



Nested Functions Exercises 1




Practical CGE, 2025 © cgemod 1

1



Outline


- Introduction
- Data
- Equation coding
 - \$STOP
 - Parameters
 - Equations
 - Clearing factor markets
- A Simple Experiment
 - Trade policy
- Comparing Results



Practical CGE, 2025 © cgemod 2


2





Course Library & Working Directory

- All the files for these exercises are already in your course library
 - C:\cgemod\cgemod_lib\pract_lib
- Create a New Project (File>New Project)
 - C:\cgemod\pract\smod\smodt2
- The library index file in Model Library Explorer
 - Practical CGE Library
 - Item 11 will be used

**Create a new directory smodt2, i.e.,
cgemod\pract\smod\smodt2** 

Practical CGE, 2025 © cgemod 3

3




Data



Practical CGE, 2025 © cgemod 4

4




Data

Same as for *smod_t* except ADD *ELASTX*

Review *smod_t_mod06_t2.xls*

Has all data needed for *smod_t2*


Model Specific Activity Elasticities			
Elasticities			
ELASTX	ELASTX	ELASTX	
		<i>sigmax</i>	<i>sigmava</i>
Agriculture	aagr	2	0.8
Natural Resources	anres	2	0.8
Manufacturing	amanu	2	0.8
Services	aserv	2	0.8




Values are 'place holders'

Practical CGE, 2025
© cgemod
5

5




Equation Coding



Practical CGE, 2025
© cgemod
6

6





Using \$STOP


GAMS Configuration: set to produce *smod***.gdx* and *smod***.ref* files with *F10*

Progress One ‘Baby’ Step at a Time

- Use **\$stop** to set end of code run
- Use *F10* (Run with GDY Creation) to produce updated gdx and ref files


Step 1: Check

- Data Entry
- Data adjustments and scaling
- Data Diagnostics
- Additional set assignments

Gets to starting point for model coding 

Practical CGE, 2025
© cgemod
7

7




Parameter Assignment – smod_t2

NB: Some people find it helps to assign the equations before the parameters

Details for parameter assignments are included in Appendix 2 of the technical document

<ol style="list-style-type: none"> 1. CES aggregation functions for Level 1 of production nest <ol style="list-style-type: none"> a. <code>deltax (a)</code> b. <code>ADX0 (a)</code> 2. Leontief aggregation functions for Level 1 of production nest <ol style="list-style-type: none"> a. <code>ioqintqx (a)</code> b. <code>ioqvaqx (a)</code> 3. CES aggregation functions for Level 2 of production nest <ol style="list-style-type: none"> a. <code>deltava (f, a)</code> b. <code>ADVA0 (a)</code> 4. Intermediate Input Demand <ol style="list-style-type: none"> a. <code>ioqtdqd (c, a)</code> 	<p style="color: red;">SUGGESTION: Use the <code>\$stop</code> after coding each step and use <code>F10</code> so that the output can be used to check your code.</p> <p style="color: blue;">HINTS:</p> <ul style="list-style-type: none"> • determine what each equation does before coding the parameters, • review the use of Euler’s theorem for linear homogenous functions.
--	--

TASK 

Try to derive such equations, to ensure that the `deltax` and `deltava` values sum to one & that none of the values are negative.

Practical CGE, 2025
© cgemod
8

8



cgemod

Equation Assignment – *smod_t2*

Equations needed are already declared

Transform the algebraic expressions, in the *smod_t* technical document (Appendix 2), into GAMS code.

- Derive interpretations of the parameters
 - `.ioqxcqx(a, c)`
 - `.ioqtdqd(c, a)`
 - `.ioqintqx(a)`
 - `.ioqvaqx(a)`
- Explain the derivation of the equation `PVADEF(a)`

Initialise the ‘new’ variables

Practical CGE, 2025 © cgemod 9

9

cgemod

Model Definition & Closure Conditions

Defining the Model

The new equations need to be added to the model definition.
This has been done for you in the template.

Checking the Model Closure Conditions

- New technology ‘variables’, i.e., *ADX*, *ADVA* and *ADFD*.
- ADX* and *ADVA* are variables in the code
- Revise the file `smod_t_cl_base.inc` to fix arguments in the code for *ADX*, *ADVA* and *ADFD*.

Practical CGE, 2025 © cgemod 10

10

cgemod

Check the Model

1. Check the data in the model are the intended data.
2. Check that the value for VAR WALRAS is zero.
3. Check that the basic prices (PE, PD, PM) are equal to one.
4. Check that all entries in ASAM1CHK are equal to zero
5. Check that all entries in ASAM2CHK are equal to one
6. Check the LHS values.
7. Numéraire check.

**Only after these checks have been passed
should you move on to using the model.**

Practical CGE, 2025

© cgemod

11

11

cgemod

A Simple Experiment




Practical CGE, 2025

© cgemod

12

12

cgemod




Experiment

Trade policy experiment from Module O5


Experiment File

1. Copy the file `smod_t_exp2.inc` used with `smod_t`,
2. Add it to your working directory.
3. Some changes are needed
 - i. Save the file as `smod_t2_exp2.inc`
 - ii. Use the same experiment
 - iii. Change the model named in the Solve statement
4. Make some changes to the code for saving results
 - i. Extend the set `scal` to include the additional scalar results, e.g., *ADXADJ* and extend the results reported in *levSCAL* and *pcSCAL*.
 - ii. Declare parameters results for the new variables, i.e., *lev*** and *pc***, and assign values to these parameters.


For each run of the model add the instruction `gdx=***`  ***** in the command line, where ***** identifies different runs.

Practical CGE, 2025
© cgemod
13

13




Comparing Results



Practical CGE, 2025
© cgemod
14

14





Comparing Results

Comparison of the Results


1. To compare the results across the 2 different assumptions about factor market clearing, use the GDX MERGE utility. Open the file `compare.gms` and save it as `compare**.gms`.
2. Edit the call statement to refer to the gdx files generate by each run of the model.
3. Run `compare**.gms` (make sure `compare**.gms` is the Main File before running the model). Review the merged file, `MERGE.GDX`.

EXPECTATIONS

Analyse the results


Emphasise explanation not simple reporting of the results.

1. Factor demands (*FD*)
2. Factor prices (*WF* and *WFDIST*) and factor incomes (*YH*)
3. Value added (*QVA*) and intermediate inputs (*QINT*)
4. Production structure (*QX* and *QXC*)
5. Domestic quantities (*QD* and *QQ*)
6. Household incomes (*YH*) consumption (*HEXP* & *QCD*) & welfare (*EV*)
7. etc.




Practical CGE, 2025
© cgemod
15

15



The End

Nested Functions Exercises 1



Practical CGE, 2025
© cgemod
16

16

