.  33927.      0.
  10392.     32.    807.    748.   2668.   1552.   1478.   1870.   1191.  -1586.
    684.    -52.     19.   -408.    428.    144.    217.    234.    102.      0.
2.8759423.4128163.3272572.9668113.1419673.1335983.2202173.3604093.3585152.156766
7493825. 266038. 532872.1394057.1255609. 429185.1198448. 574949. 295364.1547301.
2832164. 169545. 218268. 757006. 562602. 170818. 550615. 288523. 114786.      0.
2613535.  78187. 160636. 471298. 400828. 137374. 373284. 171610.  88210. 719576.
 827829.  49628.  56558. 244805. 161468.  48105. 155026.  78210.  34029.      0.
  10824.     39.    834.    820.   2763.   1583.   1531.   1920.   1216.  -1542.
    747.    -47.     25.   -378.    437.    144.    228.    235.    102.      0.
2.8673143.4025783.3172752.9579113.1325413.1241973.2105563.3503283.3484392.150296
```

```
demr     (  1) =     8467.239
demr     (  2) =     5442.033
demr     (  3) =     13964.45
demr     (  4) =     5416.329
demr     (  5) =     6214.998
demr     (  6) =     1373.936
demr     (  7) =    0.0000000
pfam     (  1) =     2624794.
pfrur    (  1) =     828637.0
pfsize   (  1) =     2.858712
pifam    (  1) =     11259.00
pifrur   (  1) =     808.2424
pop      (  1) =     7503530.
prural   (  1) =     2826850.
racel    (  1) =    0.0000000
racel    (  2) =    0.1000000
racel    (  3) =    0.1000000
rafb     (  1) =    0.0000000
rafb     (  2) =    0.0000000
rafb     (  3) =    0.0000000
rafb     (  4) =    0.0000000
rafbn    (  1) =    0.0000000
rafbn    (  2) =    0.0000000
rafbn    (  3) =    0.0000000
rafbn    (  4) =    0.0000000
rafbq    (  1) =    0.0000000
rafbq    (  2) =    0.0000000
rafbq    (  3) =    0.0000000
rafbq    (  4) =    0.0000000
rafbx    (  1) =    0.0000000
rafbx    (  2) =    0.0000000
rafbx    (  3) =    0.0000000
rafbx    (  4) =    0.0000000
rafn     (  1,  1,  1) =   0.0000000
rafn     (  1,  1,  2) =   0.0000000
rafn     (  1,  1,  3) =   0.0000000
rafn     (  1,  2,  1) =   0.0000000
rafn     (  1,  2,  2) =   0.0000000
rafn     (  1,  2,  3) =   0.0000000
rafn     (  2,  1,  1) =   0.0000000
rafn     (  2,  1,  2) =   0.0000000
rafn     (  2,  1,  3) =   0.0000000
rafn     (  2,  2,  1) =   0.0000000
rafn     (  2,  2,  2) =   0.0000000
rafn     (  2,  2,  3) =   0.0000000
rafn     (  3,  1,  1) =   0.0000000
rafn     (  3,  1,  2) =   0.0000000
rafn     (  3,  1,  3) =   0.0000000
rafn     (  3,  2,  1) =   0.0000000
rafn     (  3,  2,  2) =   0.0000000
rafn     (  3,  2,  3) =   0.0000000
rafq     (  1,  1,  1) =   0.0000000
rafq     (  1,  1,  2) =   0.0000000
rafq     (  1,  1,  3) =   0.0000000
rafq     (  1,  2,  1) =   0.0000000
rafq     (  1,  2,  2) =   0.0000000

rafq     (  1,  2,  3) =   0.0000000
rafq     (  2,  1,  1) =   0.0000000
rafq     (  2,  1,  2) =   0.0000000
rafq     (  2,  1,  3) =   0.0000000
rafq     (  2,  2,  1) =   0.0000000
rafq     (  2,  2,  2) =   0.0000000
rafq     (  2,  2,  3) =   0.0000000
rafq     (  3,  1,  1) =   0.0000000
rafq     (  3,  1,  2) =   0.0000000
rafq     (  3,  1,  3) =   0.0000000
rafq     (  3,  2,  1) =   0.0000000
rafq     (  3,  2,  2) =   0.0000000
rafq     (  3,  2,  3) =   0.0000000
raft     (  1) =    0.0000000
raft     (  2) =    0.0000000
raft     (  3) =    0.0000000
raftn    (  1) =    0.0000000
raftn    (  2) =    0.0000000
raftn    (  3) =    0.0000000
raftq    (  1) =    0.0000000
raftq    (  2) =    0.0000000
raftq    (  3) =    0.0000000
raftx    (  1) =    0.0000000
raftx    (  2) =    0.0000000
raftx    (  3) =    0.0000000
rafx     (  1,  1,  1) =   0.0000000
rafx     (  1,  1,  2) =   0.0000000
rafx     (  1,  1,  3) =   0.0000000
rafx     (  1,  2,  1) =   0.0000000
rafx     (  1,  2,  2) =   0.0000000
rafx     (  1,  2,  3) =   0.0000000
rafx     (  2,  1,  1) =   0.0000000
rafx     (  2,  1,  2) =   0.0000000
rafx     (  2,  1,  3) =   0.0000000
rafx     (  2,  2,  1) =   0.0000000
rafx     (  2,  2,  2) =   0.0000000
rafx     (  2,  2,  3) =   0.0000000
rafx     (  3,  1,  1) =   0.0000000
rafx     (  3,  1,  2) =   0.0000000
rafx     (  3,  1,  3) =   0.0000000
rafx     (  3,  2,  1) =   0.0000000
rafx     (  3,  2,  2) =   0.0000000
rafx     (  3,  2,  3) =   0.0000000
ragunk     =   107894.0
ragval   (  1) =     523084.0
ragval   (  2) =     592084.0
ragval   (  3) =     374316.0
ragval   (  4) =     513343.0
ragval   (  5) =     494421.0
ragval   (  6) =     47418.00
ragval   (  7) =     49856.00
ragval   (  8) =     58042.00
ragval   (  9) =     60581.00
ragval   ( 10) =     57955.00
ragval   ( 11) =     58137.00
ragval   ( 12) =     60931.00

ragval   ( 13) =     53966.00
ragval   ( 14) =     49921.00
ragval   ( 15) =     44159.00
ragval   ( 16) =     50373.00
ragval   ( 17) =     44193.00
ragval   ( 18) =     50131.00
ragval   ( 19) =     48570.00
rapl     (  1) =    0.0000000
rapl     (  2) =     1.000000
rapl     (  3) =     1.000000
rarepl   (  1) =    0.0000000
rarepl   (  2) =    0.5000000
rarepl   (  3) =    0.7000000
rat      (  1) =     30.00000
rat      (  2) =     30.00000
rbnf     (  1) =    0.1248232
rbnf     (  2) =    0.1730647
rbnf     (  3) =    0.4776453
rbnf     (  4) =    0.6098042e-02
rbnf     (  5) =    0.3083684e-01
rbnf     (  6) =    0.1875320
rbnf     (  7) =    0.0000000
rbnsa    (  1,  1,  1) =   0.1617099
rbnsa    (  1,  1,  2) =   0.6177928e-01
rbnsa    (  1,  2,  1) =   0.2786588e-01
rbnsa    (  1,  2,  2) =   0.2322037e-01
rbnsa    (  1,  3,  1) =   0.6475947
rbnsa    (  1,  3,  2) =   0.1695211
rbnsa    (  2,  1,  1) =   0.1949779e-01
rbnsa    (  2,  1,  2) =   0.3796997e-01
rbnsa    (  2,  2,  1) =   0.2414864e-01
rbnsa    (  2,  2,  2) =   0.2146883
rbnsa    (  2,  3,  1) =   0.4249512e-01
rbnsa    (  2,  3,  2) =   0.3550985
rbnsa    (  3,  1,  1) =   0.4360963e-01
rbnsa    (  3,  1,  2) =   0.0000000
rbnsa    (  3,  2,  1) =   0.6057138
rbnsa    (  3,  2,  2) =   0.3720613
rbnsa    (  3,  3,  1) =   0.2338777
rbnsa    (  3,  3,  2) =   0.1850019
rbnsa    (  4,  1,  1) =   0.6031197e-02
rbnsa    (  4,  1,  2) =   0.0000000
rbnsa    (  4,  2,  1) =   0.9982178e-02
rbnsa    (  4,  2,  2) =   0.0000000
rbnsa    (  4,  3,  1) =   0.0000000
rbnsa    (  4,  3,  2) =   0.0000000
rbnsa    (  5,  1,  1) =   0.9318287e-04
rbnsa    (  5,  1,  2) =   0.0000000
rbnsa    (  5,  2,  1) =   0.8088396e-01
rbnsa    (  5,  2,  2) =   0.0000000
rbnsa    (  5,  3,  1) =   0.7603247e-01
rbnsa    (  5,  3,  2) =   0.0000000
rbnsa    (  6,  1,  1) =   0.0000000
rbnsa    (  6,  1,  2) =   0.0000000
rbnsa    (  6,  2,  1) =   0.2046373e-01
rbnsa    (  6,  2,  2) =   0.2902808
```

```
rbnsa   ( 6, 3, 1) = 0.0000000
rbnsa   ( 6, 3, 2) = 0.2903785
rbnsa   ( 7, 1, 1) = 0.0000000
rbnsa   ( 7, 1, 2) = 0.0000000
rbnsa   ( 7, 2, 1) = 0.0000000
rbnsa   ( 7, 2, 2) = 0.0000000
rbnsa   ( 7, 3, 1) = 0.0000000
rbnsa   ( 7, 3, 2) = 0.0000000
rbntbf  ( 1, 1, 1) = 0.1617099
rbntbf  ( 1, 1, 2) = 0.6177928e-01
rbntbf  ( 1, 1, 3) = 0.1617099
rbntbf  ( 1, 1, 4) = 0.6177928e-01
rbntbf  ( 1, 2, 1) = 0.2786588e-01
rbntbf  ( 1, 2, 2) = 0.2322037e-01
rbntbf  ( 1, 2, 3) = 0.2786588e-01
rbntbf  ( 1, 2, 4) = 0.2322037e-01
rbntbf  ( 1, 3, 1) = 0.6475947
rbntbf  ( 1, 3, 2) = 0.1695211
rbntbf  ( 1, 3, 3) = 0.6475947
rbntbf  ( 1, 3, 4) = 0.1695211
rbntbf  ( 2, 1, 1) = 0.1949779e-01
rbntbf  ( 2, 1, 2) = 0.3796997e-01
rbntbf  ( 2, 1, 3) = 0.1949779e-01
rbntbf  ( 2, 1, 4) = 0.3796997e-01
rbntbf  ( 2, 2, 1) = 0.2414864e-01
rbntbf  ( 2, 2, 2) = 0.2146883
rbntbf  ( 2, 2, 3) = 0.2414864e-01
rbntbf  ( 2, 2, 4) = 0.2146883
rbntbf  ( 2, 3, 1) = 0.4249512e-01
rbntbf  ( 2, 3, 2) = 0.3550985
rbntbf  ( 2, 3, 3) = 0.4249512e-01
rbntbf  ( 2, 3, 4) = 0.3550985
rbntbf  ( 3, 1, 1) = 0.4360963e-01
rbntbf  ( 3, 1, 2) = 0.0000000
rbntbf  ( 3, 1, 3) = 0.4360963e-01
rbntbf  ( 3, 1, 4) = 0.0000000
rbntbf  ( 3, 2, 1) = 0.6057138
rbntbf  ( 3, 2, 2) = 0.3720613
rbntbf  ( 3, 2, 3) = 0.6057138
rbntbf  ( 3, 2, 4) = 0.3720613
rbntbf  ( 3, 3, 1) = 0.2338777
rbntbf  ( 3, 3, 2) = 0.1850019
rbntbf  ( 3, 3, 3) = 0.2338777
rbntbf  ( 3, 3, 4) = 0.1850019
rbntbf  ( 4, 1, 1) = 0.6031197e-02
rbntbf  ( 4, 1, 2) = 0.0000000
rbntbf  ( 4, 1, 3) = 0.6031197e-02
rbntbf  ( 4, 1, 4) = 0.0000000
rbntbf  ( 4, 2, 1) = 0.9982178e-02
rbntbf  ( 4, 2, 2) = 0.0000000
rbntbf  ( 4, 2, 3) = 0.9982178e-02
rbntbf  ( 4, 2, 4) = 0.0000000
rbntbf  ( 4, 3, 1) = 0.0000000
rbntbf  ( 4, 3, 2) = 0.0000000
rbntbf  ( 4, 3, 3) = 0.0000000
rbntbf  ( 4, 3, 4) = 0.0000000
rbntbf  ( 5, 1, 1) = 0.9318287e-04
rbntbf  ( 5, 1, 2) = 0.0000000
rbntbf  ( 5, 1, 3) = 0.9318287e-04
rbntbf  ( 5, 1, 4) = 0.0000000
rbntbf  ( 5, 2, 1) = 0.8088396e-01
rbntbf  ( 5, 2, 2) = 0.0000000
rbntbf  ( 5, 2, 3) = 0.8088396e-01
rbntbf  ( 5, 2, 4) = 0.0000000
rbntbf  ( 5, 3, 1) = 0.7603247e-01
rbntbf  ( 5, 3, 2) = 0.0000000
rbntbf  ( 5, 3, 3) = 0.7603247e-01
rbntbf  ( 5, 3, 4) = 0.0000000
rbntbf  ( 6, 1, 1) = 0.0000000
rbntbf  ( 6, 1, 2) = 0.0000000
rbntbf  ( 6, 1, 3) = 0.0000000
rbntbf  ( 6, 1, 4) = 0.0000000
rbntbf  ( 6, 2, 1) = 0.2046373e-01
rbntbf  ( 6, 2, 2) = 0.2902808
rbntbf  ( 6, 2, 3) = 0.2046373e-01
rbntbf  ( 6, 2, 4) = 0.2902808
rbntbf  ( 6, 3, 1) = 0.0000000
rbntbf  ( 6, 3, 2) = 0.2903785
rbntbf  ( 6, 3, 3) = 0.0000000
rbntbf  ( 6, 3, 4) = 0.2903785
rbntbf  ( 7, 1, 1) = 0.0000000
rbntbf  ( 7, 1, 2) = 0.0000000
rbntbf  ( 7, 1, 3) = 0.0000000
rbntbf  ( 7, 1, 4) = 0.0000000
rbntbf  ( 7, 2, 1) = 0.0000000
rbntbf  ( 7, 2, 2) = 0.0000000
rbntbf  ( 7, 2, 3) = 0.0000000
rbntbf  ( 7, 2, 4) = 0.0000000
rbntbf  ( 7, 3, 1) = 0.0000000
rbntbf  ( 7, 3, 2) = 0.0000000
rbntbf  ( 7, 3, 3) = 0.0000000
rbntbf  ( 7, 3, 4) = 0.0000000
rbqf    ( 1) = 0.1407935
rbqf    ( 2) = 0.1754743
rbqf    ( 3) = 0.5006768
rbqf    ( 4) = 0.8721532e-02
rbqf    ( 5) = 0.2461836e-01
rbqf    ( 6) = 0.1497155
rbqf    ( 7) = 0.0000000
rbqtbf  ( 1, 1, 1) = 0.1835202
rbqtbf  ( 1, 1, 2) = 0.8307413e-01
rbqtbf  ( 1, 1, 3) = 0.1847528
rbqtbf  ( 1, 1, 4) = 0.8506247e-01
rbqtbf  ( 1, 2, 1) = 0.1808440e-01
rbqtbf  ( 1, 2, 2) = 0.1776971e-01
rbqtbf  ( 1, 2, 3) = 0.1748694e-01
rbqtbf  ( 1, 2, 4) = 0.1694650e-01
rbqtbf  ( 1, 3, 1) = 0.6879914
rbqtbf  ( 1, 3, 2) = 0.1839200
rbqtbf  ( 1, 3, 3) = 0.6903700
rbqtbf  ( 1, 3, 4) = 0.1856905
rbqtbf  ( 2, 1, 1) = 0.1709863e-01
rbqtbf  ( 2, 1, 2) = 0.6812295e-01
rbqtbf  ( 2, 1, 3) = 0.1663799e-01
rbqtbf  ( 2, 1, 4) = 0.7171796e-01
rbqtbf  ( 2, 2, 1) = 0.1774844e-01
rbqtbf  ( 2, 2, 2) = 0.1991726
rbqtbf  ( 2, 2, 3) = 0.1721152e-01
rbqtbf  ( 2, 2, 4) = 0.1980360
rbqtbf  ( 2, 3, 1) = 0.3735735e-01
rbqtbf  ( 2, 3, 2) = 0.3820413
rbqtbf  ( 2, 3, 3) = 0.3684848e-01
rbqtbf  ( 2, 3, 4) = 0.3842114
rbqtbf  ( 3, 1, 1) = 0.7451084e-01
rbqtbf  ( 3, 1, 2) = 0.0000000
rbqtbf  ( 3, 1, 3) = 0.7640880e-01
rbqtbf  ( 3, 1, 4) = 0.0000000
rbqtbf  ( 3, 2, 1) = 0.5951478
rbqtbf  ( 3, 2, 2) = 0.3901387
rbqtbf  ( 3, 2, 3) = 0.5940778
rbqtbf  ( 3, 2, 4) = 0.3913431
rbqtbf  ( 3, 3, 1) = 0.2132628
rbqtbf  ( 3, 3, 2) = 0.1922130
rbqtbf  ( 3, 3, 3) = 0.2122414
rbqtbf  ( 3, 3, 4) = 0.1931076
rbqtbf  ( 4, 1, 1) = 0.1210228e-01
rbqtbf  ( 4, 1, 2) = 0.0000000
rbqtbf  ( 4, 1, 3) = 0.1265354e-01
rbqtbf  ( 4, 1, 4) = 0.0000000
rbqtbf  ( 4, 2, 1) = 0.9707147e-02
rbqtbf  ( 4, 2, 2) = 0.0000000
rbqtbf  ( 4, 2, 3) = 0.9720331e-02
rbqtbf  ( 4, 2, 4) = 0.0000000
rbqtbf  ( 4, 3, 1) = 0.0000000
rbqtbf  ( 4, 3, 2) = 0.0000000
rbqtbf  ( 4, 3, 3) = 0.0000000
rbqtbf  ( 4, 3, 4) = 0.0000000
rbqtbf  ( 5, 1, 1) = 0.2824232e-02
rbqtbf  ( 5, 1, 2) = 0.0000000
rbqtbf  ( 5, 1, 3) = 0.3293034e-02
rbqtbf  ( 5, 1, 4) = 0.0000000
rbqtbf  ( 5, 2, 1) = 0.5972606e-01
rbqtbf  ( 5, 2, 2) = 0.0000000
rbqtbf  ( 5, 2, 3) = 0.5892567e-01
rbqtbf  ( 5, 2, 4) = 0.0000000
rbqtbf  ( 5, 3, 1) = 0.6138852e-01
rbqtbf  ( 5, 3, 2) = 0.0000000
rbqtbf  ( 5, 3, 3) = 0.6054016e-01
rbqtbf  ( 5, 3, 4) = 0.0000000
rbqtbf  ( 6, 1, 1) = 0.0000000
rbqtbf  ( 6, 1, 2) = 0.0000000
rbqtbf  ( 6, 1, 3) = 0.0000000
rbqtbf  ( 6, 1, 4) = 0.0000000
rbqtbf  ( 6, 2, 1) = 0.9529986e-02
rbqtbf  ( 6, 2, 2) = 0.2417220
rbqtbf  ( 6, 2, 3) = 0.8831602e-02
rbqtbf  ( 6, 2, 4) = 0.2368940
rbqtbf  ( 6, 3, 1) = 0.0000000
```

```
rbqtbf ( 6, 3, 2) = 0.2418258         rbxsa  ( 6, 3, 1) = 0.2390648e-01   rbxtbf ( 5, 1, 1) = 0.2938117
rbqtbf ( 6, 3, 3) = 0.0000000         rbxsa  ( 6, 3, 2) = 0.6378736e-01   rbxtbf ( 5, 1, 2) = 0.2711164e-01
rbqtbf ( 6, 3, 4) = 0.2369905         rbxsa  ( 7, 1, 1) = 0.0000000       rbxtbf ( 5, 1, 3) = 0.2938117
rbqtbf ( 7, 1, 1) = 0.0000000         rbxsa  ( 7, 1, 2) = 0.0000000       rbxtbf ( 5, 1, 4) = 0.2711164e-01
rbqtbf ( 7, 1, 2) = 0.0000000         rbxsa  ( 7, 2, 1) = 0.0000000       rbxtbf ( 5, 2, 1) = 0.6379201e-01
rbqtbf ( 7, 1, 3) = 0.0000000         rbxsa  ( 7, 2, 2) = 0.0000000       rbxtbf ( 5, 2, 2) = 0.0000000
rbqtbf ( 7, 1, 4) = 0.0000000         rbxsa  ( 7, 3, 1) = 0.0000000       rbxtbf ( 5, 2, 3) = 0.6379201e-01
rbqtbf ( 7, 2, 1) = 0.0000000         rbxsa  ( 7, 3, 2) = 0.0000000       rbxtbf ( 5, 2, 4) = 0.0000000
rbqtbf ( 7, 2, 2) = 0.0000000         rbxtbf ( 1, 1, 1) = 0.5377958e-01   rbxtbf ( 5, 3, 1) = 0.5053163e-01
rbqtbf ( 7, 2, 3) = 0.0000000         rbxtbf ( 1, 1, 2) = 0.1244074       rbxtbf ( 5, 3, 2) = 0.0000000
rbqtbf ( 7, 2, 4) = 0.0000000         rbxtbf ( 1, 1, 3) = 0.5377958e-01   rbxtbf ( 5, 3, 3) = 0.5053163e-01
rbqtbf ( 7, 3, 1) = 0.0000000         rbxtbf ( 1, 1, 4) = 0.1244074       rbxtbf ( 5, 3, 4) = 0.0000000
rbqtbf ( 7, 3, 2) = 0.0000000         rbxtbf ( 1, 2, 1) = 0.1554613e-01   rbxtbf ( 6, 1, 1) = 0.0000000
rbqtbf ( 7, 3, 3) = 0.0000000         rbxtbf ( 1, 2, 2) = 0.1572948e-01   rbxtbf ( 6, 1, 2) = 0.0000000
rbqtbf ( 7, 3, 4) = 0.0000000         rbxtbf ( 1, 2, 3) = 0.1554613e-01   rbxtbf ( 6, 1, 3) = 0.0000000
rbxf   ( 1) = 0.1087804               rbxtbf ( 1, 2, 4) = 0.1572948e-01   rbxtbf ( 6, 1, 4) = 0.0000000
rbxf   ( 2) = 0.1172236               rbxtbf ( 1, 3, 1) = 0.5271171       rbxtbf ( 6, 2, 1) = 0.2460780e-01
rbxf   ( 3) = 0.3023763               rbxtbf ( 1, 3, 2) = 0.5735959       rbxtbf ( 6, 2, 2) = 0.5507410e-01
rbxf   ( 4) = 0.2565772               rbxtbf ( 1, 3, 3) = 0.5271171       rbxtbf ( 6, 2, 3) = 0.2460780e-01
rbxf   ( 5) = 0.1734594               rbxtbf ( 1, 3, 4) = 0.5735959       rbxtbf ( 6, 2, 4) = 0.5507410e-01
rbxf   ( 6) = 0.4158308e-01           rbxtbf ( 2, 1, 1) = 0.2642058e-01   rbxtbf ( 6, 3, 1) = 0.2390648e-01
rbxf   ( 7) = 0.0000000               rbxtbf ( 2, 1, 2) = 0.1505896       rbxtbf ( 6, 3, 2) = 0.6378736e-01
rbxsa  ( 1, 1, 1) = 0.5377958e-01     rbxtbf ( 2, 1, 3) = 0.2642058e-01   rbxtbf ( 6, 3, 3) = 0.2390648e-01
rbxsa  ( 1, 1, 2) = 0.1244074         rbxtbf ( 2, 1, 4) = 0.1505896       rbxtbf ( 6, 3, 4) = 0.6378736e-01
rbxsa  ( 1, 2, 1) = 0.1554613e-01     rbxtbf ( 2, 2, 1) = 0.2902794e-01   rbxtbf ( 7, 1, 1) = 0.0000000
rbxsa  ( 1, 2, 2) = 0.1572948e-01     rbxtbf ( 2, 2, 2) = 0.1572947e-01   rbxtbf ( 7, 1, 2) = 0.0000000
rbxsa  ( 1, 3, 1) = 0.5271171         rbxtbf ( 2, 2, 3) = 0.2902794e-01   rbxtbf ( 7, 1, 3) = 0.0000000
rbxsa  ( 1, 3, 2) = 0.5735959         rbxtbf ( 2, 2, 4) = 0.1572947e-01   rbxtbf ( 7, 1, 4) = 0.0000000
rbxsa  ( 2, 1, 1) = 0.2642058e-01     rbxtbf ( 2, 3, 1) = 0.5053163e-01   rbxtbf ( 7, 2, 1) = 0.0000000
rbxsa  ( 2, 1, 2) = 0.1505896         rbxtbf ( 2, 3, 2) = 0.1366187       rbxtbf ( 7, 2, 2) = 0.0000000
rbxsa  ( 2, 2, 1) = 0.2902794e-01     rbxtbf ( 2, 3, 3) = 0.5053163e-01   rbxtbf ( 7, 2, 3) = 0.0000000
rbxsa  ( 2, 2, 2) = 0.1572947e-01     rbxtbf ( 2, 3, 4) = 0.1366187       rbxtbf ( 7, 2, 4) = 0.0000000
rbxsa  ( 2, 3, 1) = 0.5053163e-01     rbxtbf ( 3, 1, 1) = 0.1000588       rbxtbf ( 7, 3, 1) = 0.0000000
rbxsa  ( 2, 3, 2) = 0.1366187         rbxtbf ( 3, 1, 2) = 0.2297434       rbxtbf ( 7, 3, 2) = 0.0000000
rbxsa  ( 3, 1, 1) = 0.1000588         rbxtbf ( 3, 1, 3) = 0.1000588       rbxtbf ( 7, 3, 3) = 0.0000000
rbxsa  ( 3, 1, 2) = 0.2297434         rbxtbf ( 3, 1, 4) = 0.2297434       rbxtbf ( 7, 3, 4) = 0.0000000
rbxsa  ( 3, 2, 1) = 0.1873595         rbxtbf ( 3, 2, 1) = 0.1873595       rbyear = 1971
rbxsa  ( 3, 2, 2) = 0.8452062e-01     rbxtbf ( 3, 2, 2) = 0.8452063e-01   rbzf   ( 1) = 0.1119959
rbxsa  ( 3, 3, 1) = 0.1088936         rbxtbf ( 3, 2, 3) = 0.1873595       rbzf   ( 2) = 0.1230714
rbxsa  ( 3, 3, 2) = 0.6174089e-01     rbxtbf ( 3, 2, 4) = 0.8452063e-01   rbzf   ( 3) = 0.3222888
rbxsa  ( 4, 1, 1) = 0.1362240         rbxtbf ( 3, 3, 1) = 0.1088936       rbzf   ( 4) = 0.2316923
rbxsa  ( 4, 1, 2) = 0.2932658         rbxtbf ( 3, 3, 2) = 0.6174090e-01   rbzf   ( 5) = 0.1585132
rbxsa  ( 4, 2, 1) = 0.6937194e-01     rbxtbf ( 3, 3, 3) = 0.1088936       rbzf   ( 6) = 0.5243842e-01
rbxsa  ( 4, 2, 2) = 0.3828432e-02     rbxtbf ( 3, 3, 4) = 0.6174090e-01   rbzf   ( 7) = 0.0000000
rbxsa  ( 4, 3, 1) = 0.2890410e-01     rbxtbf ( 4, 1, 1) = 0.1362240       rbztbf ( 1, 1, 1) = 0.6781166e-01
rbxsa  ( 4, 3, 2) = 0.2511321e-04     rbxtbf ( 4, 1, 2) = 0.2932658       rbztbf ( 1, 1, 2) = 0.1198337
rbxsa  ( 5, 1, 1) = 0.2938117         rbxtbf ( 4, 1, 3) = 0.1362240       rbztbf ( 1, 1, 3) = 0.6403954e-01
rbxsa  ( 5, 1, 2) = 0.2711163e-01     rbxtbf ( 4, 1, 4) = 0.2932658       rbztbf ( 1, 1, 4) = 0.1210458
rbxsa  ( 5, 2, 1) = 0.6379201e-01     rbxtbf ( 4, 2, 1) = 0.6937194e-01   rbztbf ( 1, 2, 1) = 0.1582066e-01
rbxsa  ( 5, 2, 2) = 0.0000000         rbxtbf ( 4, 2, 2) = 0.3828432e-02   rbztbf ( 1, 2, 2) = 0.1595523e-01
rbxsa  ( 5, 3, 1) = 0.5053163e-01     rbxtbf ( 4, 2, 3) = 0.6937194e-01   rbztbf ( 1, 2, 3) = 0.1569817e-01
rbxsa  ( 5, 3, 2) = 0.0000000         rbxtbf ( 4, 2, 4) = 0.3828432e-02   rbztbf ( 1, 2, 4) = 0.1583346e-01
rbxsa  ( 6, 1, 1) = 0.0000000         rbxtbf ( 4, 3, 1) = 0.2890410e-01   rbztbf ( 1, 3, 1) = 0.5445164
rbxsa  ( 6, 1, 2) = 0.0000000         rbxtbf ( 4, 3, 2) = 0.2511321e-04   rbztbf ( 1, 3, 2) = 0.5304771
rbxsa  ( 6, 2, 1) = 0.2460780e-01     rbxtbf ( 4, 3, 3) = 0.2890410e-01   rbztbf ( 1, 3, 3) = 0.5399057
rbxsa  ( 6, 2, 2) = 0.5507410e-01     rbxtbf ( 4, 3, 4) = 0.2511321e-04   rbztbf ( 1, 3, 4) = 0.5404533
```

```
rbztbf ( 2, 1, 1) = 0.2541236e-01   rbztbf ( 6, 3, 1) = 0.2132088e-01   rdem ( 19) = 0.3999752e-02
rbztbf ( 2, 1, 2) = 0.1414644       rbztbf ( 6, 3, 2) = 0.8348780e-01   rdem ( 20) = 0.4058171e-02
rbztbf ( 2, 1, 3) = 0.2565425e-01   rbztbf ( 6, 3, 3) = 0.2203374e-01   rdem ( 21) = 0.4117443e-02
rbztbf ( 2, 1, 4) = 0.1438508       rbztbf ( 6, 3, 4) = 0.7858586e-01   rdem ( 22) = 0.4177580e-02
rbztbf ( 2, 2, 1) = 0.2780801e-01   rbztbf ( 7, 1, 1) = 0.0000000       rdem ( 23) = 0.4238597e-02
rbztbf ( 2, 2, 2) = 0.3602797e-01   rbztbf ( 7, 1, 2) = 0.0000000       rdem ( 24) = 0.4300504e-02
rbztbf ( 2, 2, 3) = 0.2810229e-01   rbztbf ( 7, 1, 3) = 0.0000000       rdem ( 25) = 0.4363316e-02
rbztbf ( 2, 2, 4) = 0.3130577e-01   rbztbf ( 7, 1, 4) = 0.0000000       rdem ( 26) = 0.4427045e-02
rbztbf ( 2, 3, 1) = 0.4910676e-01   rbztbf ( 7, 2, 1) = 0.0000000       rdem ( 27) = 0.4491705e-02
rbztbf ( 2, 3, 2) = 0.1637754       rbztbf ( 7, 2, 2) = 0.0000000       rdem ( 28) = 0.4557309e-02
rbztbf ( 2, 3, 3) = 0.4945974e-01   rbztbf ( 7, 2, 3) = 0.0000000       rdem ( 29) = 0.4623871e-02
rbztbf ( 2, 3, 4) = 0.1577730       rbztbf ( 7, 2, 4) = 0.0000000       rdem ( 30) = 0.4691406e-02
rbztbf ( 3, 1, 1) = 0.9729564e-01   rbztbf ( 7, 3, 1) = 0.0000000       rdem ( 31) = 0.4759927e-02
rbztbf ( 3, 1, 2) = 0.2043216       rbztbf ( 7, 3, 2) = 0.0000000       rdem ( 32) = 0.4829449e-02
rbztbf ( 3, 1, 3) = 0.9820614e-01   rbztbf ( 7, 3, 3) = 0.0000000       rdem ( 33) = 0.4899985e-02
rbztbf ( 3, 1, 4) = 0.2101141       rbztbf ( 7, 3, 4) = 0.0000000       rdem ( 34) = 0.4971553e-02
rbztbf ( 3, 2, 1) = 0.2314638       rcdemr ( 1) =  63430.91            rdem ( 35) = 0.5044166e-02
rbztbf ( 3, 2, 2) = 0.1183381       rcdemr ( 2) =  40452.54            rdem ( 36) = 0.5117839e-02
rbztbf ( 3, 2, 3) = 0.2192204       rcdemr ( 3) =  110694.7            rdem ( 37) = 0.5192589e-02
rbztbf ( 3, 2, 4) = 0.1107356       rcdemr ( 4) =  82739.05            rdem ( 38) = 0.5268429e-02
rbztbf ( 3, 3, 1) = 0.1201816       rcdemr ( 5) =  59253.92            rdem ( 39) = 0.5345379e-02
rbztbf ( 3, 3, 2) = 0.7617801e-01   rcdemr ( 6) =  8875.352            rdem ( 40) = 0.5423450e-02
rbztbf ( 3, 3, 3) = 0.1169895       rcdemr ( 7) =  0.0000000           rdem ( 41) = 0.5502664e-02
rbztbf ( 3, 3, 4) = 0.7296489e-01   rcoef ( 1) = 0.6695324e-01         rdem ( 42) = 0.5583035e-02
rbztbf ( 4, 1, 1) = 0.1227996       rcoef ( 2) = 0.3776830e-01         rdem ( 43) = 0.5664577e-02
rbztbf ( 4, 1, 2) = 0.2608151       rcoef ( 3) = 0.1376438e-01         rdem ( 44) = 0.5747311e-02
rbztbf ( 4, 1, 3) = 0.1265439       rcoef ( 4) = 0.1090795             rdem ( 45) = 0.5831257e-02
rbztbf ( 4, 1, 4) = 0.2682092       rcoef ( 5) = 0.4514909e-01         rdem ( 46) = 0.5916424e-02
rbztbf ( 4, 2, 1) = 0.6291889e-01   rcoef ( 6) = 0.4238796e-01         rdem ( 47) = 0.6002839e-02
rbztbf ( 4, 2, 2) = 0.3404805e-02   rcoef ( 7) = 0.4514909e-01         rdem ( 48) = 0.6090513e-02
rbztbf ( 4, 2, 3) = 0.6469905e-01   rcoef ( 8) = 0.1960528e-01         rdem ( 49) = 0.6179469e-02
rbztbf ( 4, 2, 4) = 0.3501331e-02   rcoef ( 9) = 0.4514909e-01         rdem ( 50) = 0.6269724e-02
rbztbf ( 4, 3, 1) = 0.2577798e-01   rcoef ( 10) = 0.4514909e-01        rdem ( 51) = 0.6361296e-02
rbztbf ( 4, 3, 2) = 0.2233436e-04   rcoef ( 11) = 0.4514909e-01        rdem ( 52) = 0.6454208e-02
rbztbf ( 4, 3, 3) = 0.2663985e-01   rcoef ( 12) = 0.4514909e-01        rdem ( 53) = 0.6548476e-02
rbztbf ( 4, 3, 4) = 0.2296754e-04   rcoef ( 13) = 0.6695324e-01        rdem ( 54) = 0.6644120e-02
rbztbf ( 5, 1, 1) = 0.2623400       rcoef ( 14) = 0.6695324e-01        rdem ( 55) = 0.6741161e-02
rbztbf ( 5, 1, 2) = 0.2411165e-01   rctot ( 1) =  110922.5             rdem ( 56) = 0.6839620e-02
rbztbf ( 5, 1, 3) = 0.2710536       rdem ( 1) = 0.3080932e-02          rdem ( 57) = 0.6939518e-02
rbztbf ( 5, 1, 4) = 0.2479522e-01   rdem ( 2) = 0.3125931e-02          rdem ( 58) = 0.7040873e-02
rbztbf ( 5, 2, 1) = 0.6335225e-01   rdem ( 3) = 0.3171588e-02          rdem ( 59) = 0.7143710e-02
rbztbf ( 5, 2, 2) = 0.0000000       rdem ( 4) = 0.3217911e-02          rdem ( 60) = 0.7248049e-02
rbztbf ( 5, 2, 3) = 0.6341080e-01   rdem ( 5) = 0.3264910e-02          rdem ( 61) = 0.7353911e-02
rbztbf ( 5, 2, 4) = 0.0000000       rdem ( 6) = 0.3312596e-02          rdem ( 62) = 0.7461320e-02
rbztbf ( 5, 3, 1) = 0.5170585e-01   rdem ( 7) = 0.3360979e-02          rdem ( 63) = 0.7570297e-02
rbztbf ( 5, 3, 2) = 0.0000000       rdem ( 8) = 0.3410068e-02          rdem ( 64) = 0.7680866e-02
rbztbf ( 5, 3, 3) = 0.5131566e-01   rdem ( 9) = 0.3459875e-02          rdem ( 65) = 0.7793050e-02
rbztbf ( 5, 3, 4) = 0.0000000       rdem ( 10) = 0.3510408e-02         rdem ( 66) = 0.7906872e-02
rbztbf ( 6, 1, 1) = 0.0000000       rdem ( 11) = 0.3561680e-02         rdem ( 67) = 0.8022357e-02
rbztbf ( 6, 1, 2) = 0.0000000       rdem ( 12) = 0.3613700e-02         rdem ( 68) = 0.8139528e-02
rbztbf ( 6, 1, 3) = 0.0000000       rdem ( 13) = 0.3666481e-02         rdem ( 69) = 0.8258413e-02
rbztbf ( 6, 1, 4) = 0.0000000       rdem ( 14) = 0.3720033e-02         rdem ( 70) = 0.8379031e-02
rbztbf ( 6, 2, 1) = 0.2297706e-01   rdem ( 15) = 0.3774366e-02         rdem ( 71) = 0.8501412e-02
rbztbf ( 6, 2, 2) = 0.7572722e-01   rdem ( 16) = 0.3829492e-02         rdem ( 72) = 0.8625580e-02
rbztbf ( 6, 2, 3) = 0.2337195e-01   rdem ( 17) = 0.3885425e-02         rdem ( 73) = 0.8751563e-02
rbztbf ( 6, 2, 4) = 0.7060882e-01   rdem ( 18) = 0.3942174e-02         rdem ( 74) = 0.8879385e-02
```

rdem	(75) =	0.9009075e-02
rdem	(76) =	0.9140657e-02
rdem	(77) =	0.9274163e-02
rdem	(78) =	0.9409617e-02
rdem	(79) =	0.9547051e-02
rdem	(80) =	0.9686491e-02
rdem	(81) =	0.9827969e-02
rdem	(82) =	0.9971512e-02
rdem	(83) =	0.1011715e-01
rdem	(84) =	0.1026492e-01
rdem	(85) =	0.1041484e-01
rdem	(86) =	0.1056696e-01
rdem	(87) =	0.1072130e-01
rdem	(88) =	0.1087789e-01
rdem	(89) =	0.1103677e-01
rdem	(90) =	0.1119797e-01
rdem	(91) =	0.1136152e-01
rdem	(92) =	0.1152746e-01
rdem	(93) =	0.1169583e-01
rdem	(94) =	0.1186666e-01
rdem	(95) =	0.1203997e-01
rdem	(96) =	0.1221582e-01
rdem	(97) =	0.1239425e-01
rdem	(98) =	0.1257527e-01
rdem	(99) =	0.1275894e-01
rdem	(100) =	0.1294529e-01
rdem	(101) =	0.1313437e-01
rdem	(102) =	0.1332620e-01
rdem	(103) =	0.1352084e-01
rdem	(104) =	0.1371832e-01
rdem	(105) =	0.1391868e-01
rdem	(106) =	0.1412198e-01
rdem	(107) =	0.1432824e-01
rdem	(108) =	0.1453751e-01
rdem	(109) =	0.1474984e-01
rdem	(110) =	0.1496527e-01
rdem	(111) =	0.1518385e-01
rdem	(112) =	0.1540562e-01
rdem	(113) =	0.1563063e-01
rdem	(114) =	0.1585892e-01
rdem	(115) =	0.1609055e-01
rdem	(116) =	0.1632557e-01
rdem	(117) =	0.1656401e-01
rdem	(118) =	0.1680594e-01
rdem	(119) =	0.1705140e-01
rdem	(120) =	0.1730045e-01
rdem	(121) =	0.1755313e-01
rdem	(122) =	0.1780950e-01
rdem	(123) =	0.1806962e-01
rdem	(124) =	0.1833354e-01
rdem	(125) =	0.1860132e-01
rdem	(126) =	0.1887300e-01
rdem	(127) =	0.1914865e-01
rdem	(128) =	0.1942833e-01
rdem	(129) =	0.1971209e-01
rdem	(130) =	0.2000000e-01
rdem	(131) =	0.0000000
rdem	(132) =	0.0000000
rdem	(133) =	0.0000000
rdem	(134) =	0.0000000
rdem	(135) =	0.0000000
rdem	(136) =	0.0000000
rdem	(137) =	0.0000000
rdem	(138) =	0.0000000
rdem	(139) =	0.0000000
rdem	(140) =	0.0000000
rdem	(141) =	0.0000000
rdem	(142) =	0.0000000
rdem	(143) =	0.0000000
rdem	(144) =	0.0000000
rdem	(145) =	0.0000000
rdem	(146) =	0.0000000
rdem	(147) =	0.0000000
rdem	(148) =	0.0000000
rdem	(149) =	0.0000000
rdem	(150) =	0.0000000
rdem	(151) =	0.0000000
rdem	(152) =	0.0000000
rdem	(153) =	0.0000000
rdem	(154) =	0.0000000
rdem	(155) =	0.0000000
rdem	(156) =	0.0000000
rdem	(157) =	0.0000000
rdem	(158) =	0.0000000
rdem	(159) =	0.0000000
rdem	(160) =	0.0000000
rdem	(161) =	0.0000000
rdem	(162) =	0.0000000
rdem	(163) =	0.0000000
rdem	(164) =	0.0000000
rdem	(165) =	0.0000000
rdem	(166) =	0.0000000
rdem	(167) =	0.0000000
rdem	(168) =	0.0000000
rdem	(169) =	0.0000000
rdem	(170) =	0.0000000
rdem	(171) =	0.0000000
rdem	(172) =	0.0000000
rdem	(173) =	0.0000000
rdem	(174) =	0.0000000
rdem	(175) =	0.0000000
rdem	(176) =	0.0000000
rdem	(177) =	0.0000000
rdem	(178) =	0.0000000
rdem	(179) =	0.0000000
rdem	(180) =	0.0000000
rdem	(181) =	0.0000000
rdem	(182) =	0.0000000
rdem	(183) =	0.0000000
rdem	(184) =	0.0000000
rdem	(185) =	0.0000000
rdem	(186) =	0.0000000
rdem	(187) =	0.0000000
rdem	(188) =	0.0000000
rdem	(189) =	0.0000000
rdem	(190) =	0.0000000
rdem	(191) =	0.0000000
rdem	(192) =	0.0000000
rdem	(193) =	0.0000000
rdem	(194) =	0.0000000
rdem	(195) =	0.0000000
rdem	(196) =	0.0000000
rdem	(197) =	0.0000000
rdem	(198) =	0.0000000
rdem	(199) =	0.0000000
rdem	(200) =	0.0000000
rdm	(1) =	0.2000000e-01
rdm	(2) =	0.1450000e-01
rdqfrc	(1) =	0.1943642
rdqfrc	(2) =	0.5702236
rdqfrc	(3) =	0.1924527
rdqfrc	(4) =	0.4295943e-01
rdqtyp	(1) =	164.0395
rdqtyp	(2) =	481.2571
rdqtyp	(3) =	162.4262
rdqtyp	(4) =	36.25689
rdqtyp	(5) =	843.9797
rdxfrc	(1) =	0.1739674
rdxfrc	(2) =	0.5013925
rdxfrc	(3) =	0.2688102
rdxfrc	(4) =	0.5582997e-01
rdxtyp	(1) =	2985.878
rdxtyp	(2) =	8605.614
rdxtyp	(3) =	4613.704
rdxtyp	(4) =	958.2338
rdxtyp	(5) =	17163.43
rhdfrc	(1) =	0.1749234
rhdfrc	(2) =	0.5046184
rhdfrc	(3) =	0.2652314
rhdfrc	(4) =	0.5522675e-01
rhdtyp	(1) =	3149.917
rhdtyp	(2) =	9086.871
rhdtyp	(3) =	4776.130
rhdtyp	(4) =	994.4907
rhdtyp	(5) =	18007.41
rhifrc	(1) =	0.2500000
rhifrc	(2) =	0.7500000
rhifrc	(3) =	0.7500000
rhifrc	(4) =	0.2500000
rhityp	(1) =	2612.689
rhityp	(2) =	7838.068
rhityp	(3) =	606.1818
rhityp	(4) =	202.0606
rhityp	(5) =	11259.00
rhiwf	(1) =	0.3225868e-02
rhiwf	(2) =	0.2073318e-02
rhiwf	(3) =	0.5320207e-02
rhiwf	(4) =	0.2063526e-02

```
rhiwf    (  5) =  0.2367804e-02
rhiwf    (  6) =  0.5234453e-03
rhiwf    (  7) =  0.0000000
rhiwtot  =  0.1557417e-01
rhnfrc   (  1) =  0.1969017
rhnfrc   (  2) =  0.5783060
rhnfrc   (  3) =  0.1839075
rhnfrc   (  4) =  0.4088480e-01
rhntyp   (  1) =   5762.607
rhntyp   (  2) =   16924.94
rhntyp   (  3) =   5382.312
rhntyp   (  4) =   1196.551
rhntyp   (  5) =   29266.41
rhqfrc   (  1) =  0.1919576
rhqfrc   (  2) =  0.5577143
rhqfrc   (  3) =  0.2026939
rhqfrc   (  4) =  0.4763408e-01
rhqtyp   (  1) =   50927.01
rhqtyp   (  2) =   147963.5
rhqtyp   (  3) =   53775.38
rhqtyp   (  4) =   12637.48
rhqtyp   (  5) =   265303.4
rhxfrc   (  1) =  0.1766598
rhxfrc   (  2) =  0.5002763
rhxfrc   (  3) =  0.2661579
rhxfrc   (  4) =  0.5690591e-01
rhxtyp   (  1) =   419943.9
rhxtyp   (  2) =   1189223.
rhxtyp   (  3) =   632692.6
rhxtyp   (  4) =   135272.9
rhxtyp   (  5) =   2377133.
rhy      (  1) =   28901.59
rhy      (  2) =   28901.59
rhy      (  3) =   28457.19
rhy      (  4) =   28030.37
rhy      (  5) =   27613.35
rhy      (  6) =   28139.31
rhy      (  7) =   28727.58
rhy      (  8) =   29369.57
rhy      (  9) =   30077.24
rhy      ( 10) =   31221.77
rhy      ( 11) =   44795.36
rhy      ( 12) =   44775.78
rhy      ( 13) =   44755.91
rhy      ( 14) =   44735.78
rhy      ( 15) =   44715.35
rhy      ( 16) =   44694.63
rhy      ( 17) =   44673.62
rhy      ( 18) =   44652.30
rhy      ( 19) =   44630.70
rhy      ( 20) =   28947.50
rhy      ( 21) =   28933.08
rhy      ( 22) =   28918.45
rhy      ( 23) =   28903.62
rhy      ( 24) =   28888.57
rhy      ( 25) =   28873.32
rhy      ( 26) =   28857.84
rhy      ( 27) =   28842.15
rhy      ( 28) =   28826.24
rhy      ( 29) =   28810.10
rhy      ( 30) =   28793.74
rhy      ( 31) =   28777.15
rhy      ( 32) =   28760.32
rhy      ( 33) =   28743.25
rhy      ( 34) =   28725.95
rhy      ( 35) =   28708.40
rhy      ( 36) =   12874.11
rhy      ( 37) =   12866.01
rhy      ( 38) =   12857.80
rhy      ( 39) =   12849.47
rhy      ( 40) =   12841.03
rhy      ( 41) =   12832.47
rhy      ( 42) =   12823.79
rhy      ( 43) =   12814.98
rhy      ( 44) =   12806.06
rhy      ( 45) =   12797.01
rhy      ( 46) =   12787.83
rhy      ( 47) =   12778.53
rhy      ( 48) =   12769.09
rhy      ( 49) =   12759.53
rhy      ( 50) =   12749.83
rhy      ( 51) =   12739.99
rhy      ( 52) =   12730.02
rhy      ( 53) =   12719.91
rhy      ( 54) =   12709.67
rhy      ( 55) =   12699.27
rhy      ( 56) =   12688.74
rhy      ( 57) =   12678.05
rhy      ( 58) =   12667.22
rhy      ( 59) =   12656.25
rhy      ( 60) =   12645.11
rhy      ( 61) =   12633.83
rhy      ( 62) =   14523.30
rhy      ( 63) =   14509.95
rhy      ( 64) =   14496.42
rhy      ( 65) =   14482.71
rhy      ( 66) =   14468.80
rhy      ( 67) =   14454.71
rhy      ( 68) =   14440.42
rhy      ( 69) =   14425.94
rhy      ( 70) =   14411.26
rhy      ( 71) =   14396.37
rhy      ( 72) =   14381.29
rhy      ( 73) =   14366.00
rhy      ( 74) =   14350.50
rhy      ( 75) =   14334.79
rhy      ( 76) =   14318.86
rhy      ( 77) =   14302.72
rhy      ( 78) =   14286.36
rhy      ( 79) =   14269.77
rhy      ( 80) =   14252.96
rhy      ( 81) =   14235.93
rhy      ( 82) =   14218.66
rhy      ( 83) =   14201.16
rhy      ( 84) =   14183.43
rhy      ( 85) =   14165.45
rhy      ( 86) =   14147.24
rhy      ( 87) =   14128.78
rhy      ( 88) =   14110.07
rhy      ( 89) =   14091.11
rhy      ( 90) =   14071.89
rhy      ( 91) =   14052.42
rhy      ( 92) =   14032.69
rhy      ( 93) =   14012.70
rhy      ( 94) =   13992.44
rhy      ( 95) =   13971.92
rhy      ( 96) =   13951.12
rhy      ( 97) =   13930.04
rhy      ( 98) =   13908.68
rhy      ( 99) =   13887.05
rhy      (100) =   13865.12
rhy      (101) =   12364.41
rhy      (102) =   12344.31
rhy      (103) =   12323.94
rhy      (104) =   12303.31
rhy      (105) =   12282.41
rhy      (106) =   12261.24
rhy      (107) =   12239.79
rhy      (108) =   12218.06
rhy      (109) =   12196.05
rhy      (110) =   12173.75
rhy      (111) =   12151.17
rhy      (112) =   12128.29
rhy      (113) =   12105.11
rhy      (114) =   12081.64
rhy      (115) =   12057.87
rhy      (116) =   12033.79
rhy      (117) =   12009.41
rhy      (118) =   11984.71
rhy      (119) =   11959.70
rhy      (120) =   11934.37
rhy      (121) =   11908.73
rhy      (122) =   11882.75
rhy      (123) =   11856.45
rhy      (124) =   11829.81
rhy      (125) =   11802.85
rhy      (126) =   11775.54
rhy      (127) =   11747.89
rhy      (128) =   11719.90
rhy      (129) =   11691.56
rhy      (130) =   11662.87
rhy      (131) =   11633.82
rhy      (132) =   11844.77
rhy      (133) =   12062.74
rhy      (134) =   12288.02
rhy      (135) =   12520.93
rhy      (136) =   12761.78
rhy      (137) =   13010.92
```

```
rhy     (138) =    13268.71
rhy     (139) =    13535.52
rhy     (140) =   0.0000000
rhy     (141) =   0.0000000
rhy     (142) =   0.0000000
rhy     (143) =   0.0000000
rhy     (144) =   0.0000000
rhy     (145) =   0.0000000
rhy     (146) =   0.0000000
rhy     (147) =   0.0000000
rhy     (148) =   0.0000000
rhy     (149) =   0.0000000
rhy     (150) =   0.0000000
rhy     (151) =   0.0000000
rhy     (152) =   0.0000000
rhy     (153) =   0.0000000
rhy     (154) =   0.0000000
rhy     (155) =   0.0000000
rhy     (156) =   0.0000000
rhy     (157) =   0.0000000
rhy     (158) =   0.0000000
rhy     (159) =   0.0000000
rhy     (160) =   0.0000000
rhy     (161) =   0.0000000
rhy     (162) =   0.0000000
rhy     (163) =   0.0000000
rhy     (164) =   0.0000000
rhy     (165) =   0.0000000
rhy     (166) =   0.0000000
rhy     (167) =   0.0000000
rhy     (168) =   0.0000000
rhy     (169) =   0.0000000
rhy     (170) =   0.0000000
rhy     (171) =   0.0000000
rhy     (172) =   0.0000000
rhy     (173) =   0.0000000
rhy     (174) =   0.0000000
rhy     (175) =   0.0000000
rhy     (176) =   0.0000000
rhy     (177) =   0.0000000
rhy     (178) =   0.0000000
rhy     (179) =   0.0000000
rhy     (180) =   0.0000000
rhy     (181) =   0.0000000
rhy     (182) =   0.0000000
rhy     (183) =   0.0000000
rhy     (184) =   0.0000000
rhy     (185) =   0.0000000
rhy     (186) =   0.0000000
rhy     (187) =   0.0000000
rhy     (188) =   0.0000000
rhy     (189) =   0.0000000
rhy     (190) =   0.0000000
rhy     (191) =   0.0000000
rhy     (192) =   0.0000000
rhy     (193) =   0.0000000

rhy     (194) =   0.0000000
rhy     (195) =   0.0000000
rhy     (196) =   0.0000000
rhy     (197) =   0.0000000
rhy     (198) =   0.0000000
rhy     (199) =   0.0000000
rhy     (200) =   0.0000000
rhy     (201) =   0.0000000
rhytyp  (  1,   1) =    5679.220
rhytyp  (  1,   2) =    16672.52
rhytyp  (  1,   3) =    5363.352
rhytyp  (  1,   4) =    1186.500
rhytyp  (  2,   1) =    5679.220
rhytyp  (  2,   2) =    16672.52
rhytyp  (  2,   3) =    5363.352
rhytyp  (  2,   4) =    1186.500
rhytyp  (  3,   1) =    5580.696
rhytyp  (  3,   2) =    16375.74
rhytyp  (  3,   3) =    5328.012
rhytyp  (  3,   4) =    1172.744
rhytyp  (  4,   1) =    5486.789
rhytyp  (  4,   2) =    16092.76
rhytyp  (  4,   3) =    5292.113
rhytyp  (  4,   4) =    1158.709
rhytyp  (  5,   1) =    5395.096
rhytyp  (  5,   2) =    15816.37
rhytyp  (  5,   3) =    5257.035
rhytyp  (  5,   4) =    1144.851
rhytyp  (  6,   1) =    5464.723
rhytyp  (  6,   2) =    16023.88
rhytyp  (  6,   3) =    5445.356
rhytyp  (  6,   4) =    1205.359
rhytyp  (  7,   1) =    5546.240
rhytyp  (  7,   2) =    16266.99
rhytyp  (  7,   3) =    5644.854
rhytyp  (  7,   4) =    1269.492
rhytyp  (  8,   1) =    5637.125
rhytyp  (  8,   2) =    16538.15
rhytyp  (  8,   3) =    5856.676
rhytyp  (  8,   4) =    1337.627
rhytyp  (  9,   1) =    5738.058
rhytyp  (  9,   2) =    16839.36
rhytyp  (  9,   3) =    6087.763
rhytyp  (  9,   4) =    1412.052
rhytyp  ( 10,   1) =    5165.433
rhytyp  ( 10,   2) =    15115.35
rhytyp  ( 10,   3) =    8676.222
rhytyp  ( 10,   4) =    2264.770
rhytyp  ( 11,   1) =    7795.387
rhytyp  ( 11,   2) =    22465.02
rhytyp  ( 11,   3) =    12038.83
rhytyp  ( 11,   4) =    2496.113
rhytyp  ( 12,   1) =    7791.980
rhytyp  ( 12,   2) =    22455.20
rhytyp  ( 12,   3) =    12033.57
rhytyp  ( 12,   4) =    2495.022

rhytyp  ( 13,   1) =    7788.523
rhytyp  ( 13,   2) =    22445.24
rhytyp  ( 13,   3) =    12028.23
rhytyp  ( 13,   4) =    2493.916
rhytyp  ( 14,   1) =    7785.019
rhytyp  ( 14,   2) =    22435.14
rhytyp  ( 14,   3) =    12022.82
rhytyp  ( 14,   4) =    2492.793
rhytyp  ( 15,   1) =    7781.463
rhytyp  ( 15,   2) =    22424.90
rhytyp  ( 15,   3) =    12017.33
rhytyp  ( 15,   4) =    2491.655
rhytyp  ( 16,   1) =    7777.859
rhytyp  ( 16,   2) =    22414.51
rhytyp  ( 16,   3) =    12011.76
rhytyp  ( 16,   4) =    2490.501
rhytyp  ( 17,   1) =    7774.203
rhytyp  ( 17,   2) =    22403.97
rhytyp  ( 17,   3) =    12006.11
rhytyp  ( 17,   4) =    2489.330
rhytyp  ( 18,   1) =    7770.495
rhytyp  ( 18,   2) =    22393.28
rhytyp  ( 18,   3) =    12000.39
rhytyp  ( 18,   4) =    2488.142
rhytyp  ( 19,   1) =    7766.734
rhytyp  ( 19,   2) =    22382.45
rhytyp  ( 19,   3) =    11994.58
rhytyp  ( 19,   4) =    2486.938
rhytyp  ( 20,   1) =    5037.509
rhytyp  ( 20,   2) =    14517.27
rhytyp  ( 20,   3) =    7779.693
rhytyp  ( 20,   4) =    1613.030
rhytyp  ( 21,   1) =    5034.999
rhytyp  ( 21,   2) =    14510.04
rhytyp  ( 21,   3) =    7775.817
rhytyp  ( 21,   4) =    1612.226
rhytyp  ( 22,   1) =    5032.453
rhytyp  ( 22,   2) =    14502.70
rhytyp  ( 22,   3) =    7771.885
rhytyp  ( 22,   4) =    1611.411
rhytyp  ( 23,   1) =    5029.872
rhytyp  ( 23,   2) =    14495.26
rhytyp  ( 23,   3) =    7767.899
rhytyp  ( 23,   4) =    1610.585
rhytyp  ( 24,   1) =    5027.254
rhytyp  ( 24,   2) =    14487.72
rhytyp  ( 24,   3) =    7763.855
rhytyp  ( 24,   4) =    1609.746
rhytyp  ( 25,   1) =    5024.598
rhytyp  ( 25,   2) =    14480.07
rhytyp  ( 25,   3) =    7759.755
rhytyp  ( 25,   4) =    1608.896
rhytyp  ( 26,   1) =    5021.906
rhytyp  ( 26,   2) =    14472.31
rhytyp  ( 26,   3) =    7755.597
rhytyp  ( 26,   4) =    1608.034
```

```
rhytyp ( 27, 1) = 5019.176
rhytyp ( 27, 2) = 14464.44
rhytyp ( 27, 3) = 7751.380
rhytyp ( 27, 4) = 1607.159
rhytyp ( 28, 1) = 5016.407
rhytyp ( 28, 2) = 14456.46
rhytyp ( 28, 3) = 7747.104
rhytyp ( 28, 4) = 1606.273
rhytyp ( 29, 1) = 5013.599
rhytyp ( 29, 2) = 14448.36
rhytyp ( 29, 3) = 7742.768
rhytyp ( 29, 4) = 1605.374
rhytyp ( 30, 1) = 5010.751
rhytyp ( 30, 2) = 14440.16
rhytyp ( 30, 3) = 7738.369
rhytyp ( 30, 4) = 1604.462
rhytyp ( 31, 1) = 5007.863
rhytyp ( 31, 2) = 14431.84
rhytyp ( 31, 3) = 7733.909
rhytyp ( 31, 4) = 1603.537
rhytyp ( 32, 1) = 5004.934
rhytyp ( 32, 2) = 14423.40
rhytyp ( 32, 3) = 7729.386
rhytyp ( 32, 4) = 1602.599
rhytyp ( 33, 1) = 5001.965
rhytyp ( 33, 2) = 14414.84
rhytyp ( 33, 3) = 7724.800
rhytyp ( 33, 4) = 1601.648
rhytyp ( 34, 1) = 4998.953
rhytyp ( 34, 2) = 14406.16
rhytyp ( 34, 3) = 7720.150
rhytyp ( 34, 4) = 1600.684
rhytyp ( 35, 1) = 4995.899
rhytyp ( 35, 2) = 14397.36
rhytyp ( 35, 3) = 7715.433
rhytyp ( 35, 4) = 1599.706
rhytyp ( 36, 1) = 2240.380
rhytyp ( 36, 2) = 6456.408
rhytyp ( 36, 3) = 3459.938
rhytyp ( 36, 4) = 717.3785
rhytyp ( 37, 1) = 2238.971
rhytyp ( 37, 2) = 6452.347
rhytyp ( 37, 3) = 3457.762
rhytyp ( 37, 4) = 716.9273
rhytyp ( 38, 1) = 2237.542
rhytyp ( 38, 2) = 6448.229
rhytyp ( 38, 3) = 3455.556
rhytyp ( 38, 4) = 716.4697
rhytyp ( 39, 1) = 2236.094
rhytyp ( 39, 2) = 6444.054
rhytyp ( 39, 3) = 3453.318
rhytyp ( 39, 4) = 716.0057
rhytyp ( 40, 1) = 2234.624
rhytyp ( 40, 2) = 6439.818
rhytyp ( 40, 3) = 3451.048
rhytyp ( 40, 4) = 715.5352
rhytyp ( 41, 1) = 2233.134
rhytyp ( 41, 2) = 6435.526
rhytyp ( 41, 3) = 3448.748
rhytyp ( 41, 4) = 715.0582
rhytyp ( 42, 1) = 2231.624
rhytyp ( 42, 2) = 6431.171
rhytyp ( 42, 3) = 3446.415
rhytyp ( 42, 4) = 714.5746
rhytyp ( 43, 1) = 2230.092
rhytyp ( 43, 2) = 6426.758
rhytyp ( 43, 3) = 3444.050
rhytyp ( 43, 4) = 714.0841
rhytyp ( 44, 1) = 2228.539
rhytyp ( 44, 2) = 6422.282
rhytyp ( 44, 3) = 3441.650
rhytyp ( 44, 4) = 713.5868
rhytyp ( 45, 1) = 2226.964
rhytyp ( 45, 2) = 6417.743
rhytyp ( 45, 3) = 3439.218
rhytyp ( 45, 4) = 713.0823
rhytyp ( 46, 1) = 2225.367
rhytyp ( 46, 2) = 6413.141
rhytyp ( 46, 3) = 3436.752
rhytyp ( 46, 4) = 712.5711
rhytyp ( 47, 1) = 2223.748
rhytyp ( 47, 2) = 6408.475
rhytyp ( 47, 3) = 3434.252
rhytyp ( 47, 4) = 712.0526
rhytyp ( 48, 1) = 2222.106
rhytyp ( 48, 2) = 6403.744
rhytyp ( 48, 3) = 3431.716
rhytyp ( 48, 4) = 711.5269
rhytyp ( 49, 1) = 2220.441
rhytyp ( 49, 2) = 6398.946
rhytyp ( 49, 3) = 3429.145
rhytyp ( 49, 4) = 710.9939
rhytyp ( 50, 1) = 2218.753
rhytyp ( 50, 2) = 6394.081
rhytyp ( 50, 3) = 3426.538
rhytyp ( 50, 4) = 710.4534
rhytyp ( 51, 1) = 2217.042
rhytyp ( 51, 2) = 6389.150
rhytyp ( 51, 3) = 3423.895
rhytyp ( 51, 4) = 709.9055
rhytyp ( 52, 1) = 2215.307
rhytyp ( 52, 2) = 6384.149
rhytyp ( 52, 3) = 3421.216
rhytyp ( 52, 4) = 709.3498
rhytyp ( 53, 1) = 2213.548
rhytyp ( 53, 2) = 6379.080
rhytyp ( 53, 3) = 3418.499
rhytyp ( 53, 4) = 708.7866
rhytyp ( 54, 1) = 2211.764
rhytyp ( 54, 2) = 6373.940
rhytyp ( 54, 3) = 3415.745
rhytyp ( 54, 4) = 708.2153
rhytyp ( 55, 1) = 2209.955
rhytyp ( 55, 2) = 6368.728
rhytyp ( 55, 3) = 3412.951
rhytyp ( 55, 4) = 707.6364
rhytyp ( 56, 1) = 2208.122
rhytyp ( 56, 2) = 6363.444
rhytyp ( 56, 3) = 3410.120
rhytyp ( 56, 4) = 707.0491
rhytyp ( 57, 1) = 2206.263
rhytyp ( 57, 2) = 6358.087
rhytyp ( 57, 3) = 3407.250
rhytyp ( 57, 4) = 706.4540
rhytyp ( 58, 1) = 2204.379
rhytyp ( 58, 2) = 6352.656
rhytyp ( 58, 3) = 3404.338
rhytyp ( 58, 4) = 705.8506
rhytyp ( 59, 1) = 2202.468
rhytyp ( 59, 2) = 6347.150
rhytyp ( 59, 3) = 3401.388
rhytyp ( 59, 4) = 705.2388
rhytyp ( 60, 1) = 2200.531
rhytyp ( 60, 2) = 6341.567
rhytyp ( 60, 3) = 3398.396
rhytyp ( 60, 4) = 704.6185
rhytyp ( 61, 1) = 2198.567
rhytyp ( 61, 2) = 6335.908
rhytyp ( 61, 3) = 3395.364
rhytyp ( 61, 4) = 703.9896
rhytyp ( 62, 1) = 2527.377
rhytyp ( 62, 2) = 7283.482
rhytyp ( 62, 3) = 3903.162
rhytyp ( 62, 4) = 809.2758
rhytyp ( 63, 1) = 2525.054
rhytyp ( 63, 2) = 7276.790
rhytyp ( 63, 3) = 3899.575
rhytyp ( 63, 4) = 808.5320
rhytyp ( 64, 1) = 2522.700
rhytyp ( 64, 2) = 7270.006
rhytyp ( 64, 3) = 3895.939
rhytyp ( 64, 4) = 807.7783
rhytyp ( 65, 1) = 2520.313
rhytyp ( 65, 2) = 7263.127
rhytyp ( 65, 3) = 3892.253
rhytyp ( 65, 4) = 807.0140
rhytyp ( 66, 1) = 2517.893
rhytyp ( 66, 2) = 7256.154
rhytyp ( 66, 3) = 3888.517
rhytyp ( 66, 4) = 806.2393
rhytyp ( 67, 1) = 2515.441
rhytyp ( 67, 2) = 7249.087
rhytyp ( 67, 3) = 3884.729
rhytyp ( 67, 4) = 805.4539
rhytyp ( 68, 1) = 2512.954
rhytyp ( 68, 2) = 7241.921
rhytyp ( 68, 3) = 3880.889
rhytyp ( 68, 4) = 804.6579
```

```
rhytyp ( 69, 1) = 2510.434
rhytyp ( 69, 2) = 7234.658
rhytyp ( 69, 3) = 3876.996
rhytyp ( 69, 4) = 803.8508
rhytyp ( 70, 1) = 2507.879
rhytyp ( 70, 2) = 7227.294
rhytyp ( 70, 3) = 3873.051
rhytyp ( 70, 4) = 803.0326
rhytyp ( 71, 1) = 2505.289
rhytyp ( 71, 2) = 7219.831
rhytyp ( 71, 3) = 3869.051
rhytyp ( 71, 4) = 802.2033
rhytyp ( 72, 1) = 2502.664
rhytyp ( 72, 2) = 7212.265
rhytyp ( 72, 3) = 3864.997
rhytyp ( 72, 4) = 801.3627
rhytyp ( 73, 1) = 2500.003
rhytyp ( 73, 2) = 7204.596
rhytyp ( 73, 3) = 3860.887
rhytyp ( 73, 4) = 800.5106
rhytyp ( 74, 1) = 2497.305
rhytyp ( 74, 2) = 7196.823
rhytyp ( 74, 3) = 3856.721
rhytyp ( 74, 4) = 799.6468
rhytyp ( 75, 1) = 2494.571
rhytyp ( 75, 2) = 7188.943
rhytyp ( 75, 3) = 3852.499
rhytyp ( 75, 4) = 798.7714
rhytyp ( 76, 1) = 2491.800
rhytyp ( 76, 2) = 7180.956
rhytyp ( 76, 3) = 3848.218
rhytyp ( 76, 4) = 797.8840
rhytyp ( 77, 1) = 2488.990
rhytyp ( 77, 2) = 7172.861
rhytyp ( 77, 3) = 3843.881
rhytyp ( 77, 4) = 796.9845
rhytyp ( 78, 1) = 2486.143
rhytyp ( 78, 2) = 7164.656
rhytyp ( 78, 3) = 3839.483
rhytyp ( 78, 4) = 796.0728
rhytyp ( 79, 1) = 2483.258
rhytyp ( 79, 2) = 7156.338
rhytyp ( 79, 3) = 3835.026
rhytyp ( 79, 4) = 795.1486
rhytyp ( 80, 1) = 2480.333
rhytyp ( 80, 2) = 7147.909
rhytyp ( 80, 3) = 3830.510
rhytyp ( 80, 4) = 794.2121
rhytyp ( 81, 1) = 2477.368
rhytyp ( 81, 2) = 7139.366
rhytyp ( 81, 3) = 3825.930
rhytyp ( 81, 4) = 793.2627
rhytyp ( 82, 1) = 2474.363
rhytyp ( 82, 2) = 7130.707
rhytyp ( 82, 3) = 3821.291
rhytyp ( 82, 4) = 792.3007
rhytyp ( 83, 1) = 2471.318
rhytyp ( 83, 2) = 7121.932
rhytyp ( 83, 3) = 3816.587
rhytyp ( 83, 4) = 791.3257
rhytyp ( 84, 1) = 2468.231
rhytyp ( 84, 2) = 7113.037
rhytyp ( 84, 3) = 3811.821
rhytyp ( 84, 4) = 790.3373
rhytyp ( 85, 1) = 2465.104
rhytyp ( 85, 2) = 7104.022
rhytyp ( 85, 3) = 3806.990
rhytyp ( 85, 4) = 789.3358
rhytyp ( 86, 1) = 2461.934
rhytyp ( 86, 2) = 7094.887
rhytyp ( 86, 3) = 3802.095
rhytyp ( 86, 4) = 788.3207
rhytyp ( 87, 1) = 2458.721
rhytyp ( 87, 2) = 7085.629
rhytyp ( 87, 3) = 3797.133
rhytyp ( 87, 4) = 787.2920
rhytyp ( 88, 1) = 2455.466
rhytyp ( 88, 2) = 7076.247
rhytyp ( 88, 3) = 3792.106
rhytyp ( 88, 4) = 786.2495
rhytyp ( 89, 1) = 2452.166
rhytyp ( 89, 2) = 7066.738
rhytyp ( 89, 3) = 3787.010
rhytyp ( 89, 4) = 785.1930
rhytyp ( 90, 1) = 2448.823
rhytyp ( 90, 2) = 7057.102
rhytyp ( 90, 3) = 3781.846
rhytyp ( 90, 4) = 784.1225
rhytyp ( 91, 1) = 2445.434
rhytyp ( 91, 2) = 7047.339
rhytyp ( 91, 3) = 3776.614
rhytyp ( 91, 4) = 783.0375
rhytyp ( 92, 1) = 2442.001
rhytyp ( 92, 2) = 7037.444
rhytyp ( 92, 3) = 3771.312
rhytyp ( 92, 4) = 781.9380
rhytyp ( 93, 1) = 2438.522
rhytyp ( 93, 2) = 7027.418
rhytyp ( 93, 3) = 3765.938
rhytyp ( 93, 4) = 780.8242
rhytyp ( 94, 1) = 2434.997
rhytyp ( 94, 2) = 7017.258
rhytyp ( 94, 3) = 3760.494
rhytyp ( 94, 4) = 779.6953
rhytyp ( 95, 1) = 2431.424
rhytyp ( 95, 2) = 7006.964
rhytyp ( 95, 3) = 3754.977
rhytyp ( 95, 4) = 778.5513
rhytyp ( 96, 1) = 2427.804
rhytyp ( 96, 2) = 6996.531
rhytyp ( 96, 3) = 3749.387
rhytyp ( 96, 4) = 777.3923
rhytyp ( 97, 1) = 2424.136
rhytyp ( 97, 2) = 6985.962
rhytyp ( 97, 3) = 3743.723
rhytyp ( 97, 4) = 776.2178
rhytyp ( 98, 1) = 2420.420
rhytyp ( 98, 2) = 6975.252
rhytyp ( 98, 3) = 3737.983
rhytyp ( 98, 4) = 775.0278
rhytyp ( 99, 1) = 2416.655
rhytyp ( 99, 2) = 6964.401
rhytyp ( 99, 3) = 3732.168
rhytyp ( 99, 4) = 773.8223
rhytyp (100, 1) = 2412.840
rhytyp (100, 2) = 6953.407
rhytyp (100, 3) = 3726.276
rhytyp (100, 4) = 772.6006
rhytyp (101, 1) = 2151.682
rhytyp (101, 2) = 6200.792
rhytyp (101, 3) = 3322.956
rhytyp (101, 4) = 688.9767
rhytyp (102, 1) = 2148.184
rhytyp (102, 2) = 6190.712
rhytyp (102, 3) = 3317.554
rhytyp (102, 4) = 687.8568
rhytyp (103, 1) = 2144.640
rhytyp (103, 2) = 6180.500
rhytyp (103, 3) = 3312.081
rhytyp (103, 4) = 686.7220
rhytyp (104, 1) = 2141.050
rhytyp (104, 2) = 6170.153
rhytyp (104, 3) = 3306.537
rhytyp (104, 4) = 685.5724
rhytyp (105, 1) = 2137.412
rhytyp (105, 2) = 6159.670
rhytyp (105, 3) = 3300.919
rhytyp (105, 4) = 684.4077
rhytyp (106, 1) = 2133.728
rhytyp (106, 2) = 6149.052
rhytyp (106, 3) = 3295.229
rhytyp (106, 4) = 683.2279
rhytyp (107, 1) = 2129.995
rhytyp (107, 2) = 6138.295
rhytyp (107, 3) = 3289.464
rhytyp (107, 4) = 682.0327
rhytyp (108, 1) = 2126.214
rhytyp (108, 2) = 6127.397
rhytyp (108, 3) = 3283.625
rhytyp (108, 4) = 680.8217
rhytyp (109, 1) = 2122.383
rhytyp (109, 2) = 6116.358
rhytyp (109, 3) = 3277.709
rhytyp (109, 4) = 679.5952
rhytyp (110, 1) = 2118.503
rhytyp (110, 2) = 6105.177
rhytyp (110, 3) = 3271.717
rhytyp (110, 4) = 678.3528
```

```
rhytyp (111, 1) = 2114.573
rhytyp (111, 2) = 6093.850
rhytyp (111, 3) = 3265.647
rhytyp (111, 4) = 677.0943
rhytyp (112, 1) = 2110.592
rhytyp (112, 2) = 6082.377
rhytyp (112, 3) = 3259.498
rhytyp (112, 4) = 675.8196
rhytyp (113, 1) = 2106.559
rhytyp (113, 2) = 6070.756
rhytyp (113, 3) = 3253.271
rhytyp (113, 4) = 674.5283
rhytyp (114, 1) = 2102.475
rhytyp (114, 2) = 6058.986
rhytyp (114, 3) = 3246.963
rhytyp (114, 4) = 673.2205
rhytyp (115, 1) = 2098.338
rhytyp (115, 2) = 6047.063
rhytyp (115, 3) = 3240.574
rhytyp (115, 4) = 671.8958
rhytyp (116, 1) = 2094.147
rhytyp (116, 2) = 6034.989
rhytyp (116, 3) = 3234.103
rhytyp (116, 4) = 670.5542
rhytyp (117, 1) = 2089.904
rhytyp (117, 2) = 6022.759
rhytyp (117, 3) = 3227.550
rhytyp (117, 4) = 669.1953
rhytyp (118, 1) = 2085.606
rhytyp (118, 2) = 6010.374
rhytyp (118, 3) = 3220.913
rhytyp (118, 4) = 667.8192
rhytyp (119, 1) = 2081.254
rhytyp (119, 2) = 5997.831
rhytyp (119, 3) = 3214.191
rhytyp (119, 4) = 666.4255
rhytyp (120, 1) = 2076.847
rhytyp (120, 2) = 5985.129
rhytyp (120, 3) = 3207.384
rhytyp (120, 4) = 665.0143
rhytyp (121, 1) = 2072.383
rhytyp (121, 2) = 5972.267
rhytyp (121, 3) = 3200.491
rhytyp (121, 4) = 663.5849
rhytyp (122, 1) = 2067.863
rhytyp (122, 2) = 5959.240
rhytyp (122, 3) = 3193.510
rhytyp (122, 4) = 662.1376
rhytyp (123, 1) = 2063.285
rhytyp (123, 2) = 5946.049
rhytyp (123, 3) = 3186.441
rhytyp (123, 4) = 660.6719
rhytyp (124, 1) = 2058.650
rhytyp (124, 2) = 5932.692
rhytyp (124, 3) = 3179.283
rhytyp (124, 4) = 659.1878
rhytyp (125, 1) = 2053.957
rhytyp (125, 2) = 5919.168
rhytyp (125, 3) = 3172.035
rhytyp (125, 4) = 657.6851
rhytyp (126, 1) = 2049.206
rhytyp (126, 2) = 5905.474
rhytyp (126, 3) = 3164.697
rhytyp (126, 4) = 656.1636
rhytyp (127, 1) = 2044.394
rhytyp (127, 2) = 5891.608
rhytyp (127, 3) = 3157.267
rhytyp (127, 4) = 654.6230
rhytyp (128, 1) = 2039.523
rhytyp (128, 2) = 5877.570
rhytyp (128, 3) = 3149.744
rhytyp (128, 4) = 653.0632
rhytyp (129, 1) = 2034.591
rhytyp (129, 2) = 5863.356
rhytyp (129, 3) = 3142.127
rhytyp (129, 4) = 651.4840
rhytyp (130, 1) = 2029.598
rhytyp (130, 2) = 5848.968
rhytyp (130, 3) = 3134.416
rhytyp (130, 4) = 649.8852
rhytyp (131, 1) = 2024.544
rhytyp (131, 2) = 5834.401
rhytyp (131, 3) = 3126.610
rhytyp (131, 4) = 648.2667
rhytyp (132, 1) = 2061.253
rhytyp (132, 2) = 5940.193
rhytyp (132, 3) = 3183.303
rhytyp (132, 4) = 660.0213
rhytyp (133, 1) = 2099.185
rhytyp (133, 2) = 6049.505
rhytyp (133, 3) = 3241.882
rhytyp (133, 4) = 672.1671
rhytyp (134, 1) = 2138.389
rhytyp (134, 2) = 6162.485
rhytyp (134, 3) = 3302.427
rhytyp (134, 4) = 684.7205
rhytyp (135, 1) = 2178.919
rhytyp (135, 2) = 6279.288
rhytyp (135, 3) = 3365.022
rhytyp (135, 4) = 697.6985
rhytyp (136, 1) = 2220.834
rhytyp (136, 2) = 6400.077
rhytyp (136, 3) = 3429.751
rhytyp (136, 4) = 711.1194
rhytyp (137, 1) = 2264.190
rhytyp (137, 2) = 6525.022
rhytyp (137, 3) = 3496.708
rhytyp (137, 4) = 725.0023
rhytyp (138, 1) = 2309.051
rhytyp (138, 2) = 6654.304
rhytyp (138, 3) = 3565.990
rhytyp (138, 4) = 739.3669
rhytyp (139, 1) = 2355.482
rhytyp (139, 2) = 6788.112
rhytyp (139, 3) = 3637.696
rhytyp (139, 4) = 754.2345
rhytyp (140, 1) = 0.0000000
rhytyp (140, 2) = 0.0000000
rhytyp (140, 3) = 0.0000000
rhytyp (140, 4) = 0.0000000
rhytyp (141, 1) = 0.0000000
rhytyp (141, 2) = 0.0000000
rhytyp (141, 3) = 0.0000000
rhytyp (141, 4) = 0.0000000
rhytyp (142, 1) = 0.0000000
rhytyp (142, 2) = 0.0000000
rhytyp (142, 3) = 0.0000000
rhytyp (142, 4) = 0.0000000
rhytyp (143, 1) = 0.0000000
rhytyp (143, 2) = 0.0000000
rhytyp (143, 3) = 0.0000000
rhytyp (143, 4) = 0.0000000
rhytyp (144, 1) = 0.0000000
rhytyp (144, 2) = 0.0000000
rhytyp (144, 3) = 0.0000000
rhytyp (144, 4) = 0.0000000
rhytyp (145, 1) = 0.0000000
rhytyp (145, 2) = 0.0000000
rhytyp (145, 3) = 0.0000000
rhytyp (145, 4) = 0.0000000
rhytyp (146, 1) = 0.0000000
rhytyp (146, 2) = 0.0000000
rhytyp (146, 3) = 0.0000000
rhytyp (146, 4) = 0.0000000
rhytyp (147, 1) = 0.0000000
rhytyp (147, 2) = 0.0000000
rhytyp (147, 3) = 0.0000000
rhytyp (147, 4) = 0.0000000
rhytyp (148, 1) = 0.0000000
rhytyp (148, 2) = 0.0000000
rhytyp (148, 3) = 0.0000000
rhytyp (148, 4) = 0.0000000
rhytyp (149, 1) = 0.0000000
rhytyp (149, 2) = 0.0000000
rhytyp (149, 3) = 0.0000000
rhytyp (149, 4) = 0.0000000
rhytyp (150, 1) = 0.0000000
rhytyp (150, 2) = 0.0000000
rhytyp (150, 3) = 0.0000000
rhytyp (150, 4) = 0.0000000
rhytyp (151, 1) = 0.0000000
rhytyp (151, 2) = 0.0000000
rhytyp (151, 3) = 0.0000000
rhytyp (151, 4) = 0.0000000
rhytyp (152, 1) = 0.0000000
rhytyp (152, 2) = 0.0000000
rhytyp (152, 3) = 0.0000000
rhytyp (152, 4) = 0.0000000
```

```
rhytyp  (153,  1) =  0.0000000
rhytyp  (153,  2) =  0.0000000
rhytyp  (153,  3) =  0.0000000
rhytyp  (153,  4) =  0.0000000
rhytyp  (154,  1) =  0.0000000
rhytyp  (154,  2) =  0.0000000
rhytyp  (154,  3) =  0.0000000
rhytyp  (154,  4) =  0.0000000
rhytyp  (155,  1) =  0.0000000
rhytyp  (155,  2) =  0.0000000
rhytyp  (155,  3) =  0.0000000
rhytyp  (155,  4) =  0.0000000
rhytyp  (156,  1) =  0.0000000
rhytyp  (156,  2) =  0.0000000
rhytyp  (156,  3) =  0.0000000
rhytyp  (156,  4) =  0.0000000
rhytyp  (157,  1) =  0.0000000
rhytyp  (157,  2) =  0.0000000
rhytyp  (157,  3) =  0.0000000
rhytyp  (157,  4) =  0.0000000
rhytyp  (158,  1) =  0.0000000
rhytyp  (158,  2) =  0.0000000
rhytyp  (158,  3) =  0.0000000
rhytyp  (158,  4) =  0.0000000
rhytyp  (159,  1) =  0.0000000
rhytyp  (159,  2) =  0.0000000
rhytyp  (159,  3) =  0.0000000
rhytyp  (159,  4) =  0.0000000
rhytyp  (160,  1) =  0.0000000
rhytyp  (160,  2) =  0.0000000
rhytyp  (160,  3) =  0.0000000
rhytyp  (160,  4) =  0.0000000
rhytyp  (161,  1) =  0.0000000
rhytyp  (161,  2) =  0.0000000
rhytyp  (161,  3) =  0.0000000
rhytyp  (161,  4) =  0.0000000
rhytyp  (162,  1) =  0.0000000
rhytyp  (162,  2) =  0.0000000
rhytyp  (162,  3) =  0.0000000
rhytyp  (162,  4) =  0.0000000
rhytyp  (163,  1) =  0.0000000
rhytyp  (163,  2) =  0.0000000
rhytyp  (163,  3) =  0.0000000
rhytyp  (163,  4) =  0.0000000
rhytyp  (164,  1) =  0.0000000
rhytyp  (164,  2) =  0.0000000
rhytyp  (164,  3) =  0.0000000
rhytyp  (164,  4) =  0.0000000
rhytyp  (165,  1) =  0.0000000
rhytyp  (165,  2) =  0.0000000
rhytyp  (165,  3) =  0.0000000
rhytyp  (165,  4) =  0.0000000
rhytyp  (166,  1) =  0.0000000
rhytyp  (166,  2) =  0.0000000
rhytyp  (166,  3) =  0.0000000
rhytyp  (166,  4) =  0.0000000
rhytyp  (167,  1) =  0.0000000
rhytyp  (167,  2) =  0.0000000
rhytyp  (167,  3) =  0.0000000
rhytyp  (167,  4) =  0.0000000
rhytyp  (168,  1) =  0.0000000
rhytyp  (168,  2) =  0.0000000
rhytyp  (168,  3) =  0.0000000
rhytyp  (168,  4) =  0.0000000
rhytyp  (169,  1) =  0.0000000
rhytyp  (169,  2) =  0.0000000
rhytyp  (169,  3) =  0.0000000
rhytyp  (169,  4) =  0.0000000
rhytyp  (170,  1) =  0.0000000
rhytyp  (170,  2) =  0.0000000
rhytyp  (170,  3) =  0.0000000
rhytyp  (170,  4) =  0.0000000
rhytyp  (171,  1) =  0.0000000
rhytyp  (171,  2) =  0.0000000
rhytyp  (171,  3) =  0.0000000
rhytyp  (171,  4) =  0.0000000
rhytyp  (172,  1) =  0.0000000
rhytyp  (172,  2) =  0.0000000
rhytyp  (172,  3) =  0.0000000
rhytyp  (172,  4) =  0.0000000
rhytyp  (173,  1) =  0.0000000
rhytyp  (173,  2) =  0.0000000
rhytyp  (173,  3) =  0.0000000
rhytyp  (173,  4) =  0.0000000
rhytyp  (174,  1) =  0.0000000
rhytyp  (174,  2) =  0.0000000
rhytyp  (174,  3) =  0.0000000
rhytyp  (174,  4) =  0.0000000
rhytyp  (175,  1) =  0.0000000
rhytyp  (175,  2) =  0.0000000
rhytyp  (175,  3) =  0.0000000
rhytyp  (175,  4) =  0.0000000
rhytyp  (176,  1) =  0.0000000
rhytyp  (176,  2) =  0.0000000
rhytyp  (176,  3) =  0.0000000
rhytyp  (176,  4) =  0.0000000
rhytyp  (177,  1) =  0.0000000
rhytyp  (177,  2) =  0.0000000
rhytyp  (177,  3) =  0.0000000
rhytyp  (177,  4) =  0.0000000
rhytyp  (178,  1) =  0.0000000
rhytyp  (178,  2) =  0.0000000
rhytyp  (178,  3) =  0.0000000
rhytyp  (178,  4) =  0.0000000
rhytyp  (179,  1) =  0.0000000
rhytyp  (179,  2) =  0.0000000
rhytyp  (179,  3) =  0.0000000
rhytyp  (179,  4) =  0.0000000
rhytyp  (180,  1) =  0.0000000
rhytyp  (180,  2) =  0.0000000
rhytyp  (180,  3) =  0.0000000
rhytyp  (180,  4) =  0.0000000
rhytyp  (181,  1) =  0.0000000
rhytyp  (181,  2) =  0.0000000
rhytyp  (181,  3) =  0.0000000
rhytyp  (181,  4) =  0.0000000
rhytyp  (182,  1) =  0.0000000
rhytyp  (182,  2) =  0.0000000
rhytyp  (182,  3) =  0.0000000
rhytyp  (182,  4) =  0.0000000
rhytyp  (183,  1) =  0.0000000
rhytyp  (183,  2) =  0.0000000
rhytyp  (183,  3) =  0.0000000
rhytyp  (183,  4) =  0.0000000
rhytyp  (184,  1) =  0.0000000
rhytyp  (184,  2) =  0.0000000
rhytyp  (184,  3) =  0.0000000
rhytyp  (184,  4) =  0.0000000
rhytyp  (185,  1) =  0.0000000
rhytyp  (185,  2) =  0.0000000
rhytyp  (185,  3) =  0.0000000
rhytyp  (185,  4) =  0.0000000
rhytyp  (186,  1) =  0.0000000
rhytyp  (186,  2) =  0.0000000
rhytyp  (186,  3) =  0.0000000
rhytyp  (186,  4) =  0.0000000
rhytyp  (187,  1) =  0.0000000
rhytyp  (187,  2) =  0.0000000
rhytyp  (187,  3) =  0.0000000
rhytyp  (187,  4) =  0.0000000
rhytyp  (188,  1) =  0.0000000
rhytyp  (188,  2) =  0.0000000
rhytyp  (188,  3) =  0.0000000
rhytyp  (188,  4) =  0.0000000
rhytyp  (189,  1) =  0.0000000
rhytyp  (189,  2) =  0.0000000
rhytyp  (189,  3) =  0.0000000
rhytyp  (189,  4) =  0.0000000
rhytyp  (190,  1) =  0.0000000
rhytyp  (190,  2) =  0.0000000
rhytyp  (190,  3) =  0.0000000
rhytyp  (190,  4) =  0.0000000
rhytyp  (191,  1) =  0.0000000
rhytyp  (191,  2) =  0.0000000
rhytyp  (191,  3) =  0.0000000
rhytyp  (191,  4) =  0.0000000
rhytyp  (192,  1) =  0.0000000
rhytyp  (192,  2) =  0.0000000
rhytyp  (192,  3) =  0.0000000
rhytyp  (192,  4) =  0.0000000
rhytyp  (193,  1) =  0.0000000
rhytyp  (193,  2) =  0.0000000
rhytyp  (193,  3) =  0.0000000
rhytyp  (193,  4) =  0.0000000
rhytyp  (194,  1) =  0.0000000
rhytyp  (194,  2) =  0.0000000
rhytyp  (194,  3) =  0.0000000
rhytyp  (194,  4) =  0.0000000
```

```
rhytyp   (195,  1) =   0.0000000
rhytyp   (195,  2) =   0.0000000
rhytyp   (195,  3) =   0.0000000
rhytyp   (195,  4) =   0.0000000
rhytyp   (196,  1) =   0.0000000
rhytyp   (196,  2) =   0.0000000
rhytyp   (196,  3) =   0.0000000
rhytyp   (196,  4) =   0.0000000
rhytyp   (197,  1) =   0.0000000
rhytyp   (197,  2) =   0.0000000
rhytyp   (197,  3) =   0.0000000
rhytyp   (197,  4) =   0.0000000
rhytyp   (198,  1) =   0.0000000
rhytyp   (198,  2) =   0.0000000
rhytyp   (198,  3) =   0.0000000
rhytyp   (198,  4) =   0.0000000
rhytyp   (199,  1) =   0.0000000
rhytyp   (199,  2) =   0.0000000
rhytyp   (199,  3) =   0.0000000
rhytyp   (199,  4) =   0.0000000
rhytyp   (200,  1) =   0.0000000
rhytyp   (200,  2) =   0.0000000
rhytyp   (200,  3) =   0.0000000
rhytyp   (200,  4) =   0.0000000
rhytyp   (201,  1) =   0.0000000
rhytyp   (201,  2) =   0.0000000
rhytyp   (201,  3) =   0.0000000
rhytyp   (201,  4) =   0.0000000
rhzfrc   (  1) =   0.1781958
rhzfrc   (  2) =   0.5060431
rhzfrc   (  3) =   0.2597860
rhzfrc   (  4) =   0.5597500e-01
rhzsiz   (  1) =    1149612.
rhzsiz   (  2) =    1475182.
rhztyp   (  1) =    468493.1
rhztyp   (  2) =    1327664.
rhztyp   (  3) =    681118.8
rhztyp   (  4) =    147518.3
rhztyp   (  5) =    2624794.
ribf     (  1) =   0.6136509e-03
ribf     (  2) =   0.5783233e-03
ribf     (  3) =   0.1861050e-02
ribf     (  4) =   0.7218375e-03
ribf     (  5) =   0.8282766e-03
ribf     (  6) =   0.1831053e-03
ribf     (  7) =   0.0000000
ribtot   =   0.4786244e-02
risf     (  1) =   0.5147832e-03
risf     (  2) =   0.1469397e-03
ristot   =   0.6617229e-03
riwf     (  1) =   0.1128434e-02
riwf     (  2) =   0.7252630e-03
riwf     (  3) =   0.1861050e-02
riwf     (  4) =   0.7218375e-03
riwf     (  5) =   0.8282766e-03
riwf     (  6) =   0.1831053e-03
```

```
riwf     (  7) =   0.0000000
riwtot   =   0.5447967e-02
rkw      (  1) =      9.845627
rkw      (  2) =      6.327945
rkw      (  3) =      16.23773
rkw      (  4) =      6.298057
rkw      (  5) =      7.226741
rkw      (  6) =      1.597600
rkw      (  7) =   0.0000000
rmxage   =    130
rphthy   (  1) =      1500.000
rphthy   (  2) =      1500.000
rphthy   (  3) =      1300.000
rphthy   (  4) =      1300.000
rpkcal   (  1) =      120.0000
rpkcal   (  2) =      90.00000
rpkcal   (  3) =      120.0000
rpkcal   (  4) =      90.00000
rpn      =   0.1871723e-01
rpnb     (  1) =   0.2348861e-02
rpnb     (  2) =   0.1453873e-01
rpnb     (  3) =   0.1829640e-02
rpnbf    (  1,  1) =   0.1431884e-02
rpnbf    (  1,  2) =   0.4618037e-03
rpnbf    (  1,  3) =   0.4551739e-03
rpnbf    (  1,  4) =   0.0000000
rpnbf    (  1,  5) =   0.0000000
rpnbf    (  1,  6) =   0.0000000
rpnbf    (  1,  7) =   0.0000000
rpnbf    (  2,  1) =   0.3165472e-03
rpnbf    (  2,  2) =   0.1828449e-02
rpnbf    (  2,  3) =   0.9730049e-02
rpnbf    (  2,  4) =   0.9746677e-04
rpnbf    (  2,  5) =   0.8169903e-03
rpnbf    (  2,  6) =   0.1749228e-02
rpnbf    (  2,  7) =   0.0000000
rpnbf    (  3,  1) =   0.4570487e-03
rpnbf    (  3,  2) =   0.4357598e-03
rpnbf    (  3,  3) =   0.6278115e-03
rpnbf    (  3,  4) =   0.0000000
rpnbf    (  3,  5) =   0.7528006e-04
rpnbf    (  3,  6) =   0.2337396e-03
rpnbf    (  3,  7) =   0.0000000
rpnf     (  1) =   0.2205480e-02
rpnf     (  2) =   0.2726013e-02
rpnf     (  3) =   0.1081304e-01
rpnf     (  4) =   0.9746677e-04
rpnf     (  5) =   0.8922704e-03
rpnf     (  6) =   0.1982967e-02
rpnf     (  7) =   0.0000000
rpnhms   (  1) =      105.0000
rpnhms   (  2) =      67.00000
rpnhms   (  3) =      105.0000
rpnhms   (  4) =      67.00000
rpnhms   (  5) =      81.51884
rpnins   (  1) =      1.000000
```

```
rpnins   (  2) =      1.000000
rpnins   (  3) =      1.000000
rpnins   (  4) =      1.000000
rpnlac   (  1) =      36.72983
rpnlac   (  2) =      23.43713
rpnlac   (  3) =      36.72983
rpnlac   (  4) =      23.43713
rpnlac   (  5) =      28.51594
rpnt     (  1) =   0.2865646e-01
rpnt     (  2) =   0.1376668e-01
rpnt     (  3) =   0.2510020e-01
rpnt     (  4) =   0.1216237e-01
rpntb    (  1,  1) =   0.4831348e-02
rpntb    (  1,  2) =   0.2184059e-01
rpntb    (  1,  3) =   0.1984521e-02
rpntb    (  2,  1) =   0.1022435e-02
rpntb    (  2,  2) =   0.1100986e-01
rpntb    (  2,  3) =   0.1734386e-02
rpntb    (  3,  1) =   0.4187168e-02
rpntb    (  3,  2) =   0.1892851e-01
rpntb    (  3,  3) =   0.1984521e-02
rpntb    (  4,  1) =   0.8861107e-03
rpntb    (  4,  2) =   0.9541878e-02
rpntb    (  4,  3) =   0.1734386e-02
rpntbf   (  1,  1,  1) =     1.000000
rpntbf   (  1,  1,  2) =     1.000000
rpntbf   (  1,  1,  3) =     1.000000
rpntbf   (  1,  1,  4) =     1.000000
rpntbf   (  1,  2,  1) =     1.000000
rpntbf   (  1,  2,  2) =     1.000000
rpntbf   (  1,  2,  3) =     1.000000
rpntbf   (  1,  2,  4) =     1.000000
rpntbf   (  1,  3,  1) =     1.000000
rpntbf   (  1,  3,  2) =     1.000000
rpntbf   (  1,  3,  3) =     1.000000
rpntbf   (  1,  3,  4) =     1.000000
rpntbf   (  2,  1,  1) =     1.350000
rpntbf   (  2,  1,  2) =     1.350000
rpntbf   (  2,  1,  3) =     1.350000
rpntbf   (  2,  1,  4) =     1.350000
rpntbf   (  2,  2,  1) =     1.350000
rpntbf   (  2,  2,  2) =     1.350000
rpntbf   (  2,  2,  3) =     1.350000
rpntbf   (  2,  2,  4) =     1.350000
rpntbf   (  2,  3,  1) =     1.420000
rpntbf   (  2,  3,  2) =     1.420000
rpntbf   (  2,  3,  3) =     1.420000
rpntbf   (  2,  3,  4) =     1.420000
rpntbf   (  3,  1,  1) =     1.550000
rpntbf   (  3,  1,  2) =     1.550000
rpntbf   (  3,  1,  3) =     1.550000
rpntbf   (  3,  1,  4) =     1.550000
rpntbf   (  3,  2,  1) =     1.550000
rpntbf   (  3,  2,  2) =     1.650000
rpntbf   (  3,  2,  3) =     1.550000
rpntbf   (  3,  2,  4) =     1.650000
```

```
rpntbf   ( 3, 3, 1) =   2.850000
rpntbf   ( 3, 3, 2) =   2.000000
rpntbf   ( 3, 3, 3) =   2.850000
rpntbf   ( 3, 3, 4) =   2.000000
rpntbf   ( 4, 1, 1) =  0.0000000
rpntbf   ( 4, 1, 2) =  0.0000000
rpntbf   ( 4, 1, 3) =  0.0000000
rpntbf   ( 4, 1, 4) =  0.0000000
rpntbf   ( 4, 2, 1) =   1.450000
rpntbf   ( 4, 2, 2) =   1.450000
rpntbf   ( 4, 2, 3) =   1.450000
rpntbf   ( 4, 2, 4) =   1.450000
rpntbf   ( 4, 3, 1) =   4.000000
rpntbf   ( 4, 3, 2) =   3.000000
rpntbf   ( 4, 3, 3) =   4.000000
rpntbf   ( 4, 3, 4) =   3.000000
rpntbf   ( 5, 1, 1) =  0.0000000
rpntbf   ( 5, 1, 2) =  0.0000000
rpntbf   ( 5, 1, 3) =  0.0000000
rpntbf   ( 5, 1, 4) =  0.0000000
rpntbf   ( 5, 2, 1) =   1.500000
rpntbf   ( 5, 2, 2) =   1.500000
rpntbf   ( 5, 2, 3) =   1.500000
rpntbf   ( 5, 2, 4) =   1.500000
rpntbf   ( 5, 3, 1) =   2.000000
rpntbf   ( 5, 3, 2) =   2.000000
rpntbf   ( 5, 3, 3) =   2.000000
rpntbf   ( 5, 3, 4) =   2.000000
rpntbf   ( 6, 1, 1) =  0.0000000
rpntbf   ( 6, 1, 2) =   1.000000
rpntbf   ( 6, 1, 3) =  0.0000000
rpntbf   ( 6, 1, 4) =   1.000000
rpntbf   ( 6, 2, 1) =   1.000000
rpntbf   ( 6, 2, 2) =   1.000000
rpntbf   ( 6, 2, 3) =   1.000000
rpntbf   ( 6, 2, 4) =   1.000000
rpntbf   ( 6, 3, 1) =   1.000000
rpntbf   ( 6, 3, 2) =   1.000000
rpntbf   ( 6, 3, 3) =   1.000000
rpntbf   ( 6, 3, 4) =   1.000000
rpntbf   ( 7, 1, 1) =  0.0000000
rpntbf   ( 7, 1, 2) =   1.000000
rpntbf   ( 7, 1, 3) =  0.0000000
rpntbf   ( 7, 1, 4) =   1.000000
rpntbf   ( 7, 2, 1) =   1.000000
rpntbf   ( 7, 2, 2) =   1.000000
rpntbf   ( 7, 2, 3) =   1.000000
rpntbf   ( 7, 2, 4) =   1.000000
rpntbf   ( 7, 3, 1) =   1.000000
rpntbf   ( 7, 3, 2) =   1.000000
rpntbf   ( 7, 3, 3) =   1.000000
rpntbf   ( 7, 3, 4) =   1.000000
rpntf    ( 1, 1) =  0.4424857e-02
rpntf    ( 1, 2) =  0.1192085e-02
rpntf    ( 1, 3) =  0.1988845e-01
rpntf    ( 1, 4) =  0.2735616e-03
rpntf    ( 1, 5) =  0.2490744e-02
rpntf    ( 1, 6) =  0.3867646e-03
rpntf    ( 1, 7) =  0.0000000
rpntf    ( 2, 1) =  0.9891993e-03
rpntf    ( 2, 2) =  0.3740659e-02
rpntf    ( 2, 3) =  0.6033740e-02
rpntf    ( 2, 4) =  0.0000000
rpntf    ( 2, 5) =  0.0000000
rpntf    ( 2, 6) =  0.3003082e-02
rpntf    ( 2, 7) =  0.0000000
rpntf    ( 3, 1) =  0.3947125e-02
rpntf    ( 3, 2) =  0.1043600e-02
rpntf    ( 3, 3) =  0.1735219e-01
rpntf    ( 3, 4) =  0.2370867e-03
rpntf    ( 3, 5) =  0.2185003e-02
rpntf    ( 3, 6) =  0.3351959e-03
rpntf    ( 3, 7) =  0.0000000
rpntf    ( 4, 1) =  0.8866896e-03
rpntf    ( 4, 2) =  0.3329306e-02
rpntf    ( 4, 3) =  0.5293375e-02
rpntf    ( 4, 4) =  0.0000000
rpntf    ( 4, 5) =  0.0000000
rpntf    ( 4, 6) =  0.2653003e-02
rpntf    ( 4, 7) =  0.0000000
rpnwat   ( 1) =  0.1300000e-02
rpnwat   ( 2) =  0.1300000e-02
rpnwat   ( 3) =  0.1300000e-02
rpnwat   ( 4) =  0.1300000e-02
rpnx     ( 1, 1) =  0.1890000e-01
rpnx     ( 1, 2) =  0.1890000e-01
rpnx     ( 1, 3) =  0.1300000e-02
rpnx     ( 2, 1) =  0.9044999e-02
rpnx     ( 2, 2) =  0.9044999e-02
rpnx     ( 2, 3) =  0.1300000e-02
rpnx     ( 3, 1) =  0.1638000e-01
rpnx     ( 3, 2) =  0.1638000e-01
rpnx     ( 3, 3) =  0.1300000e-02
rpnx     ( 4, 1) =  0.7839000e-02
rpnx     ( 4, 2) =  0.7839000e-02
rpnx     ( 4, 3) =  0.1300000e-02
rpq              =  0.1883991e-01
rpqb     ( 1) =  0.3200662e-02
rpqb     ( 2) =  0.1382042e-01
rpqb     ( 3) =  0.1818830e-02
rpqbf    ( 1, 1) =  0.1730046e-02
rpqbf    ( 1, 2) =  0.6583974e-03
rpqbf    ( 1, 3) =  0.8122187e-03
rpqbf    ( 1, 4) =  0.0000000
rpqbf    ( 1, 5) =  0.0000000
rpqbf    ( 1, 6) =  0.0000000
rpqbf    ( 1, 7) =  0.0000000
rpqbf    ( 2, 1) =  0.2196367e-03
rpqbf    ( 2, 2) =  0.1620289e-02
rpqbf    ( 2, 3) =  0.9892412e-02
rpqbf    ( 2, 4) =  0.9786086e-04
rpqbf    ( 2, 5) =  0.6184901e-03
rpqbf    ( 2, 6) =  0.1371727e-02
rpqbf    ( 2, 7) =  0.0000000
rpqbf    ( 3, 1) =  0.4984447e-03
rpqbf    ( 3, 2) =  0.4541372e-03
rpqbf    ( 3, 3) =  0.6136983e-03
rpqbf    ( 3, 4) =  0.0000000
rpqbf    ( 3, 5) =  0.6254330e-04
rpqbf    ( 3, 6) =  0.1900061e-03
rpqbf    ( 3, 7) =  0.0000000
rpqf     ( 1) =  0.2448127e-02
rpqf     ( 2) =  0.2732824e-02
rpqf     ( 3) =  0.1131833e-01
rpqf     ( 4) =  0.9786086e-04
rpqf     ( 5) =  0.6810334e-03
rpqf     ( 6) =  0.1561733e-02
rpqf     ( 7) =  0.0000000
rpqhms   ( 1) =   105.0000
rpqhms   ( 2) =   67.00000
rpqhms   ( 3) =   105.0000
rpqhms   ( 4) =   67.00000
rpqhms   ( 5) =   81.94981
rpqins   ( 1) =   1.000000
rpqins   ( 2) =   1.000000
rpqins   ( 3) =   1.000000
rpqins   ( 4) =   1.000000
rpqlac   ( 1) =   36.72983
rpqlac   ( 2) =   23.43713
rpqlac   ( 3) =   36.72983
rpqlac   ( 4) =   23.43713
rpqlac   ( 5) =   28.66669
rpqt     ( 1) =  0.2836957e-01
rpqt     ( 2) =  0.1394339e-01
rpqt     ( 3) =  0.2481678e-01
rpqt     ( 4) =  0.1233396e-01
rpqtb    ( 1, 1) =  0.6087598e-02
rpqtb    ( 1, 2) =  0.2036888e-01
rpqtb    ( 1, 3) =  0.1913099e-02
rpqtb    ( 2, 1) =  0.1583238e-02
rpqtb    ( 2, 2) =  0.1060168e-01
rpqtb    ( 2, 3) =  0.1758472e-02
rpqtb    ( 3, 1) =  0.5334109e-02
rpqtb    ( 3, 2) =  0.1757341e-01
rpqtb    ( 3, 3) =  0.1909262e-02
rpqtb    ( 4, 1) =  0.1425771e-02
rpqtb    ( 4, 2) =  0.9147370e-02
rpqtb    ( 4, 3) =  0.1760819e-02
rpqtbf   ( 1, 1, 1) =   1.000000
rpqtbf   ( 1, 1, 2) =   1.000000
rpqtbf   ( 1, 1, 3) =   1.000000
rpqtbf   ( 1, 1, 4) =   1.000000
rpqtbf   ( 1, 2, 1) =   1.000000
rpqtbf   ( 1, 2, 2) =   1.000000
rpqtbf   ( 1, 2, 3) =   1.000000
rpqtbf   ( 1, 2, 4) =   1.000000
rpqtbf   ( 1, 3, 1) =   1.000000
rpqtbf   ( 1, 3, 2) =   1.000000
```

```
rpqtbf  ( 1, 3, 3) =  1.000000
rpqtbf  ( 1, 3, 4) =  1.000000
rpqtbf  ( 2, 1, 1) =  1.350000
rpqtbf  ( 2, 1, 2) =  1.350000
rpqtbf  ( 2, 1, 3) =  1.350000
rpqtbf  ( 2, 1, 4) =  1.350000
rpqtbf  ( 2, 2, 1) =  1.350000
rpqtbf  ( 2, 2, 2) =  1.350000
rpqtbf  ( 2, 2, 3) =  1.350000
rpqtbf  ( 2, 2, 4) =  1.350000
rpqtbf  ( 2, 3, 1) =  1.420000
rpqtbf  ( 2, 3, 2) =  1.420000
rpqtbf  ( 2, 3, 3) =  1.420000
rpqtbf  ( 2, 3, 4) =  1.420000
rpqtbf  ( 3, 1, 1) =  1.550000
rpqtbf  ( 3, 1, 2) =  1.550000
rpqtbf  ( 3, 1, 3) =  1.550000
rpqtbf  ( 3, 1, 4) =  1.550000
rpqtbf  ( 3, 2, 1) =  1.550000
rpqtbf  ( 3, 2, 2) =  1.650000
rpqtbf  ( 3, 2, 3) =  1.550000
rpqtbf  ( 3, 2, 4) =  1.650000
rpqtbf  ( 3, 3, 1) =  2.850000
rpqtbf  ( 3, 3, 2) =  2.000000
rpqtbf  ( 3, 3, 3) =  2.850000
rpqtbf  ( 3, 3, 4) =  2.000000
rpqtbf  ( 4, 1, 1) = 0.0000000
rpqtbf  ( 4, 1, 2) = 0.0000000
rpqtbf  ( 4, 1, 3) = 0.0000000
rpqtbf  ( 4, 1, 4) = 0.0000000
rpqtbf  ( 4, 2, 1) =  1.450000
rpqtbf  ( 4, 2, 2) =  1.450000
rpqtbf  ( 4, 2, 3) =  1.450000
rpqtbf  ( 4, 2, 4) =  1.450000
rpqtbf  ( 4, 3, 1) =  4.000000
rpqtbf  ( 4, 3, 2) =  3.000000
rpqtbf  ( 4, 3, 3) =  4.000000
rpqtbf  ( 4, 3, 4) =  3.000000
rpqtbf  ( 5, 1, 1) = 0.0000000
rpqtbf  ( 5, 1, 2) = 0.0000000
rpqtbf  ( 5, 1, 3) = 0.0000000
rpqtbf  ( 5, 1, 4) = 0.0000000
rpqtbf  ( 5, 2, 1) =  1.500000
rpqtbf  ( 5, 2, 2) =  1.500000
rpqtbf  ( 5, 2, 3) =  1.500000
rpqtbf  ( 5, 2, 4) =  1.500000
rpqtbf  ( 5, 3, 1) =  2.000000
rpqtbf  ( 5, 3, 2) =  2.000000
rpqtbf  ( 5, 3, 3) =  2.000000
rpqtbf  ( 5, 3, 4) =  2.000000
rpqtbf  ( 6, 1, 1) = 0.0000000
rpqtbf  ( 6, 1, 2) =  1.000000
rpqtbf  ( 6, 1, 3) = 0.0000000
rpqtbf  ( 6, 1, 4) =  1.000000
rpqtbf  ( 6, 2, 1) =  1.000000
rpqtbf  ( 6, 2, 2) =  1.000000
rpqtbf  ( 6, 2, 3) =  1.000000
rpqtbf  ( 6, 2, 4) =  1.000000
rpqtbf  ( 6, 3, 1) =  1.000000
rpqtbf  ( 6, 3, 2) =  1.000000
rpqtbf  ( 6, 3, 3) =  1.000000
rpqtbf  ( 6, 3, 4) =  1.000000
rpqtbf  ( 7, 1, 1) = 0.0000000
rpqtbf  ( 7, 1, 2) =  1.000000
rpqtbf  ( 7, 1, 3) = 0.0000000
rpqtbf  ( 7, 1, 4) =  1.000000
rpqtbf  ( 7, 2, 1) =  1.000000
rpqtbf  ( 7, 2, 2) =  1.000000
rpqtbf  ( 7, 2, 3) =  1.000000
rpqtbf  ( 7, 2, 4) =  1.000000
rpqtbf  ( 7, 3, 1) =  1.000000
rpqtbf  ( 7, 3, 2) =  1.000000
rpqtbf  ( 7, 3, 3) =  1.000000
rpqtbf  ( 7, 3, 4) =  1.000000
rpqtf   ( 1, 1) = 0.4704715e-02
rpqtf   ( 1, 2) = 0.9580847e-03
rpqtf   ( 1, 3) = 0.2040779e-01
rpqtf   ( 1, 4) = 0.2660244e-03
rpqtf   ( 1, 5) = 0.1852844e-02
rpqtf   ( 1, 6) = 0.1801167e-03
rpqtf   ( 1, 7) = 0.0000000
rpqtf   ( 2, 1) = 0.1151228e-02
rpqtf   ( 2, 2) = 0.3969127e-02
rpqtf   ( 2, 3) = 0.6322280e-02
rpqtf   ( 2, 4) = 0.0000000
rpqtf   ( 2, 5) = 0.0000000
rpqtf   ( 2, 6) = 0.2500749e-02
rpqtf   ( 2, 7) = 0.0000000
rpqtf   ( 3, 1) = 0.4210167e-02
rpqtf   ( 3, 2) = 0.8165365e-03
rpqtf   ( 3, 3) = 0.1780934e-01
rpqtf   ( 3, 4) = 0.2308676e-03
rpqtf   ( 3, 5) = 0.1605208e-02
rpqtf   ( 3, 6) = 0.1446616e-03
rpqtf   ( 3, 7) = 0.0000000
rpqtf   ( 4, 1) = 0.1041046e-02
rpqtf   ( 4, 2) = 0.3563966e-02
rpqtf   ( 4, 3) = 0.5563848e-02
rpqtf   ( 4, 4) = 0.0000000
rpqtf   ( 4, 5) = 0.0000000
rpqtf   ( 4, 6) = 0.2165100e-02
rpqtf   ( 4, 7) = 0.0000000
rpqwat  ( 1) = 0.1300000e-02
rpqwat  ( 2) = 0.1300000e-02
rpqwat  ( 3) = 0.1300000e-02
rpqwat  ( 4) = 0.1300000e-02
rpqx    ( 1, 1) = 0.1890000e-01
rpqx    ( 1, 2) = 0.1890000e-01
rpqx    ( 1, 3) = 0.1300000e-02
rpqx    ( 2, 1) = 0.9044999e-02
rpqx    ( 2, 2) = 0.9044999e-02
rpqx    ( 2, 3) = 0.1300000e-02
rpqx    ( 3, 1) = 0.1638000e-01
rpqx    ( 3, 2) = 0.1638000e-01
rpqx    ( 3, 3) = 0.1300000e-02
rpqx    ( 4, 1) = 0.7839000e-02
rpqx    ( 4, 2) = 0.7839000e-02
rpqx    ( 4, 3) = 0.1300000e-02
rpsk    ( 1) = 0.2150000e-03
rpsk    ( 2) = 0.4300000e-03
rpsk    ( 3) = 0.6450000e-03
rpsk    ( 4) = 0.3440000e-03
rpsk    ( 5) = 0.4300000e-03
rpsk    ( 6) = 0.6010000e-03
rpsk    ( 7) = 0.3000000e-04
rpsk    ( 8) = 0.1200000e-03
rpsk    ( 9) = 0.2580000e-03
rpsk    ( 10) = 0.1300000e-03
rpsk    ( 11) = 0.1710000e-03
rpsk    ( 12) = 0.1720000e-03
rpsk    ( 13) = 0.1100000e-02
rpsk    ( 14) = 0.7000000e-03
rptemp  ( 1) =  1.000000
rptemp  ( 2) =  1.000000
rptemp  ( 3) =  1.000000
rptemp  ( 4) =  1.000000
rpx          = 0.1362418e-01
rpxb    ( 1) = 0.7724416e-02
rpxb    ( 2) = 0.4578853e-02
rpxb    ( 3) = 0.1320907e-02
rpxbf   ( 1, 1) = 0.8164896e-03
rpxbf   ( 1, 2) = 0.1039076e-02
rpxbf   ( 1, 3) = 0.2329004e-02
rpxbf   ( 1, 4) = 0.1645240e-02
rpxbf   ( 1, 5) = 0.1894607e-02
rpxbf   ( 1, 6) = 0.0000000
rpxbf   ( 1, 7) = 0.0000000
rpxbf   ( 2, 1) = 0.1564600e-03
rpxbf   ( 2, 2) = 0.3260423e-03
rpxbf   ( 2, 3) = 0.2227833e-02
rpxbf   ( 2, 4) = 0.7257824e-03
rpxbf   ( 2, 5) = 0.7764138e-03
rpxbf   ( 2, 6) = 0.3663216e-03
rpxbf   ( 2, 7) = 0.0000000
rpxbf   ( 3, 1) = 0.7189184e-03
rpxbf   ( 3, 2) = 0.1818269e-03
rpxbf   ( 3, 3) = 0.2354070e-03
rpxbf   ( 3, 4) = 0.6661064e-04
rpxbf   ( 3, 5) = 0.5817838e-04
rpxbf   ( 3, 6) = 0.5996561e-04
rpxbf   ( 3, 7) = 0.0000000
rpxf    ( 1) = 0.1691868e-02
rpxf    ( 2) = 0.1546945e-02
rpxf    ( 3) = 0.4792244e-02
rpxf    ( 4) = 0.2437633e-02
rpxf    ( 5) = 0.2729199e-02
rpxf    ( 6) = 0.4262873e-03
rpxf    ( 7) = 0.0000000
```

```
rpxhms  (  1) =   83.00000
rpxhms  (  2) =   53.00000
rpxhms  (  3) =   83.00000
rpxhms  (  4) =   53.00000
rpxhms  (  5) =   66.28452
rpxins  (  1) =   1.000000
rpxins  (  2) =   1.000000
rpxins  (  3) =   1.000000
rpxins  (  4) =   1.000000
rpxlac  (  1) =   29.03406
rpxlac  (  2) =   18.53982
rpxlac  (  3) =   29.03406
rpxlac  (  4) =   18.53982
rpxlac  (  5) =   23.18685
rpxt    (  1) =  0.2032120e-01
rpxt    (  2) =  0.9599463e-02
rpxt    (  3) =  0.1784413e-01
rpxt    (  4) =  0.8478850e-02
rpxtb   (  1,  1) =   0.9729049e-02
rpxtb   (  1,  2) =   0.9135624e-02
rpxtb   (  1,  3) =   0.1456525e-02
rpxtb   (  2,  1) =   0.6844102e-02
rpxtb   (  2,  2) =   0.1560493e-02
rpxtb   (  2,  3) =   0.1194868e-02
rpxtb   (  3,  1) =   0.8431843e-02
rpxtb   (  3,  2) =   0.7917541e-02
rpxtb   (  3,  3) =   0.1494746e-02
rpxtb   (  4,  1) =   0.5931555e-02
rpxtb   (  4,  2) =   0.1352427e-02
rpxtb   (  4,  3) =   0.1194868e-02
rpxtbf  (  1,  1,  1) =   1.000000
rpxtbf  (  1,  1,  2) =   1.000000
rpxtbf  (  1,  1,  3) =   1.000000
rpxtbf  (  1,  1,  4) =   1.000000
rpxtbf  (  1,  2,  1) =   1.000000
rpxtbf  (  1,  2,  2) =   1.000000
rpxtbf  (  1,  2,  3) =   1.000000
rpxtbf  (  1,  2,  4) =   1.000000
rpxtbf  (  1,  3,  1) =   1.000000
rpxtbf  (  1,  3,  2) =   1.000000
rpxtbf  (  1,  3,  3) =   1.000000
rpxtbf  (  1,  3,  4) =   1.000000
rpxtbf  (  2,  1,  1) =   1.380000
rpxtbf  (  2,  1,  2) =   1.380000
rpxtbf  (  2,  1,  3) =   1.380000
rpxtbf  (  2,  1,  4) =   1.380000
rpxtbf  (  2,  2,  1) =   1.380000
rpxtbf  (  2,  2,  2) =   1.330000
rpxtbf  (  2,  2,  3) =   1.380000
rpxtbf  (  2,  2,  4) =   1.330000
rpxtbf  (  2,  3,  1) =   1.420000
rpxtbf  (  2,  3,  2) =   1.420000
rpxtbf  (  2,  3,  3) =   1.420000
rpxtbf  (  2,  3,  4) =   1.420000
rpxtbf  (  3,  1,  1) =   1.540000
rpxtbf  (  3,  1,  2) =   1.540000
rpxtbf  (  3,  1,  3) =   1.540000
rpxtbf  (  3,  1,  4) =   1.540000
rpxtbf  (  3,  2,  1) =   1.540000
rpxtbf  (  3,  2,  2) =   1.420000
rpxtbf  (  3,  2,  3) =   1.540000
rpxtbf  (  3,  2,  4) =   1.420000
rpxtbf  (  3,  3,  1) =   2.580000
rpxtbf  (  3,  3,  2) =   1.420000
rpxtbf  (  3,  3,  3) =   2.850000
rpxtbf  (  3,  3,  4) =   1.420000
rpxtbf  (  4,  1,  1) =  0.8300000
rpxtbf  (  4,  1,  2) =  0.8300000
rpxtbf  (  4,  1,  3) =  0.8300000
rpxtbf  (  4,  1,  4) =  0.8300000
rpxtbf  (  4,  2,  1) =   1.660000
rpxtbf  (  4,  2,  2) =   1.660000
rpxtbf  (  4,  2,  3) =   1.660000
rpxtbf  (  4,  2,  4) =   1.660000
rpxtbf  (  4,  3,  1) =   4.000000
rpxtbf  (  4,  3,  2) =   3.000000
rpxtbf  (  4,  3,  3) =   4.000000
rpxtbf  (  4,  3,  4) =   3.000000
rpxtbf  (  5,  1,  1) =   1.000000
rpxtbf  (  5,  1,  2) =   1.000000
rpxtbf  (  5,  1,  3) =   1.000000
rpxtbf  (  5,  1,  4) =   1.000000
rpxtbf  (  5,  2,  1) =   2.000000
rpxtbf  (  5,  2,  2) =   2.000000
rpxtbf  (  5,  2,  3) =   2.000000
rpxtbf  (  5,  2,  4) =   2.000000
rpxtbf  (  5,  3,  1) =   2.000000
rpxtbf  (  5,  3,  2) =   2.000000
rpxtbf  (  5,  3,  3) =   2.000000
rpxtbf  (  5,  3,  4) =   2.000000
rpxtbf  (  6,  1,  1) =   1.000000
rpxtbf  (  6,  1,  2) =   1.000000
rpxtbf  (  6,  1,  3) =   1.000000
rpxtbf  (  6,  1,  4) =   1.000000
rpxtbf  (  6,  2,  1) =   1.000000
rpxtbf  (  6,  2,  2) =   1.000000
rpxtbf  (  6,  2,  3) =   1.000000
rpxtbf  (  6,  2,  4) =   1.000000
rpxtbf  (  6,  3,  1) =   1.000000
rpxtbf  (  6,  3,  2) =   1.000000
rpxtbf  (  6,  3,  3) =   1.000000
rpxtbf  (  6,  3,  4) =   1.000000
rpxtbf  (  7,  1,  1) =   1.000000
rpxtbf  (  7,  1,  2) =   1.000000
rpxtbf  (  7,  1,  3) =   1.000000
rpxtbf  (  7,  1,  4) =   1.000000
rpxtbf  (  7,  2,  1) =   1.000000
rpxtbf  (  7,  2,  2) =   1.000000
rpxtbf  (  7,  2,  3) =   1.000000
rpxtbf  (  7,  2,  4) =   1.000000
rpxtbf  (  7,  3,  1) =   1.000000
rpxtbf  (  7,  3,  2) =   1.000000
rpxtbf  (  7,  3,  3) =   1.000000
rpxtbf  (  7,  3,  4) =   1.000000
rpxtf   (  1,  1) =  0.1720978e-02
rpxtf   (  1,  2) =  0.1236475e-02
rpxtf   (  1,  3) =  0.6978035e-02
rpxtf   (  1,  4) =  0.3559957e-02
rpxtf   (  1,  5) =  0.6427034e-02
rpxtf   (  1,  6) =  0.3987189e-03
rpxtf   (  1,  7) =  0.0000000
rpxtf   (  2,  1) =  0.1748354e-02
rpxtf   (  2,  2) =  0.1888789e-02
rpxtf   (  2,  3) =  0.3504185e-02
rpxtf   (  2,  4) =  0.1787173e-02
rpxtf   (  2,  5) =  0.1939838e-03
rpxtf   (  2,  6) =  0.4769787e-03
rpxtf   (  2,  7) =  0.0000000
rpxtf   (  3,  1) =  0.1582882e-02
rpxtf   (  3,  2) =  0.1084049e-02
rpxtf   (  3,  3) =  0.6134549e-02
rpxtf   (  3,  4) =  0.3105337e-02
rpxtf   (  3,  5) =  0.5587614e-02
rpxtf   (  3,  6) =  0.3497002e-03
rpxtf   (  3,  7) =  0.0000000
rpxtf   (  4,  1) =  0.1614663e-02
rpxtf   (  4,  2) =  0.1670577e-02
rpxtf   (  4,  3) =  0.3052157e-02
rpxtf   (  4,  4) =  0.1548896e-02
rpxtf   (  4,  5) =  0.1681193e-03
rpxtf   (  4,  6) =  0.4244380e-03
rpxtf   (  4,  7) =  0.0000000
rpxwat  (  1) =  0.1300000e-02
rpxwat  (  2) =  0.1300000e-02
rpxwat  (  3) =  0.1300000e-02
rpxwat  (  4) =  0.1300000e-02
rpxx    (  1,  1) =  0.1494000e-01
rpxx    (  1,  2) =  0.1494000e-01
rpxx    (  1,  3) =  0.1300000e-02
rpxx    (  2,  1) =  0.7155000e-02
rpxx    (  2,  2) =  0.7155000e-02
rpxx    (  2,  3) =  0.1300000e-02
rpxx    (  3,  1) =  0.1294800e-01
rpxx    (  3,  2) =  0.1294800e-01
rpxx    (  3,  3) =  0.1300000e-02
rpxx    (  4,  1) =  0.6201000e-02
rpxx    (  4,  2) =  0.6201000e-02
rpxx    (  4,  3) =  0.1300000e-02
rpz           =  0.1368202e-01
rpzb    (  1) =  0.7010751e-02
rpzb    (  2) =  0.5306336e-02
rpzb    (  3) =  0.1364937e-02
rpzbf   (  1,  1) =  0.8951218e-03
rpzbf   (  1,  2) =  0.1012715e-02
rpzbf   (  1,  3) =  0.2218843e-02
rpzbf   (  1,  4) =  0.1335360e-02
rpzbf   (  1,  5) =  0.1548712e-02
rpzbf   (  1,  6) =  0.0000000
```

```
rpzbf   ( 1, 7) = 0.0000000
rpzbf   ( 2, 1) = 0.1623045e-03
rpzbf   ( 2, 2) = 0.4313983e-03
rpzbf   ( 2, 3) = 0.2833677e-02
rpzbf   ( 2, 4) = 0.6681892e-03
rpzbf   ( 2, 5) = 0.7603301e-03
rpzbf   ( 2, 6) = 0.4504372e-03
rpzbf   ( 2, 7) = 0.0000000
rpzbf   ( 3, 1) = 0.6967826e-03
rpzbf   ( 3, 2) = 0.2091671e-03
rpzbf   ( 3, 3) = 0.2674259e-03
rpzbf   ( 3, 4) = 0.5992287e-04
rpzbf   ( 3, 5) = 0.5861662e-04
rpzbf   ( 3, 6) = 0.7302180e-04
rpzbf   ( 3, 7) = 0.0000000
rpzf    ( 1) = 0.1754209e-02
rpzf    ( 2) = 0.1653280e-02
rpzf    ( 3) = 0.5319946e-02
rpzf    ( 4) = 0.2063472e-02
rpzf    ( 5) = 0.2367658e-02
rpzf    ( 6) = 0.5234590e-03
rpzf    ( 7) = 0.0000000
rpzhms  ( 1) = 85.37941
rpzhms  ( 2) = 54.54913
rpzhms  ( 3) = 84.72340
rpzhms  ( 4) = 54.19617
rpzhms  ( 5) = 67.85733
rpzlac  ( 1) = 29.86639
rpzlac  ( 2) = 19.08172
rpzlac  ( 3) = 29.63691
rpzlac  ( 4) = 18.95825
rpzlac  ( 5) = 23.73703
rpzt    ( 1) = 0.2036788e-01
rpzt    ( 2) = 0.9747915e-02
rpzt    ( 3) = 0.1785902e-01
rpzt    ( 4) = 0.8578257e-02
rpztb   ( 1, 1) = 0.8737752e-02
rpztb   ( 1, 2) = 0.1012775e-01
rpztb   ( 1, 3) = 0.1502372e-02
rpztb   ( 2, 1) = 0.6165995e-02
rpztb   ( 2, 2) = 0.2334369e-02
rpztb   ( 2, 3) = 0.1247551e-02
rpztb   ( 3, 1) = 0.7801557e-02
rpztb   ( 3, 2) = 0.8530241e-02
rpztb   ( 3, 3) = 0.1527218e-02
rpztb   ( 4, 1) = 0.5479711e-02
rpztb   ( 4, 2) = 0.1863063e-02
rpztb   ( 4, 3) = 0.1235483e-02
rpztbf  ( 1, 1, 1) = 0.1536829e-01
rpztbf  ( 1, 1, 2) = 0.7364133e-02
rpztbf  ( 1, 1, 3) = 0.1321685e-01
rpztbf  ( 1, 1, 4) = 0.6340952e-02
rpztbf  ( 1, 2, 1) = 0.1536829e-01
rpztbf  ( 1, 2, 2) = 0.7364133e-02
rpztbf  ( 1, 2, 3) = 0.1321685e-01
rpztbf  ( 1, 2, 4) = 0.6340952e-02

rpztbf  ( 1, 3, 1) = 0.1300000e-02
rpztbf  ( 1, 3, 2) = 0.1300000e-02
rpztbf  ( 1, 3, 3) = 0.1300000e-02
rpztbf  ( 1, 3, 4) = 0.1300000e-02
rpztbf  ( 2, 1, 1) = 0.2114692e-01
rpztbf  ( 2, 1, 2) = 0.1013248e-01
rpztbf  ( 2, 1, 3) = 0.1820076e-01
rpztbf  ( 2, 1, 4) = 0.8730420e-02
rpztbf  ( 2, 2, 1) = 0.2114692e-01
rpztbf  ( 2, 2, 2) = 0.9814315e-02
rpztbf  ( 2, 2, 3) = 0.1820076e-01
rpztbf  ( 2, 2, 4) = 0.8446861e-02
rpztbf  ( 2, 3, 1) = 0.1846000e-02
rpztbf  ( 2, 3, 2) = 0.1846000e-02
rpztbf  ( 2, 3, 3) = 0.1846000e-02
rpztbf  ( 2, 3, 4) = 0.1846000e-02
rpztbf  ( 3, 1, 1) = 0.2368761e-01
rpztbf  ( 3, 1, 2) = 0.1135077e-01
rpztbf  ( 3, 1, 3) = 0.2036678e-01
rpztbf  ( 3, 1, 4) = 0.9771763e-02
rpztbf  ( 3, 2, 1) = 0.2368761e-01
rpztbf  ( 3, 2, 2) = 0.1068727e-01
rpztbf  ( 3, 2, 3) = 0.2036678e-01
rpztbf  ( 3, 2, 4) = 0.9158197e-02
rpztbf  ( 3, 3, 1) = 0.3391962e-02
rpztbf  ( 3, 3, 2) = 0.1929432e-02
rpztbf  ( 3, 3, 3) = 0.3705000e-02
rpztbf  ( 3, 3, 4) = 0.1910422e-02
rpztbf  ( 4, 1, 1) = 0.1105906e-01
rpztbf  ( 4, 1, 2) = 0.5281521e-02
rpztbf  ( 4, 1, 3) = 0.9904972e-02
rpztbf  ( 4, 1, 4) = 0.4707085e-02
rpztbf  ( 4, 2, 1) = 0.2508210e-01
rpztbf  ( 4, 2, 2) = 0.1201428e-01
rpztbf  ( 4, 2, 3) = 0.2167051e-01
rpztbf  ( 4, 2, 4) = 0.1038533e-01
rpztbf  ( 4, 3, 1) = 0.5200000e-02
rpztbf  ( 4, 3, 2) = 0.3900000e-02
rpztbf  ( 4, 3, 3) = 0.5200000e-02
rpztbf  ( 4, 3, 4) = 0.3900001e-02
rpztbf  ( 5, 1, 1) = 0.1332416e-01
rpztbf  ( 5, 1, 2) = 0.6363278e-02
rpztbf  ( 5, 1, 3) = 0.1193370e-01
rpztbf  ( 5, 1, 4) = 0.5671186e-02
rpztbf  ( 5, 2, 1) = 0.2971452e-01
rpztbf  ( 5, 2, 2) = 0.1422784e-01
rpztbf  ( 5, 2, 3) = 0.2579213e-01
rpztbf  ( 5, 2, 4) = 0.1234702e-01
rpztbf  ( 5, 3, 1) = 0.2600000e-02
rpztbf  ( 5, 3, 2) = 0.2600000e-02
rpztbf  ( 5, 3, 3) = 0.2600000e-02
rpztbf  ( 5, 3, 4) = 0.2600000e-02
rpztbf  ( 6, 1, 1) = 0.1332416e-01
rpztbf  ( 6, 1, 2) = 0.7364133e-02
rpztbf  ( 6, 1, 3) = 0.1193370e-01
rpztbf  ( 6, 1, 4) = 0.6340952e-02

rpztbf  ( 6, 2, 1) = 0.1536829e-01
rpztbf  ( 6, 2, 2) = 0.7364133e-02
rpztbf  ( 6, 2, 3) = 0.1321685e-01
rpztbf  ( 6, 2, 4) = 0.6340952e-02
rpztbf  ( 6, 3, 1) = 0.1300000e-02
rpztbf  ( 6, 3, 2) = 0.1300000e-02
rpztbf  ( 6, 3, 3) = 0.1300000e-02
rpztbf  ( 6, 3, 4) = 0.1300000e-02
rpztbf  ( 7, 1, 1) = 0.0000000
rpztbf  ( 7, 1, 2) = 0.0000000
rpztbf  ( 7, 1, 3) = 0.0000000
rpztbf  ( 7, 1, 4) = 0.0000000
rpztbf  ( 7, 2, 1) = 0.0000000
rpztbf  ( 7, 2, 2) = 0.0000000
rpztbf  ( 7, 2, 3) = 0.0000000
rpztbf  ( 7, 2, 4) = 0.0000000
rpztbf  ( 7, 3, 1) = 0.0000000
rpztbf  ( 7, 3, 2) = 0.0000000
rpztbf  ( 7, 3, 3) = 0.0000000
rpztbf  ( 7, 3, 4) = 0.0000000
rpztf   ( 1, 1) = 0.1993157e-02
rpztf   ( 1, 2) = 0.1216098e-02
rpztf   ( 1, 3) = 0.8195179e-02
rpztf   ( 1, 4) = 0.3070231e-02
rpztf   ( 1, 5) = 0.5512378e-02
rpztf   ( 1, 6) = 0.3808353e-03
rpztf   ( 1, 7) = 0.0000000
rpztf   ( 2, 1) = 0.1689588e-02
rpztf   ( 2, 2) = 0.2089304e-02
rpztf   ( 2, 3) = 0.3730900e-02
rpztf   ( 2, 4) = 0.1418494e-02
rpztf   ( 2, 5) = 0.1534292e-03
rpztf   ( 2, 6) = 0.6661994e-03
rpztf   ( 2, 7) = 0.0000000
rpztf   ( 3, 1) = 0.1755759e-02
rpztf   ( 3, 2) = 0.1069712e-02
rpztf   ( 3, 3) = 0.6898402e-02
rpztf   ( 3, 4) = 0.2794003e-02
rpztf   ( 3, 5) = 0.5003592e-02
rpztf   ( 3, 6) = 0.3375474e-03
rpztf   ( 3, 7) = 0.0000000
rpztf   ( 4, 1) = 0.1570534e-02
rpztf   ( 4, 2) = 0.1811563e-02
rpztf   ( 4, 3) = 0.3206718e-02
rpztf   ( 4, 4) = 0.1298935e-02
rpztf   ( 4, 5) = 0.1406183e-03
rpztf   ( 4, 6) = 0.5498888e-03
rpztf   ( 4, 7) = 0.0000000
rqqfrc  ( 1) = 0.1081549
rqqfrc  ( 2) = 0.1106528
rqqfrc  ( 3) = 0.7833632e-01
rqqfrc  ( 4) = 0.8544015e-01
rqqfrc  ( 5) = 0.1004011
rrbhom  = 1796157.
rregn   ( 1) = 1
rruhom  = 828637.0
```

```
rsbb     (  1) =   0.0000000
rsbb     (  2) =   0.0000000
rsbb     (  3) =   0.0000000
rsbbf    ( 1,  1) =   0.0000000
rsbbf    ( 1,  2) =   0.0000000
rsbbf    ( 1,  3) =   0.0000000
rsbbf    ( 1,  4) =   0.0000000
rsbbf    ( 1,  5) =   0.0000000
rsbbf    ( 1,  6) =   0.0000000
rsbbf    ( 1,  7) =   0.0000000
rsbbf    ( 2,  1) =   0.0000000
rsbbf    ( 2,  2) =   0.0000000
rsbbf    ( 2,  3) =   0.0000000
rsbbf    ( 2,  4) =   0.0000000
rsbbf    ( 2,  5) =   0.0000000
rsbbf    ( 2,  6) =   0.0000000
rsbbf    ( 2,  7) =   0.0000000
rsbbf    ( 3,  1) =   0.0000000
rsbbf    ( 3,  2) =   0.0000000
rsbbf    ( 3,  3) =   0.0000000
rsbbf    ( 3,  4) =   0.0000000
rsbbf    ( 3,  5) =   0.0000000
rsbbf    ( 3,  6) =   0.0000000
rsbbf    ( 3,  7) =   0.0000000
rsbf     (  1) =   0.0000000
rsbf     (  2) =   0.0000000
rsbf     (  3) =   0.0000000
rsbf     (  4) =   0.0000000
rsbf     (  5) =   0.0000000
rsbf     (  6) =   0.0000000
rsbf     (  7) =   0.0000000
rsbtbf   ( 1,  1,  1) =   0.0000000
rsbtbf   ( 1,  1,  2) =   0.0000000
rsbtbf   ( 1,  1,  3) =   0.0000000
rsbtbf   ( 1,  1,  4) =   0.0000000
rsbtbf   ( 1,  2,  1) =   0.0000000
rsbtbf   ( 1,  2,  2) =   0.0000000
rsbtbf   ( 1,  2,  3) =   0.0000000
rsbtbf   ( 1,  2,  4) =   0.0000000
rsbtbf   ( 1,  3,  1) =   0.0000000
rsbtbf   ( 1,  3,  2) =   0.0000000
rsbtbf   ( 1,  3,  3) =   0.0000000
rsbtbf   ( 1,  3,  4) =   0.0000000
rsbtbf   ( 2,  1,  1) =   0.0000000
rsbtbf   ( 2,  1,  2) =   0.0000000
rsbtbf   ( 2,  1,  3) =   0.0000000
rsbtbf   ( 2,  1,  4) =   0.0000000
rsbtbf   ( 2,  2,  1) =   0.0000000
rsbtbf   ( 2,  2,  2) =   0.0000000
rsbtbf   ( 2,  2,  3) =   0.0000000
rsbtbf   ( 2,  2,  4) =   0.0000000
rsbtbf   ( 2,  3,  1) =   0.0000000
rsbtbf   ( 2,  3,  2) =   0.0000000
rsbtbf   ( 2,  3,  3) =   0.0000000
rsbtbf   ( 2,  3,  4) =   0.0000000
rsbtbf   ( 3,  1,  1) =   0.0000000

rsbtbf   ( 3,  1,  2) =   0.0000000
rsbtbf   ( 3,  1,  3) =   0.0000000
rsbtbf   ( 3,  1,  4) =   0.0000000
rsbtbf   ( 3,  2,  1) =   0.0000000
rsbtbf   ( 3,  2,  2) =   0.0000000
rsbtbf   ( 3,  2,  3) =   0.0000000
rsbtbf   ( 3,  2,  4) =   0.0000000
rsbtbf   ( 3,  3,  1) =   0.0000000
rsbtbf   ( 3,  3,  2) =   0.0000000
rsbtbf   ( 3,  3,  3) =   0.0000000
rsbtbf   ( 3,  3,  4) =   0.0000000
rsbtbf   ( 4,  1,  1) =   0.0000000
rsbtbf   ( 4,  1,  2) =   0.0000000
rsbtbf   ( 4,  1,  3) =   0.0000000
rsbtbf   ( 4,  1,  4) =   0.0000000
rsbtbf   ( 4,  2,  1) =   0.0000000
rsbtbf   ( 4,  2,  2) =   0.0000000
rsbtbf   ( 4,  2,  3) =   0.0000000
rsbtbf   ( 4,  2,  4) =   0.0000000
rsbtbf   ( 4,  3,  1) =   0.0000000
rsbtbf   ( 4,  3,  2) =   0.0000000
rsbtbf   ( 4,  3,  3) =   0.0000000
rsbtbf   ( 4,  3,  4) =   0.0000000
rsbtbf   ( 5,  1,  1) =   0.0000000
rsbtbf   ( 5,  1,  2) =   0.0000000
rsbtbf   ( 5,  1,  3) =   0.0000000
rsbtbf   ( 5,  1,  4) =   0.0000000
rsbtbf   ( 5,  2,  1) =   0.0000000
rsbtbf   ( 5,  2,  2) =   0.0000000
rsbtbf   ( 5,  2,  3) =   0.0000000
rsbtbf   ( 5,  2,  4) =   0.0000000
rsbtbf   ( 5,  3,  1) =   0.0000000
rsbtbf   ( 5,  3,  2) =   0.0000000
rsbtbf   ( 5,  3,  3) =   0.0000000
rsbtbf   ( 5,  3,  4) =   0.0000000
rsbtbf   ( 6,  1,  1) =   0.0000000
rsbtbf   ( 6,  1,  2) =   0.0000000
rsbtbf   ( 6,  1,  3) =   0.0000000
rsbtbf   ( 6,  1,  4) =   0.0000000
rsbtbf   ( 6,  2,  1) =   0.0000000
rsbtbf   ( 6,  2,  2) =   0.0000000
rsbtbf   ( 6,  2,  3) =   0.0000000
rsbtbf   ( 6,  2,  4) =   0.0000000
rsbtbf   ( 6,  3,  1) =   0.0000000
rsbtbf   ( 6,  3,  2) =   0.0000000
rsbtbf   ( 6,  3,  3) =   0.0000000
rsbtbf   ( 6,  3,  4) =   0.0000000
rsbtbf   ( 7,  1,  1) =   0.0000000
rsbtbf   ( 7,  1,  2) =   0.0000000
rsbtbf   ( 7,  1,  3) =   0.0000000
rsbtbf   ( 7,  1,  4) =   0.0000000
rsbtbf   ( 7,  2,  1) =   0.0000000
rsbtbf   ( 7,  2,  2) =   0.0000000
rsbtbf   ( 7,  2,  3) =   0.0000000
rsbtbf   ( 7,  2,  4) =   0.0000000
rsbtbf   ( 7,  3,  1) =   0.0000000

rsbtbf   ( 7,  3,  2) =   0.0000000
rsbtbf   ( 7,  3,  3) =   0.0000000
rsbtbf   ( 7,  3,  4) =   0.0000000
rsbtot    =   0.0000000
rsc       =   0.1891675e-02
rscf     (  1) =   0.1471617e-02
rscf     (  2) =   0.4200583e-03
rscfrc   (  1) =   0.9142467
rscfrc   (  2) =   0.3898479
rscfrc   (  3) =   0.9684592e-01
rscfrc   (  4) =   0.7585475
rscfrc   (  5) =   0.5680376
rscfrc   (  6) =   0.1307041
rscfrc   (  7) =   0.7010282
rscfrc   (  8) =   0.2134315
rscfrc   (  9) =   0.8350470
rscfrc   ( 10) =   0.9934019
rscfrc   ( 11) =   0.2381317
rscfrc   ( 12) =   0.1700941
rscfrc   ( 13) =   0.3214383
rscfrc   ( 14) =   0.9496589e-01
rschyr   (  1) =     10.00000
rschyr   (  2) =     18.00000
rschyr   (  3) =     50.00000
rschyr   (  4) =     6.000000
rschyr   (  5) =     15.00000
rschyr   (  6) =     16.00000
rschyr   (  7) =     15.00000
rschyr   (  8) =     35.00000
rschyr   (  9) =     15.00000
rschyr   ( 10) =     15.00000
rschyr   ( 11) =     15.00000
rschyr   ( 12) =     15.00000
rschyr   ( 13) =     10.00000
rschyr   ( 14) =     10.00000
rscsat   (  1) =     1.000000
rscsat   (  2) =   0.8000000
rscsat   (  3) =   0.6000000
rscsat   (  4) =   0.9000000
rscsat   (  5) =   0.7000000
rscsat   (  6) =   0.3000000
rscsat   (  7) =   0.8000000
rscsat   (  8) =     1.000000
rscsat   (  9) =     1.000000
rscsat   ( 10) =     1.000000
rscsat   ( 11) =   0.7000000
rscsat   ( 12) =   0.5000000
rscsat   ( 13) =   0.3000000
rscsat   ( 14) =   0.1700000
rsum     (  1) =     8467.239
rsum     (  2) =     13909.27
rsum     (  3) =     27873.72
rsum     (  4) =     33290.05
rsum     (  5) =     39505.04
rsum     (  6) =     40878.98
rsum     (  7) =     40878.98
```

```
rubb    (  1) =    18402.09
rubb    (  2) =    13928.98
rubb    (  3) =    3582.652
rubbf   (  1,  1) =    2349.592
rubbf   (  1,  2) =    2658.116
rubbf   (  1,  3) =    5824.083
rubbf   (  1,  4) =    3505.056
rubbf   (  1,  5) =    4065.241
rubbf   (  1,  6) =  0.0000000
rubbf   (  1,  7) =  0.0000000
rubbf   (  2,  1) =    426.0331
rubbf   (  2,  2) =    1132.337
rubbf   (  2,  3) =    7438.449
rubbf   (  2,  4) =    1753.991
rubbf   (  2,  5) =    1995.899
rubbf   (  2,  6) =    1182.273
rubbf   (  2,  7) =  0.0000000
rubbf   (  3,  1) =    1828.923
rubbf   (  3,  2) =    549.0135
rubbf   (  3,  3) =    701.9138
rubbf   (  3,  4) =    157.2818
rubbf   (  3,  5) =    153.8573
rubbf   (  3,  6) =    191.6627
rubbf   (  3,  7) =  0.0000000
rubf    (  1) =    4604.548
rubf    (  2) =    4339.466
rubf    (  3) =    13964.45
rubf    (  4) =    5416.329
rubf    (  5) =    6214.998
rubf    (  6) =    1373.936
rubf    (  7) =  0.0000000
rubtbf  (  1,  1,  1) =    488.2398
rubtbf  (  1,  1,  2) =    1171.625
rubtbf  (  1,  1,  3) =    576.4995
rubtbf  (  1,  1,  4) =    113.2269
rubtbf  (  1,  2,  1) =    113.9078
rubtbf  (  1,  2,  2) =    155.9958
rubtbf  (  1,  2,  3) =    141.3188
rubtbf  (  1,  2,  4) =    14.81072
rubtbf  (  1,  3,  1) =    331.6328
rubtbf  (  1,  3,  2) =    915.5836
rubtbf  (  1,  3,  3) =    478.0619
rubtbf  (  1,  3,  4) =    103.6447
rubtbf  (  2,  1,  1) =    251.7650
rubtbf  (  2,  1,  2) =    1903.053
rubtbf  (  2,  1,  3) =    318.0326
rubtbf  (  2,  1,  4) =    185.2649
rubtbf  (  2,  2,  1) =    275.4991
rubtbf  (  2,  2,  2) =    469.4484
rubtbf  (  2,  2,  3) =    348.3806
rubtbf  (  2,  2,  4) =    39.00906
rubtbf  (  2,  3,  1) =    42.46941
rubtbf  (  2,  3,  2) =    401.3916
rubtbf  (  2,  3,  3) =    62.18797
rubtbf  (  2,  3,  4) =    42.96455
rubtbf  (  3,  1,  1) =    1079.737
rubtbf  (  3,  1,  2) =    3079.128
rubtbf  (  3,  1,  3) =    1362.335
rubtbf  (  3,  1,  4) =    302.8823
rubtbf  (  3,  2,  1) =    2568.666
rubtbf  (  3,  2,  2) =    1679.111
rubtbf  (  3,  2,  3) =    3041.068
rubtbf  (  3,  2,  4) =    149.6039
rubtbf  (  3,  3,  1) =    190.9819
rubtbf  (  3,  3,  2) =    195.1404
rubtbf  (  3,  3,  3) =    295.2283
rubtbf  (  3,  3,  4) =    20.56312
rubtbf  (  4,  1,  1) =    636.2359
rubtbf  (  4,  1,  2) =    1828.857
rubtbf  (  4,  1,  3) =    853.7236
rubtbf  (  4,  1,  4) =    186.2393
rubtbf  (  4,  2,  1) =    739.3468
rubtbf  (  4,  2,  2) =    54.30979
rubtbf  (  4,  2,  3) =    954.9704
rubtbf  (  4,  2,  4) =    5.364129
rubtbf  (  4,  3,  1) =    62.79938
rubtbf  (  4,  3,  2) =  0.1156448
rubtbf  (  4,  3,  3) =    94.35351
rubtbf  (  4,  3,  4) =  0.1321371e-01
rubtbf  (  5,  1,  1) =    1637.599
rubtbf  (  5,  1,  2) =    203.7023
rubtbf  (  5,  1,  3) =    2203.196
rubtbf  (  5,  1,  4) =    20.74377
rubtbf  (  5,  2,  1) =    881.9297
rubtbf  (  5,  2,  2) =  0.0000000
rubtbf  (  5,  2,  3) =    1113.969
rubtbf  (  5,  2,  4) =  0.0000000
rubtbf  (  5,  3,  1) =    62.98197
rubtbf  (  5,  3,  2) =  0.0000000
rubtbf  (  5,  3,  3) =    90.87535
rubtbf  (  5,  3,  4) =  0.0000000
rubtbf  (  6,  1,  1) =  0.0000000
rubtbf  (  6,  1,  2) =  0.0000000
rubtbf  (  6,  1,  3) =  0.0000000
rubtbf  (  6,  1,  4) =  0.0000000
rubtbf  (  6,  2,  1) =    165.4334
rubtbf  (  6,  2,  2) =    740.3920
rubtbf  (  6,  2,  3) =    210.4000
rubtbf  (  6,  2,  4) =    66.04792
rubtbf  (  6,  3,  1) =    12.98529
rubtbf  (  6,  3,  2) =    144.0968
rubtbf  (  6,  3,  3) =    19.50987
rubtbf  (  6,  3,  4) =    15.07071
rubtbf  (  7,  1,  1) =  0.0000000
rubtbf  (  7,  1,  2) =  0.0000000
rubtbf  (  7,  1,  3) =  0.0000000
rubtbf  (  7,  1,  4) =  0.0000000
rubtbf  (  7,  2,  1) =  0.0000000
rubtbf  (  7,  2,  2) =  0.0000000
rubtbf  (  7,  2,  3) =  0.0000000
rubtbf  (  7,  2,  4) =  0.0000000
rubtbf  (  7,  3,  1) =  0.0000000
rubtbf  (  7,  3,  2) =  0.0000000
rubtbf  (  7,  3,  3) =  0.0000000
rubtbf  (  7,  3,  4) =  0.0000000
rubtot  =    35913.72
rurfrc  (  1,  1,  1) =  0.5000000
rurfrc  (  1,  1,  2) =  0.5000000
rurfrc  (  1,  1,  3) =  0.5000000
rurfrc  (  1,  1,  4) =  0.5000000
rurfrc  (  1,  2,  1) =  0.5000000
rurfrc  (  1,  2,  2) =  0.5000000
rurfrc  (  1,  2,  3) =  0.5000000
rurfrc  (  1,  2,  4) =  0.5000000
rurfrc  (  1,  3,  1) =  0.5000000
rurfrc  (  1,  3,  2) =  0.5000000
rurfrc  (  1,  3,  3) =  0.5000000
rurfrc  (  1,  3,  4) =  0.5000000
rurfrc  (  2,  1,  1) =  0.5000000
rurfrc  (  2,  1,  2) =  0.5000000
rurfrc  (  2,  1,  3) =  0.5000000
rurfrc  (  2,  1,  4) =  0.5000000
rurfrc  (  2,  2,  1) =  0.5000000
rurfrc  (  2,  2,  2) =  0.5000000
rurfrc  (  2,  2,  3) =  0.5000000
rurfrc  (  2,  2,  4) =  0.5000000
rurfrc  (  2,  3,  1) =  0.5000000
rurfrc  (  2,  3,  2) =  0.5000000
rurfrc  (  2,  3,  3) =  0.5000000
rurfrc  (  2,  3,  4) =  0.5000000
rurfrc  (  3,  1,  1) =  0.5000000
rurfrc  (  3,  1,  2) =  0.5000000
rurfrc  (  3,  1,  3) =  0.5000000
rurfrc  (  3,  1,  4) =  0.5000000
rurfrc  (  3,  2,  1) =  0.5000000
rurfrc  (  3,  2,  2) =  0.5000000
rurfrc  (  3,  2,  3) =  0.5000000
rurfrc  (  3,  2,  4) =  0.5000000
rurfrc  (  3,  3,  1) =  0.5000000
rurfrc  (  3,  3,  2) =  0.5000000
rurfrc  (  3,  3,  3) =  0.5000000
rurfrc  (  3,  3,  4) =  0.5000000
rurfrc  (  4,  1,  1) =  0.5000000
rurfrc  (  4,  1,  2) =  0.5000000
rurfrc  (  4,  1,  3) =  0.5000000
rurfrc  (  4,  1,  4) =  0.5000000
rurfrc  (  4,  2,  1) =  0.5000000
rurfrc  (  4,  2,  2) =  0.5000000
rurfrc  (  4,  2,  3) =  0.5000000
rurfrc  (  4,  2,  4) =  0.5000000
rurfrc  (  4,  3,  1) =  0.5000000
rurfrc  (  4,  3,  2) =  0.5000000
rurfrc  (  4,  3,  3) =  0.5000000
rurfrc  (  4,  3,  4) =  0.5000000
rurfrc  (  5,  1,  1) =  0.5000000
rurfrc  (  5,  1,  2) =  0.5000000
rurfrc  (  5,  1,  3) =  0.5000000
rurfrc  (  5,  1,  4) =  0.5000000
```

```
rurfrc   ( 5, 2, 1) =  Ø.5000000
rurfrc   ( 5, 2, 2) =  Ø.5000000
rurfrc   ( 5, 2, 3) =  Ø.5000000
rurfrc   ( 5, 2, 4) =  Ø.5000000
rurfrc   ( 5, 3, 1) =  Ø.5000000
rurfrc   ( 5, 3, 2) =  Ø.5000000
rurfrc   ( 5, 3, 3) =  Ø.5000000
rurfrc   ( 5, 3, 4) =  Ø.5000000
rurfrc   ( 6, 1, 1) =  Ø.5000000
rurfrc   ( 6, 1, 2) =  Ø.5000000
rurfrc   ( 6, 1, 3) =  Ø.5000000
rurfrc   ( 6, 1, 4) =  Ø.5000000
rurfrc   ( 6, 2, 1) =  Ø.5000000
rurfrc   ( 6, 2, 2) =  Ø.5000000
rurfrc   ( 6, 2, 3) =  Ø.5000000
rurfrc   ( 6, 2, 4) =  Ø.5000000
rurfrc   ( 6, 3, 1) =  Ø.5000000
rurfrc   ( 6, 3, 2) =  Ø.5000000
rurfrc   ( 6, 3, 3) =  Ø.5000000
rurfrc   ( 6, 3, 4) =  Ø.5000000
rurfrc   ( 7, 1, 1) =  Ø.5000000
rurfrc   ( 7, 1, 2) =  Ø.5000000
rurfrc   ( 7, 1, 3) =  Ø.5000000
rurfrc   ( 7, 1, 4) =  Ø.5000000
rurfrc   ( 7, 2, 1) =  Ø.5000000
rurfrc   ( 7, 2, 2) =  Ø.5000000
rurfrc   ( 7, 2, 3) =  Ø.5000000
rurfrc   ( 7, 2, 4) =  Ø.5000000
rurfrc   ( 7, 3, 1) =  Ø.5000000
rurfrc   ( 7, 3, 2) =  Ø.5000000
rurfrc   ( 7, 3, 3) =  Ø.5000000
rurfrc   ( 7, 3, 4) =  Ø.5000000
rusf     (  1) =   3862.691
rusf     (  2) =   11Ø2.566
rusf     (  3) =  Ø.0000000
rusf     (  4) =  Ø.0000000
rusf     (  5) =  Ø.0000000
rusf     (  6) =  Ø.0000000
rusf     (  7) =  Ø.0000000
rusk     (  1) =   515.9374
rusk     (  2) =   44Ø.ØØ63
rusk     (  3) =   163.9594
rusk     (  4) =   684.9146
rusk     (  5) =   641.1221
rusk     (  6) =   2Ø6.1859
rusk     (  7) =   55.2Ø163
rusk     (  8) =   67.22565
rusk     (  9) =   565.4911
rusk     ( 1Ø) =   338.9718
rusk     ( 11) =   1Ø6.883Ø
rusk     ( 12) =   76.79145
rusk     ( 13) =   928.Ø8Ø2
rusk     ( 14) =   174.4861
rustot   =   4965.257
ruwcum   =   3734Ø1.7
ruwf     (  1) =   8467.239
ruwf     (  2) =   5442.Ø33
ruwf     (  3) =   13964.45
ruwf     (  4) =   5416.329
ruwf     (  5) =   6214.998
ruwf     (  6) =   1373.936
ruwf     (  7) =  Ø.ØØØØØØØ
ruwtot   =   4Ø878.98
rxxfrc   (  1) =  Ø.891845Ø
rxxfrc   (  2) =  Ø.8893471
rxxfrc   (  3) =  Ø.9216637
rxxfrc   (  4) =  Ø.91456ØØ
rxxfrc   (  5) =  Ø.8995989
rzcur    =   198Ø
rzfile   =      1
rzreg    =     11
rzzalf   =      Ø
rzzbas   =      1
rzzcnv   =      1
rzzrnw   =      1
rzzsec   =      1
```