Agricultural Impact Analysis using GAMS
Introduction to GAMS

Bruce A. McCarl

Specialist in Applied Optimization
Professor of Agricultural Economics, Texas A&M
Principal, McCarl and Associates

mccarl@tamu.edu
mccarl@bihs.net
agrinet.tamu.edu/mccarl

409-693-5694
409-845-1706

Introduction to GAMS

GAMS Generalized Algebraic Modeling System

GAMS, the way we use it, is a language for setting up and solving mathematical programming optimization models. (GAMS can also solve simultaneous systems of equations and deal with computable general equilibrium models)

GAMS allows one to specify the structure of an optimization model, specify and calculate data that go into that model, solve that model, do report writing on a model, and do a comparative statics analysis on that model, all in one package.

The GAMS documentation is in the documents directory and is called userguide.pdf

Introduction to GAMS

GAMS at its simplest

Suppose we wish to solve the optimization problem

$$
\begin{array}{lrcrcrcrl}
\text{Maximize} & 109(X_{corn} & \% & 90(X_{wheat} & \% & 115(X_{cotton} & & & \\
\text{subject to} & X_{corn} & \% & X_{wheat} & \% & X_{cotton} & \# & 100 & \text{(land)} \\
 & 6(X_{corn} & \% & 4(X_{wheat} & \% & 8(X_{cotton} & \# & 500 & \text{(labor)} \\
 & X_{corn} & , & X_{wheat} & , & X_{cotton} & \$ & 0 & \text{(nonnegativity)}
\end{array}
$$

The simplest GAMS formulation I can conceive of is

(file is in examples INTRO.GPR called SIMPfarm.GMS)

```
VARIABLES                 Z;
POSITIVE VARIABLES    Xcorn ,    Xwheat , Xcotton;
EQUATIONS     OBJ,  land ,  labor;
OBJ.. Z =E=
      109 * Xcorn + 90 * Xwheat + 115 * Xcotton;
land..      Xcorn +      Xwheat +      Xcotton =L= 100;
labor..   6*Xcorn +  4 * Xwheat +  8 * Xcotton =L= 500;
MODEL PROBLEM /ALL/;
SOLVE PROBLEM USING LP MAXIMIZING Z;
```

Introduction to GAMS

Dissecting the GAMS formulation

Why the new symbol Z

```
VARIABLES             Z;
POSITIVE VARIABLES    Xcorn ,    Xwheat , Xcotton;
EQUATIONS     OBJ,  land ,  labor;
OBJ.. Z =E=
      109 * Xcorn + 90 * Xwheat + 115 * Xcotton;
land..      Xcorn +      Xwheat +      Xcotton =L= 100;
labor..   6*Xcorn +  4 * Xwheat +  8 * Xcotton =L= 500;
MODEL PROBLEM /ALL/;
SOLVE PROBLEM USING LP MAXIMIZING Z;
```

GAMS requires all models to be of a special form

Namely given the model

maximize cx

It must be re written as

Maximize R

R=CX

where R is a variable unrestricted in sign named however you want it named

Introduction to GAMS

Dissecting the GAMS formulation

The VARIABLES specification

```
VARIABLES             Z;
POSITIVE VARIABLES    Xcorn ,    Xwheat , Xcotton;
EQUATIONS     OBJ,  land ,  labor;
OBJ.. Z =E=
      109 * Xcorn + 90 * Xwheat + 115 * Xcotton;
land..      Xcorn +      Xwheat +      Xcotton =L= 100;
labor..   6*Xcorn +  4 * Xwheat +  8 * Xcotton =L= 500;
MODEL PROBLEM /ALL/;
SOLVE PROBLEM USING LP MAXIMIZING Z;
```

GAMS requires the variables in each problem to be identified. In the reference problem we have variables

Z, Xcorn , Xwheat , Xcotton

The POSITIVE modifier on the variable definition means that these variables are nonnegative ie Xcorn , Xwheat , Xcotton

The use of the word VARIABLES without the POSITIVE modifier (note several other modifiers are possible) means that the named variables are unrestricted in sign. Z above

There always must be at least one of these in every problem which is the objective function variable.

Introduction to GAMS

Dissecting the GAMS formulation
The EQUATIONS specification

```
VARIABLES             Z;
POSITIVE VARIABLES    Xcorn ,    Xwheat , Xcotton;
EQUATIONS     OBJ,   land ,  labor;
OBJ.. Z =E=
      109 * Xcorn + 90 * Xwheat + 115 * Xcotton;
land..      Xcorn +      Xwheat +      Xcotton =L= 100;
labor..   6*Xcorn +  4 * Xwheat +  8 * Xcotton =L= 500;
MODEL PROBLEM /ALL/;
SOLVE PROBLEM USING LP MAXIMIZING Z;
```

GAMS requires that the modeler name each equation which is active in the optimization model. Later each equation is specified using the .. notation

In this formulation the equations are named in the EQUATION line

OBJ is the name for the objective function equation Land is the name for the first constraint equation
Labor the name for the second constraint equation

The objective function is always counted as one of the equations and must always been named.

Introduction to GAMS

Dissecting the GAMS formulation

The .. specification

```
VARIABLES             Z;
POSITIVE VARIABLES    Xcorn ,    Xwheat , Xcotton;
EQUATIONS     OBJ,  land ,  labor;
OBJ.. Z =E=
      109 * Xcorn + 90 * Xwheat + 115 * Xcotton;
land..      Xcorn +      Xwheat +      Xcotton =L= 100;
labor..   6*Xcorn +  4 * Xwheat +  8 * Xcotton =L= 500;
MODEL PROBLEM /ALL/;
SOLVE PROBLEM USING LP MAXIMIZING Z;
```

The GAMS equations specification actually consists of two parts.

The first part naming equations, was discussed on the previous page.

The second part involves specifying the exact algebraic structure of the equations. This is done using the .. notation. In this notation we give equation name followed by a .. then algebraic form of the equation in the model. This algebraic form involves use of a special syntax to tell the exact form of the equation which may be actually be an inequality.

=E= is used to indicate an equality constraint
=L= indicates a less than or equal to constraints
=G= indicates a greater than or equal to constraint.

Introduction to GAMS
Dissecting the GAMS formulation

The MODEL specification

```
VARIABLES             Z;
POSITIVE VARIABLES    Xcorn ,    Xwheat , Xcotton;
EQUATIONS     OBJ,  land ,  labor;
OBJ.. Z =E=
      109 * Xcorn + 90 * Xwheat + 115 * Xcotton;
land..      Xcorn +      Xwheat +      Xcotton =L= 100;
labor..   6*Xcorn +  4 * Xwheat +  8 * Xcotton =L= 500;
MODEL PROBLEM /ALL/;
SOLVE PROBLEM USING LP MAXIMIZING Z;
```

Once all the model structural elements have been defined than one employs a MODEL statement to identify models that will be solved. Generally I use a MODEL statement of the form above. Therein following MODEL than a name for the model is given followed by the names of the equations enclosed in slashes. Using /ALL/ includes all the equations.

One could alternatively have a model statement like that below.

MODEL FARM /obj, Land,labor/;

or omitting   CONSTRAIN1   from the model

MODEL ALTPROBLEM / obj,CONSTRAIN1/;

Introduction to GAMS

Dissecting the GAMS formulation

The SOLVE specification

```
VARIABLES             Z;
POSITIVE VARIABLES    Xcorn ,    Xwheat , Xcotton;
EQUATIONS     OBJ,  land ,  labor;
OBJ.. Z =E=
      109 * Xcorn + 90 * Xwheat + 115 * Xcotton;
land..      Xcorn +      Xwheat +       Xcotton =L= 100;
labor..   6*Xcorn +  4 * Xwheat +  8 * Xcotton =L= 500;
MODEL PROBLEM /ALL/;
SOLVE PROBLEM USING LP MAXIMIZING Z;
```

The SOLVE statement causes GAMS to use a solver to optimize the model named immediately after the SOLVE statement. That model must already have been defined in a MODEL statement.

The solve statement tells the solver to maximize or minimize a defined variable. That variable must be unrestricted in sign and is the variable we referred to above as the objective function variable.

The example statement solves a linear programming problem ("using LP").
One also can have solve statements which

solve nonlinear programs using the syntax "using nlp", mixed integer programs using the syntax "using MIP"
or a number of other forms.

Introduction to GAMS

Dissecting the GAMS formulation

The ;

```
VARIABLES             Z;
POSITIVE VARIABLES    Xcorn ,    Xwheat , Xcotton;
EQUATIONS     OBJ,  land ,  labor;
OBJ.. Z =E=
      109 * Xcorn + 90 * Xwheat + 115 * Xcotton;
land..      Xcorn +      Xwheat +      Xcotton =L= 100;
labor..   6*Xcorn +  4 * Xwheat +  8 * Xcotton =L= 500;
MODEL PROBLEM /ALL/;
SOLVE PROBLEM USING LP MAXIMIZING Z;
```

GAMS requires users to terminate each statement with a ;

Statements may be several lines long or may contain several elements.

;'s are a very important part of the syntax. Their omission often causes many syntax errors to be reported by the GAMS compiler.

Invoking GAMS

GAMS is used in two phases.

First, one uses a text editor and creates a file which contains GAMS instructions.

Second, one submits the file to GAMS which executes those instructions doing calculations, invoking the solver and creating a file of results.

Two ways to do this.

Traditional method – use a text editor set up the model then use DOS (or UNIX) instructions to run.

A newer way the GAMS IDE. Here one uses a graphical interface to run GAMS

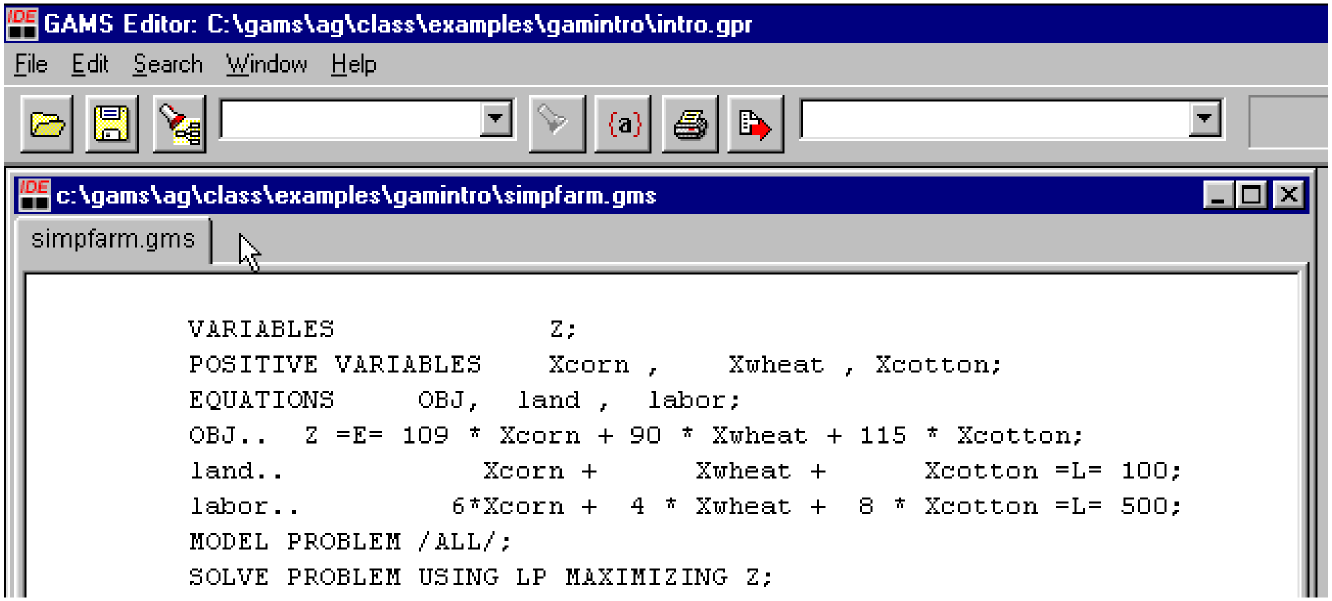
There costs and benefits of these approaches.

The IDE is much easier for simple models but currently limited to PCs.

The DOS approach can be better for multiple stage models.

Introduction to GAMS
GAMS IDE

This class will concentrate on the IDE approach.   However what you learn about setting up GAMS instructions applies equally well to either approach


GAMS Editor: C:\gams\ag\class\examples\gamintro\intro.gpr
File Edit Search Window Help
c:\gams\ag\class\examples\gamintro\simpfarm.gms
simpfarm.gms

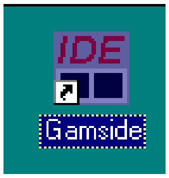
```
VARIABLES                Z;
POSITIVE VARIABLES     Xcorn ,     Xwheat , Xcotton;
EQUATIONS       OBJ,  land ,  labor;
OBJ..  Z =E= 109 * Xcorn + 90 * Xwheat + 115 * Xcotton;
land..              Xcorn +      Xwheat +      Xcotton =L= 100;
labor..           6*Xcorn +  4 * Xwheat +  8 * Xcotton =L= 500;
MODEL PROBLEM /ALL/;
SOLVE PROBLEM USING LP MAXIMIZING Z;
```

Introduction to GAMS

Using GAMS through the IDE.

Steps to using

Steps to using



1. Install on Computer

2. Click on IDE icon

3. Open an existing project or define a new project

4. Open a file

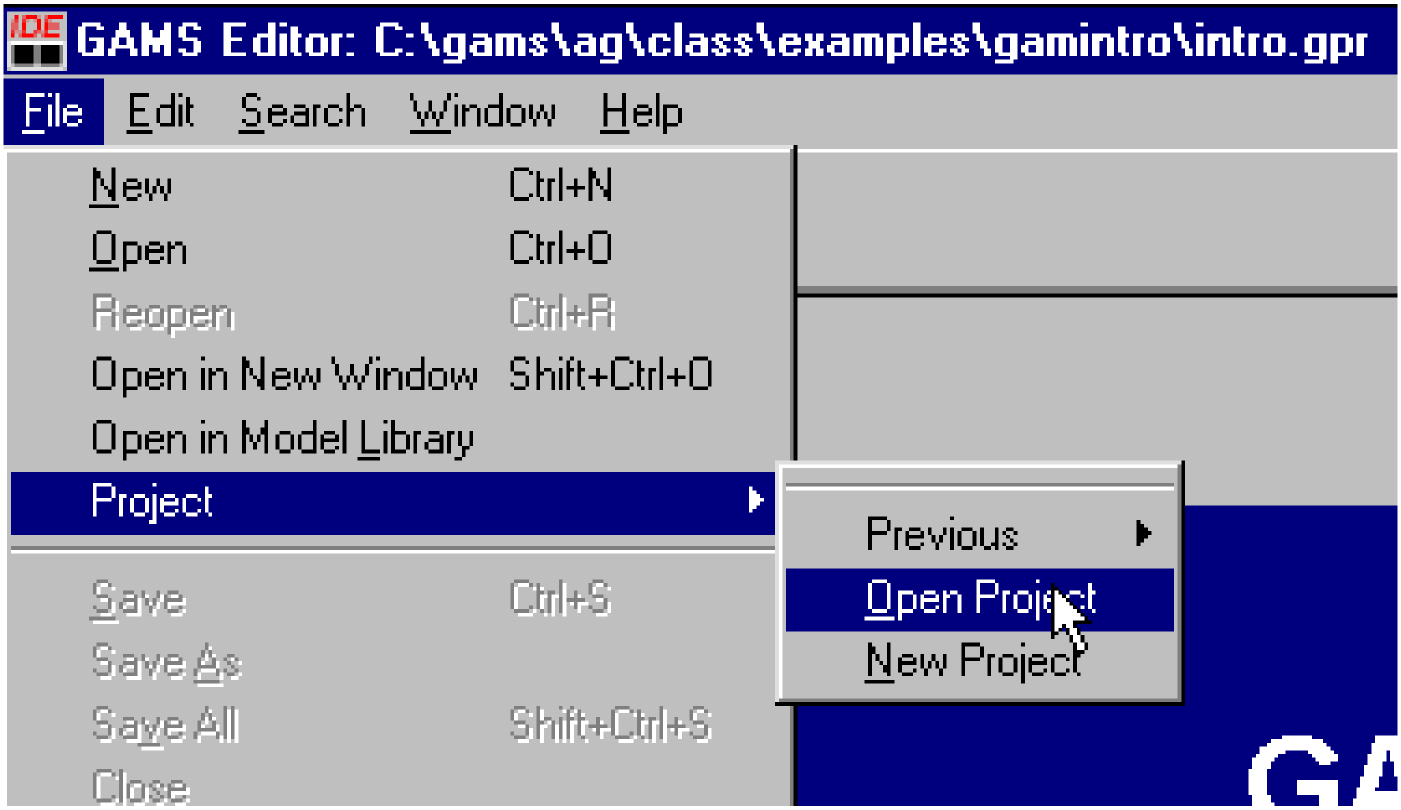
5. Run it by punching run button

6. Access LST file through process window

Introduction to GAMS
Using GAMS through the IDE.
Steps to using

Steps to using assuming install done and IDE open

3. Open project called INTRO on /example/class

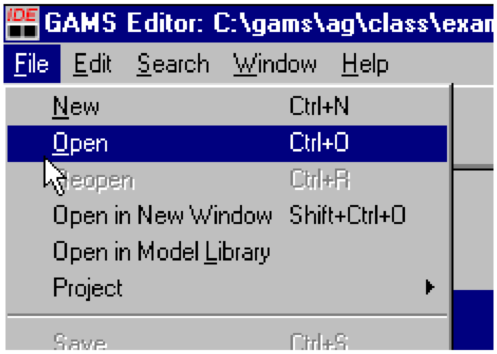


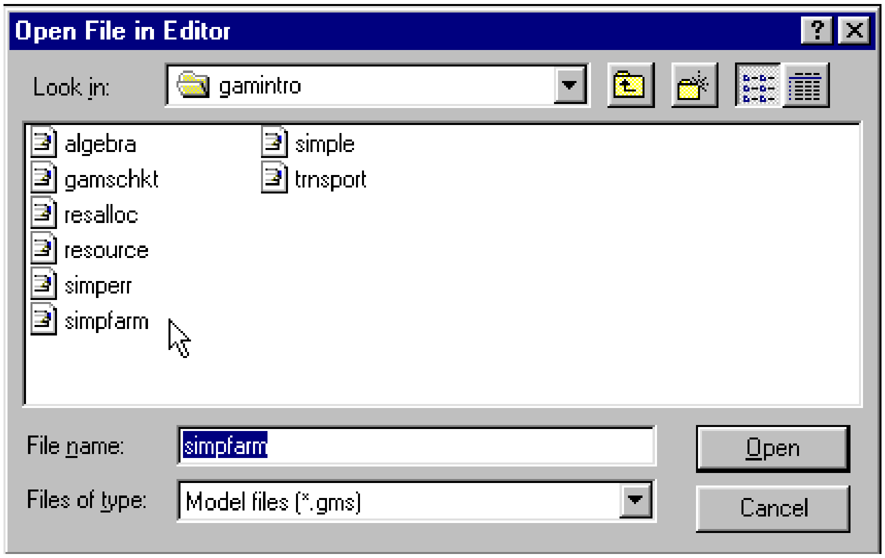
Introduction to GAMS

Using GAMS through the IDE.

Steps to using

4. Open file called simpfarm.gms





Look in: gamintro

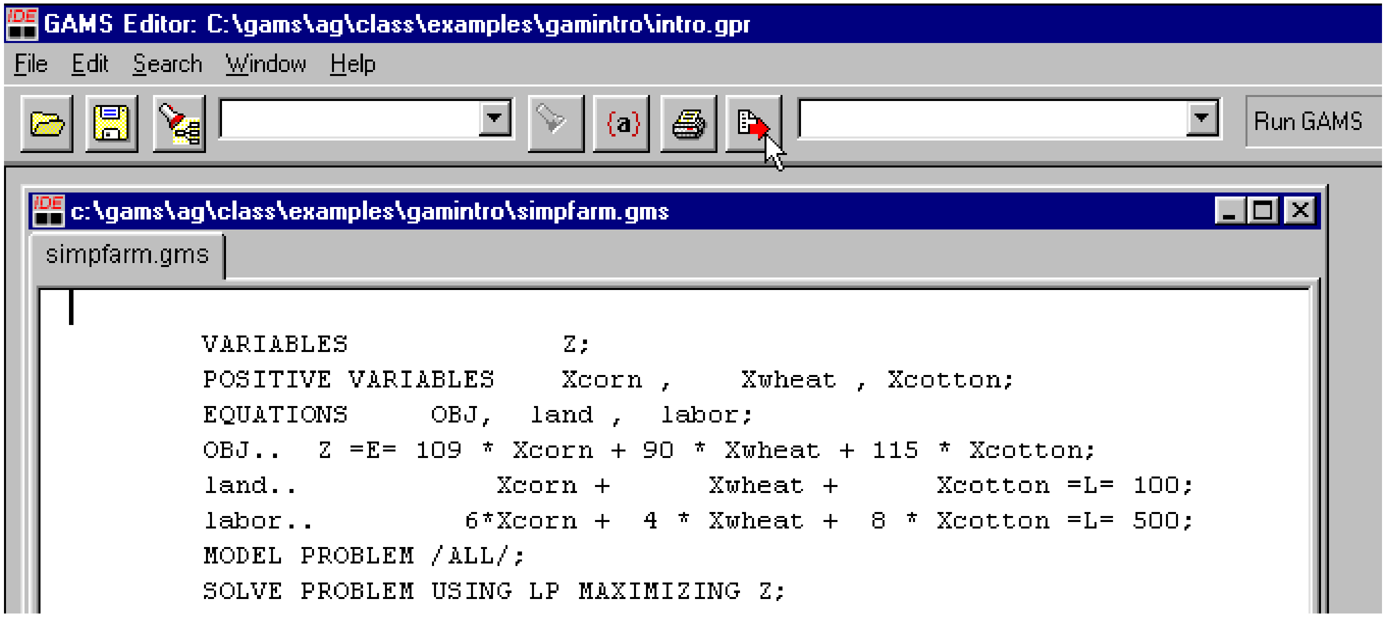
File name: simpfarm

Files of type: Model files (*.gms)

Introduction to GAMS
Using GAMS through the IDE.
Steps to using

Do a Little Housekeeping

Drag the window that shows the file into the upper left hand corner


GAMS Editor: C:\gams\ag\class\examples\gamintro\intro.gpr
File Edit Search Window Help
Run GAMS
c:\gams\ag\class\examples\gamintro\simpfarm.gms
simpfarm.gms

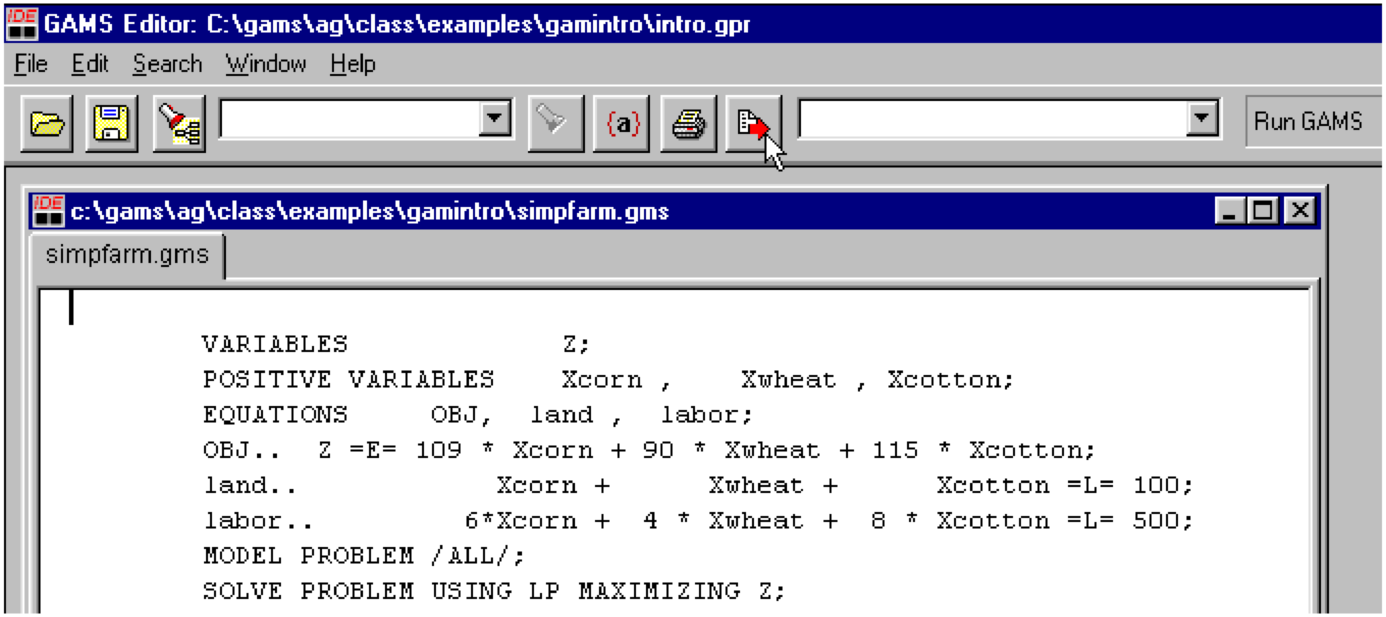
```
        VARIABLES                 Z;
        POSITIVE VARIABLES    Xcorn ,     Xwheat , Xcotton;
        EQUATIONS      OBJ,  land ,  labor;
        OBJ..  Z =E= 109 * Xcorn + 90 * Xwheat + 115 * Xcotton;
        land..              Xcorn +      Xwheat +      Xcotton =L= 100;
        labor..           6*Xcorn +  4 * Xwheat +  8 * Xcotton =L= 500;
        MODEL PROBLEM /ALL/;
        SOLVE PROBLEM USING LP MAXIMIZING Z;
```

Introduction to GAMS

Using GAMS through the IDE.

Steps to using

5. Run it by punching run button

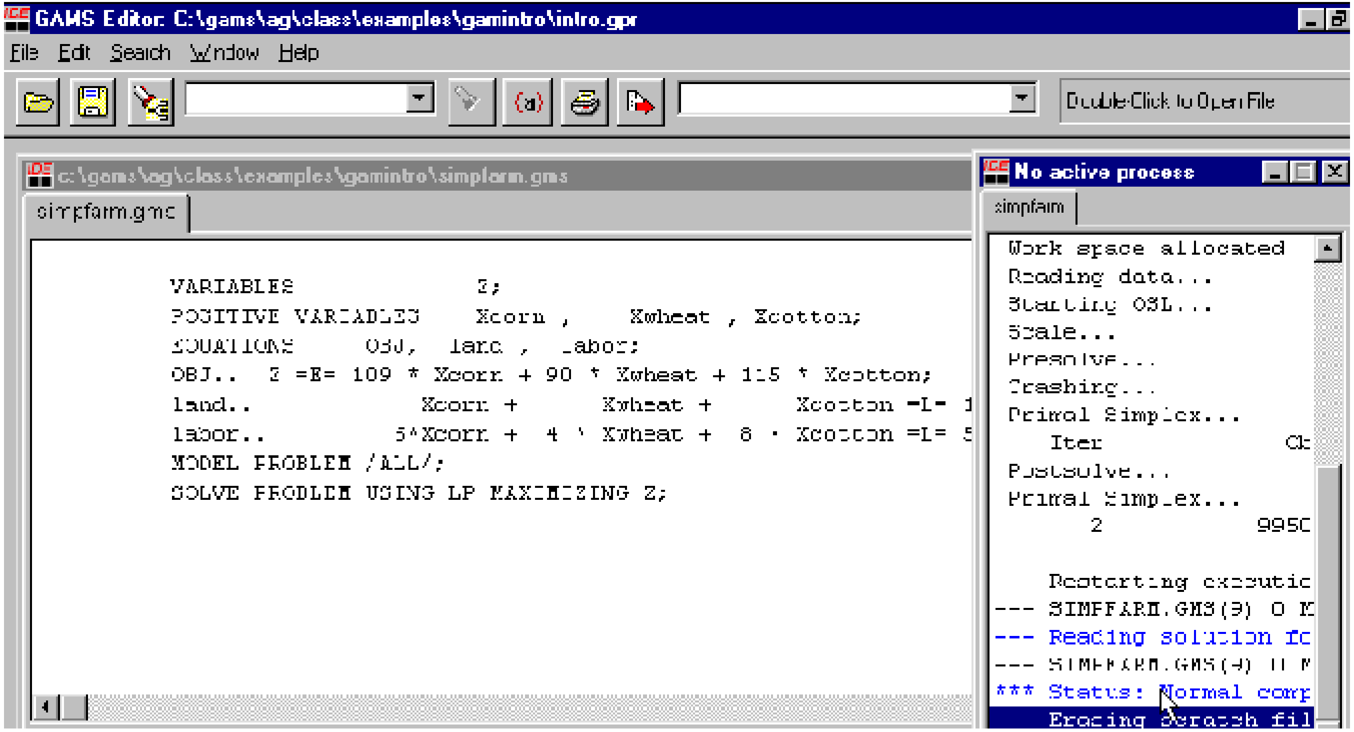

GAMS Editor: C:\gams\ag\class\examples\gamintro\intro.gpr
File Edit Search Window Help
Run GAMS
c:\gams\ag\class\examples\gamintro\simpfarm.gms
simpfarm.gms

```
VARIABLES                  Z;
POSITIVE VARIABLES    Xcorn ,     Xwheat , Xcotton;
EQUATIONS     OBJ,  land ,  labor;
OBJ..  Z =E= 109 * Xcorn + 90 * Xwheat + 115 * Xcotton;
land..             Xcorn +      Xwheat +      Xcotton =L= 100;
labor..          6*Xcorn +  4 * Xwheat +  8 * Xcotton =L= 500;
MODEL PROBLEM /ALL/;
SOLVE PROBLEM USING LP MAXIMIZING Z;
```

Introduction to GAMS
Using GAMS through the IDE.
Steps to using

Do a Little more Housekeeping.

Drag the new window that shows the process history to the far right

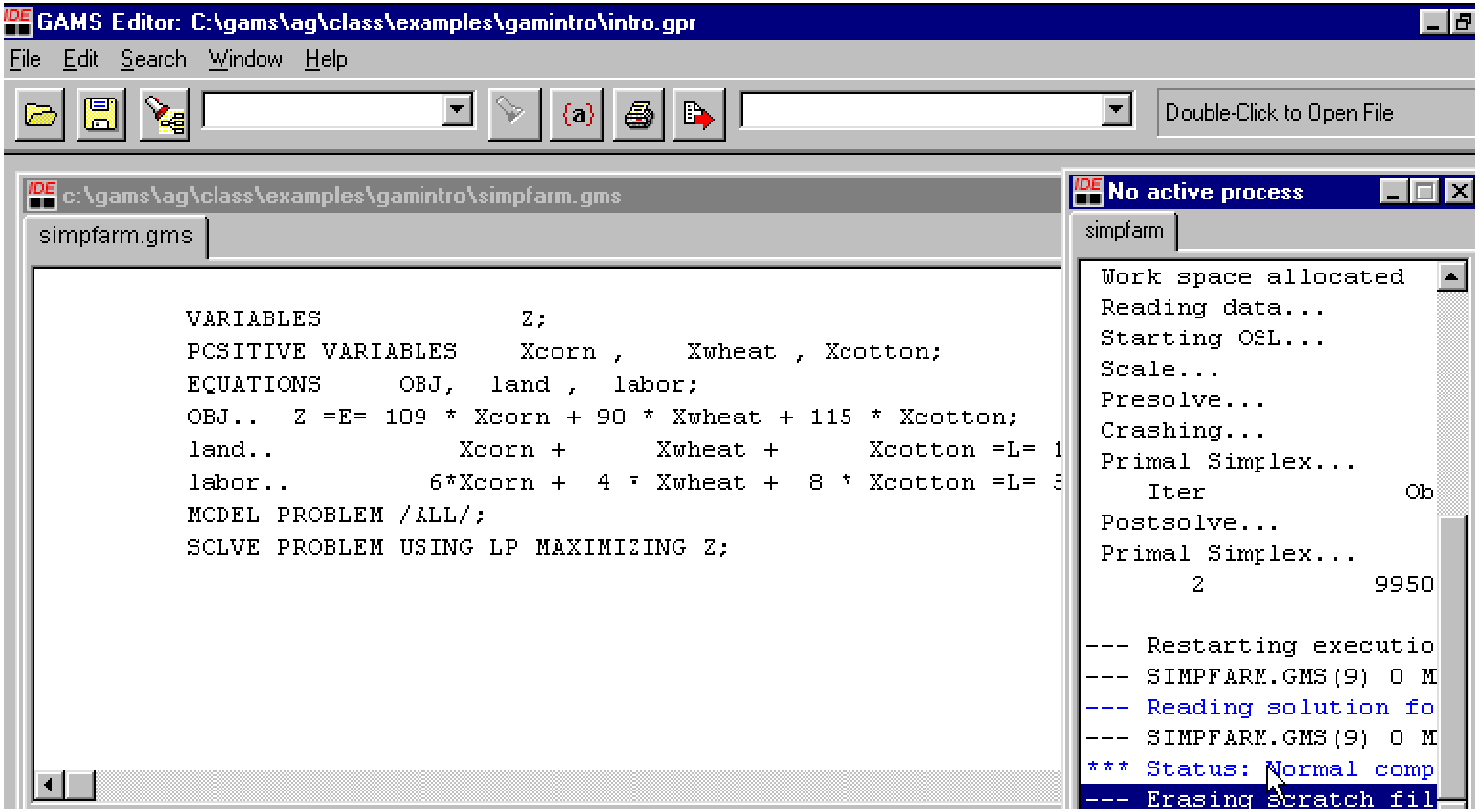

GAMS Editor: C:\gams\ag\class\examples\gamintro\intro.gpr
File
Edit
Search
Window
Help
Double-Click to Open File
c:\gams\ag\class\examples\gamintro\simpfarm.gms
simpfarm.gms
No active process
simpfarm

```
VARIABLES                  Z;
POSITIVE VARIABLES    Xcorn ,     Xwheat , Xcotton;
EQUATIONS      OBJ,  land ,  labor;
OBJ..  Z =E= 109 * Xcorn + 90 * Xwheat + 115 * Xcotton;
land..             Xcorn +      Xwheat +      Xcotton =L= 1
labor..          5*Xcorn +  4 * Xwheat +  8 * Xcotton =L= 5
MODEL PROBLEM /ALL/;
SOLVE PROBLEM USING LP MAXIMIZING Z;
```

```
 Work space allocated
 Reading data...
 Starting OSL...
 Scale...
 Presolve...
 Crashing...
 Primal Simplex...
    Iter                Ob
 Postsolve...
 Primal Simplex...
       2             9950


    Restarting executio
--- SIMPFARM.GMS(3) 0 M
--- Reading solution fo
--- SIMPFARM.GMS(4) 0 M
*** Status: Normal comp
    Erasing scratch fil
```

Introduction to GAMS
Using GAMS through the IDE.
Steps to using

6. Navigate using the process window. Double Click on Last blue line


GAMS Editor: C:\gams\ag\class\examples\gamintro\intro.gpr
File Edit Search Window Help
Double-Click to Open File
c:\gams\ag\class\examples\gamintro\simpfarm.gms
simpfarm.gms
No active process
simpfarm

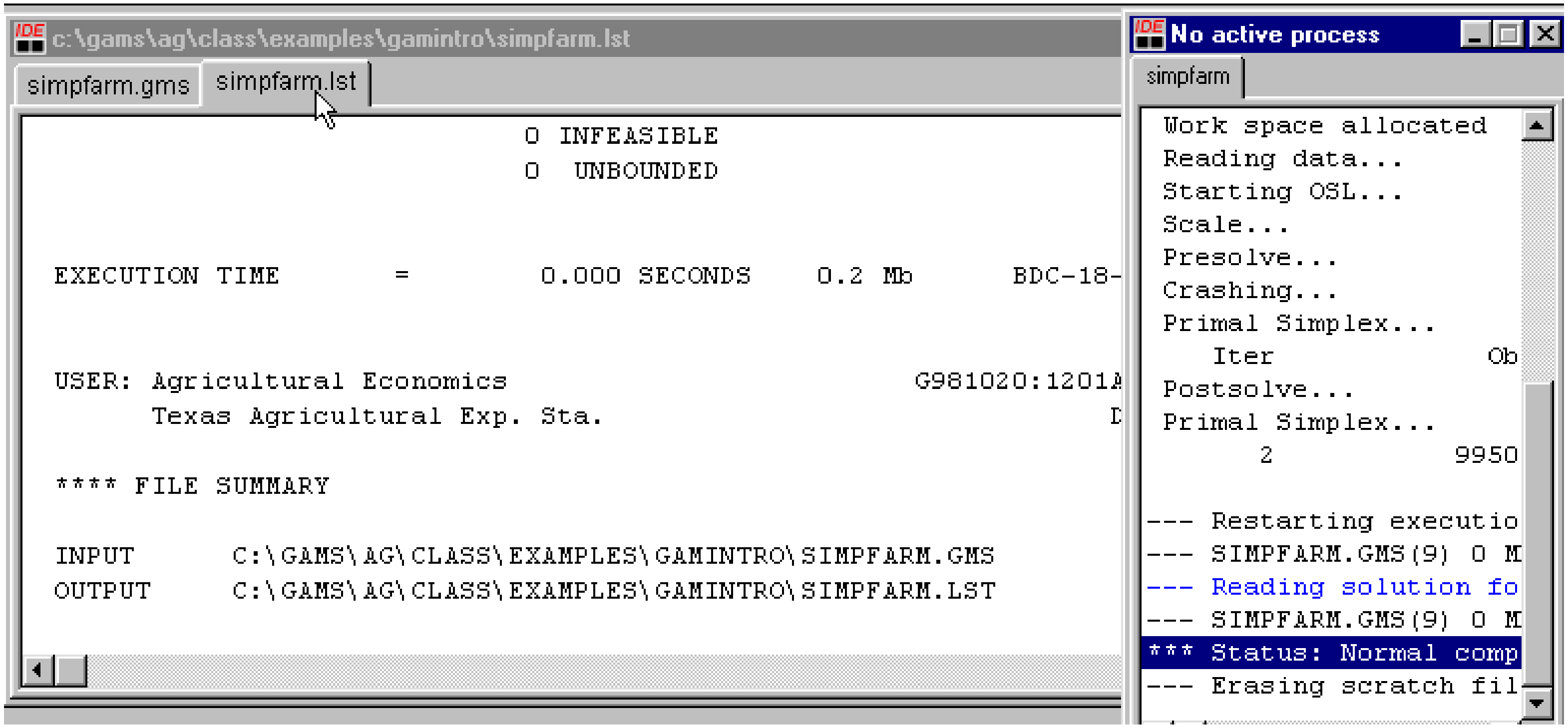
```
VARIABLES                Z;
PCSITIVE VARIABLES    Xcorn ,    Xwheat , Xcotton;
EQUATIONS      OBJ,   land ,   labor;
OBJ..  Z =E= 109 * Xcorn + 90 * Xwheat + 115 * Xcotton;
land..              Xcorn +      Xwheat +      Xcotton =L= 1
labor..           6*Xcorn +  4 * Xwheat +  8 * Xcotton =L= 3
MCDEL PROBLEM /ALL/;
SCLVE PROBLEM USING LP MAXIMIZING Z;
```

```
 Work space allocated
 Reading data...
 Starting OSL...
 Scale...
 Presolve...
 Crashing...
 Primal Simplex...
    Iter              Ob
 Postsolve...
 Primal Simplex...
       2            9950

--- Restarting executio
--- SIMPFARM.GMS(9) 0 M
--- Reading solution fo
--- SIMPFARM.GMS(9) 0 M
*** Status: Normal comp
--- Erasing scratch fil
```

©Bruce A. McCarl, Oct 1998.

Introduction to GAMS
Using GAMS through the IDE.
Steps to using

6. You have now been placed in the simpfarm..LST file



```
                                0 INFEASIBLE
                                0  UNBOUNDED


EXECUTION TIME       =        0.000 SECONDS    0.2 Mb       BDC-18-


USER: Agricultural Economics                            G981020:1201A
      Texas Agricultural Exp. Sta.                                  D

**** FILE SUMMARY

INPUT      C:\GAMS\AG\CLASS\EXAMPLES\GAMINTRO\SIMPFARM.GMS
OUTPUT     C:\GAMS\AG\CLASS\EXAMPLES\GAMINTRO\SIMPFARM.LST
```

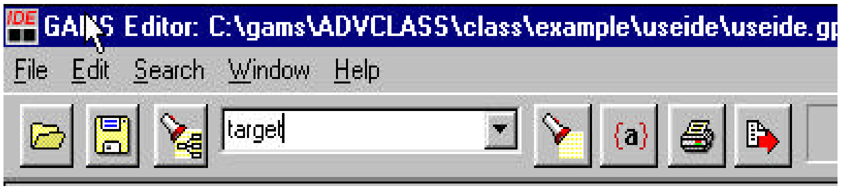
```
 Work space allocated
 Reading data...
 Starting OSL...
 Scale...
 Presolve...
 Crashing...
 Primal Simplex...
    Iter            Ob
 Postsolve...
 Primal Simplex...
       2           9950

--- Restarting executio
--- SIMPFARM.GMS(9) 0 M
--- Reading solution fo
--- SIMPFARM.GMS(9) 0 M
*** Status: Normal comp
--- Erasing scratch fil
```

Introduction to GAMS
Finding Text

The IDE provides four ways to find and/or replace text strings.

For finding strings three dialogs can be used

The fundamental ones involve use of the flashlight and search windows


GAMS Editor: C:\gams\ADVCLASS\class\example\useide\useide.gp
File
Edit
Search
Window
Help
target
(a)

Type the text string target you are after in the widow

Hitting the 🔦 finds what you want in the current file

Hitting the 🔦 finds what you want in the directory where the project is located

You can also access search and replace through the search menu. That dialogue gives more options, but only searches or replaces within the current file

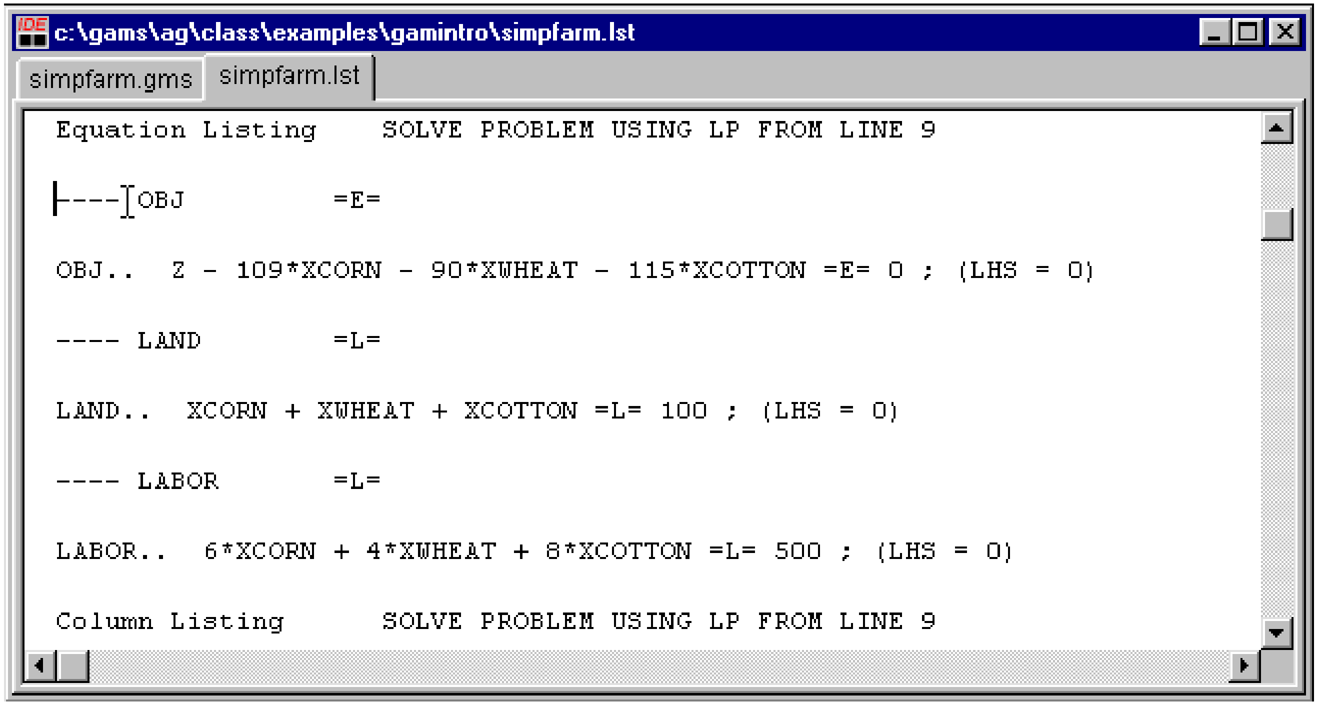
Introduction to GAMS

GAMS Usage

What Does That Model Look Like (not only IDE)

Find equation listing in lst file (look for first ----)

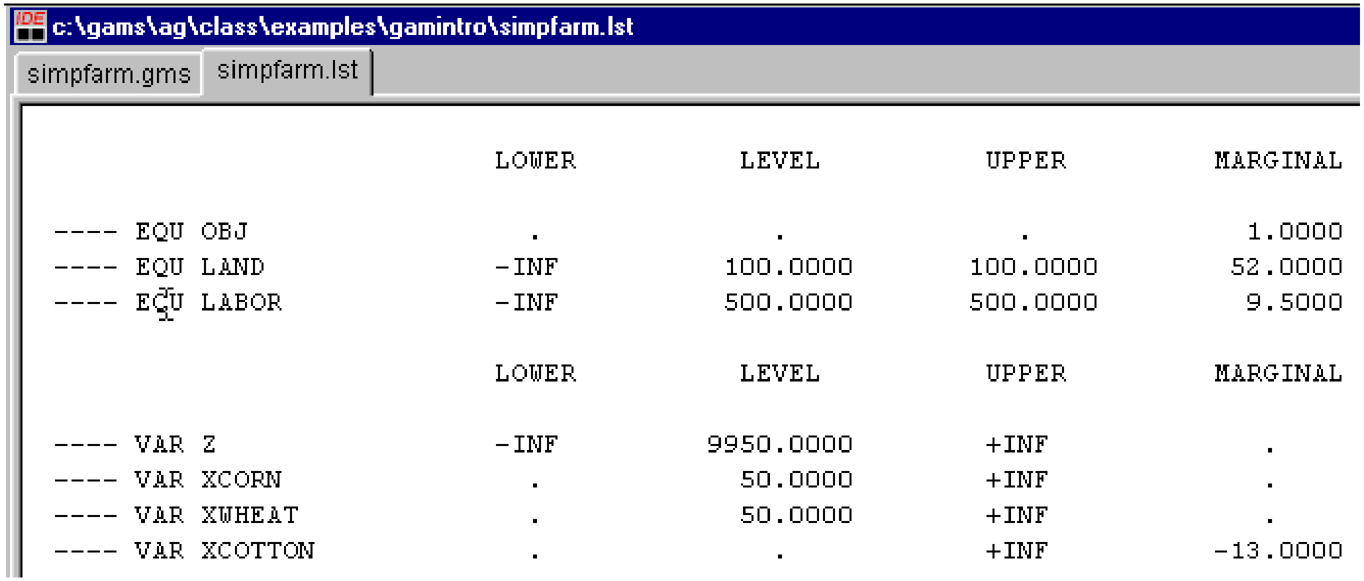
Controlled
by
LIMROW
LIMCOL
Options



```
Equation Listing    SOLVE PROBLEM USING LP FROM LINE 9

----OBJ         =E=

OBJ..  Z - 109*XCORN - 90*XWHEAT - 115*XCOTTON =E= 0 ; (LHS = 0)

---- LAND        =L=

LAND..  XCORN + XWHEAT + XCOTTON =L= 100 ; (LHS = 0)

---- LABOR       =L=

LABOR..  6*XCORN + 4*XWHEAT + 8*XCOTTON =L= 500 ; (LHS = 0)

Column Listing      SOLVE PROBLEM USING LP FROM LINE 9
```

Introduction to GAMS
Finding the Solution (not only IDE)

Find solution (look for word solution) and page down or look for ----



```
                              LOWER          LEVEL          UPPER         MARGINAL

---- EQU OBJ                    .              .              .             1.0000
---- EQU LAND                 -INF          100.0000       100.0000        52.0000
---- EQU LABOR                -INF          500.0000       500.0000         9.5000

                              LOWER          LEVEL          UPPER         MARGINAL

---- VAR Z                    -INF         9950.0000        +INF              .
---- VAR XCORN                  .            50.0000        +INF              .
---- VAR XWHEAT                 .            50.0000        +INF              .
---- VAR XCOTTON                .              .            +INF          -13.0000
```

Introduction to GAMS
Using GAMS

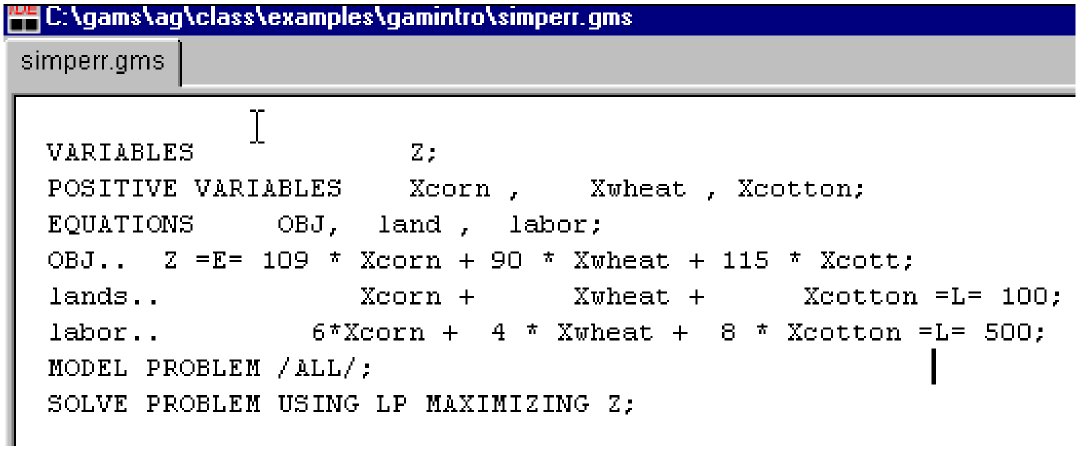
What happened during the run
-- The Process window or screen in DOS/UNIX

```
GAMS 2.50.094 Copyright (C) 1988-1998 GAMS Development. All rights reserved
--- Starting compilation                        *** Checks if your file is ok
--- SIMPFARM.GMS(9) 0 Mb                        ***(9) tells line it is on
--- Starting execution                          *** Executes your file
--- Generating model PROBLEM                    ***Sets up the LP Problem
--- SIMPFARM.GMS(9) 1 Mb                        ***(9) tells line it is on
---    3 rows, 4 columns, and 10 non-zeroes.    ***Size of LP
--- Executing OSL                               ***GAMS ceases automatically
                                                   Starting solver and gives name
                                                   of Solver used
 OSL Release 2,  GAMS Link level 3  ---  386/486 DOS  1.3.055-033
 Work space allocated          --    0.09 Mb    ***Output from Solver
 Reading data...
 Starting OSL...
 Scale...
 Presolve...
 Crashing...
 Primal Simplex...
    Iter          Objective   Sum Infeasibilities
 Postsolve...
 Primal Simplex...
       2        9950.000000     Normal Completion
                           Optimal
--- Restarting execution                        ***GAMS restarts
--- SIMPFARM.GMS(9) 0 Mb
--- Reading solution for model PROBLEM
--- SIMPFARM.GMS(9) 0 Mb
*** Status: Normal completion                      ***GAMS stops
```

Introduction to GAMS

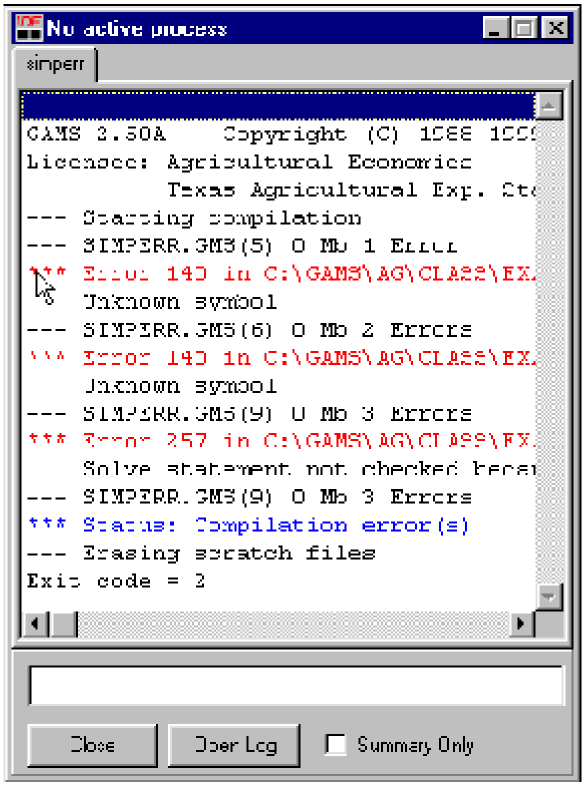
Using GAMS – Finding Errors

Spelling Mistakes (simperr.gms)



```
VARIABLES                 Z;
POSITIVE VARIABLES     Xcorn ,     Xwheat , Xcotton;
EQUATIONS      OBJ,   land ,   labor;
OBJ..  Z =E= 109 * Xcorn + 90 * Xwheat + 115 * Xcott;
lands..             Xcorn +      Xwheat +      Xcotton =L= 100;
labor..          6*Xcorn +  4 * Xwheat +  8 * Xcotton =L= 500;
MODEL PROBLEM /ALL/;
SOLVE PROBLEM USING LP MAXIMIZING Z;
```

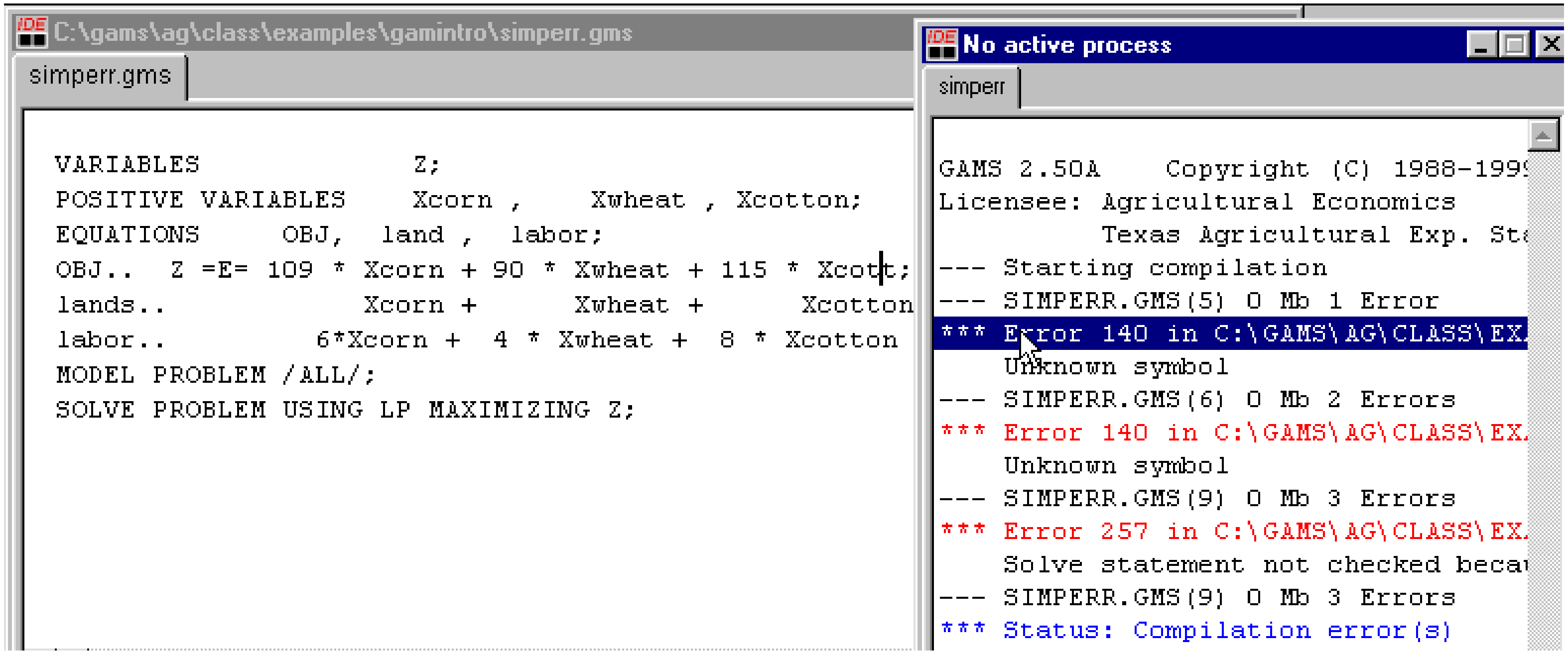
Punch the run button



```
GAMS 2.50A    Copyright (C) 1988 199
Licensee: Agricultural Economics
          Texas Agricultural Exp. Sta
--- Starting compilation
--- SIMPERR.GMS(5) 0 Mb 1 Error
*** Error 140 in C:\GAMS\AG\CLASS\EX
    Unknown symbol
--- SIMPERR.GMS(6) 0 Mb 2 Errors
*** Error 140 in C:\GAMS\AG\CLASS\EX
    Unknown symbol
--- SIMPERR.GMS(9) 0 Mb 3 Errors
*** Error 257 in C:\GAMS\AG\CLASS\EX
    Solve statement not checked becau
--- SIMPERR.GMS(9) 0 Mb 3 Errors
*** Status: Compilation error(s)
--- Erasing scratch files
Exit code = 2
```

Introduction to GAMS
Using GAMS Finding Errors

Double Click on the first Red Line



```
VARIABLES                  Z;
POSITIVE VARIABLES    Xcorn ,     Xwheat , Xcotton;
EQUATIONS      OBJ,  land ,  labor;
OBJ..  Z =E= 109 * Xcorn + 90 * Xwheat + 115 * Xcott;
lands..             Xcorn +       Xwheat +       Xcotton
labor..          6*Xcorn +  4 * Xwheat +  8 * Xcotton
MODEL PROBLEM /ALL/;
SOLVE PROBLEM USING LP MAXIMIZING Z;
```

```
GAMS 2.50A    Copyright (C) 1988-199
Licensee: Agricultural Economics
          Texas Agricultural Exp. St
--- Starting compilation
--- SIMPERR.GMS(5) 0 Mb 1 Error
*** Error 140 in C:\GAMS\AG\CLASS\EX
    Unknown symbol
--- SIMPERR.GMS(6) 0 Mb 2 Errors
*** Error 140 in C:\GAMS\AG\CLASS\EX
    Unknown symbol
--- SIMPERR.GMS(9) 0 Mb 3 Errors
*** Error 257 in C:\GAMS\AG\CLASS\EX
    Solve statement not checked becau
--- SIMPERR.GMS(9) 0 Mb 3 Errors
*** Status: Compilation error(s)
```

Why this error, cotton is misspelled in previous line (for a discussion of error repair see fixmodel.pdf)

Introduction to GAMS
GAMS and Algebra

The problem above is a special case of the general resource allocation problem

$$\begin{array}{llll} \text{Max} & \underset{j}{'} \; c_j X_j & & \\ \text{s.t.} & \underset{j}{'} \; a_{ij} X_j & \# \;\; b_i & \text{for all } i \\ & \quad X_j & \$ \;\; 0 & \text{for all } j \end{array}$$

$$\begin{array}{llllllll} \text{Maximize} & 109(X_{corn} & \% & 90(X_{wheat} & \% & 115(X_{cotton} & & \\ \text{subject to} & X_{corn} & \% & X_{wheat} & \% & X_{cotton} & \# \; 100 & \text{(land)} \\ & 6(X_{corn} & \% & 4(X_{wheat} & \% & 8(X_{cotton} & \# \; 500 & \text{(labor)} \\ & X_{corn} & , & X_{wheat} & , & X_{cotton} & \$ \quad 0 & \text{(nonnegativity)} \end{array}$$

$$\begin{array}{llllll} j = & \{ & \text{corn} & \text{wheat} & \text{cotton} & \} \\ i = & \{ & \text{land} & \text{labor} & & \} \\ x_j = & \{ & X_{corn} & X_{wheat} & X_{cotton} & \} \\ c_j = & \{ & 109 & 90 & 115 & \} \\ a_{ij} = & & 1 & 1 & 1 & \\ & & 6 & 4 & 8 & \\ b_i = & \{ & 100 & 500 & & \}' \end{array}$$

Introduction to GAMS

GAMS and Algebra

GAMS is built around summation notation

You have to be comfortable using summation notation to use GAMS *(reference for those wishing to review summation notation and see its inner links to GAMS -- see appendix one of newbook.pdf)*

Suppose x_i is defined with three elements

Algebra

$$\underset{i}{'}\ x_i\ '\ x_1\ \%\ x_2\ \%x_3$$

GAMS

```
z = SUM(I, X(I));
```

i   is a set in GAMS

z   is a scalar or variable

x(i) is a parameter or variable defined over set i

the sum automatically treats all cases of i

This equation can be either a model equation in an LP or an item to be calculated in the code

Introduction to GAMS
A Better algebraic model

```
VARIABLES              Z;
POSITIVE VARIABLES    Xcorn ,    Xwheat , Xcotton;
EQUATIONS     OBJ,  land ,  labor;
OBJ..  Z =E= 109 * Xcorn + 90 * Xwheat + 115 * Xcotton;
land..            Xcorn +      Xwheat +      Xcotton =L= 100;
labor..         6*Xcorn +  4 * Xwheat +  8 * Xcotton =L= 500;
MODEL PROBLEM /ALL/;
SOLVE PROBLEM USING LP MAXIMIZING Z;
```

New (algebra.gms)

```
SET       j            /Corn,Wheat,Cotton/
          i            /Land ,Labor/;
PARAMETER
          c(j)         / corn     109    ,wheat   90 ,cotton    115/
          b(i)         /land 100 ,labor 500/;
TABLE a(i,j)
                          corn      wheat      cotton
          land              1          1           1
          labor             6          4           8      ;
POSITIVE VARIABLES      x(j);
VARIABLES                     PROFIT            ;
EQUATIONS                    OBJective         ,  constraint(i) ;
 OBJective..     PROFIT=E=   SUM(J,(c(J))*x(J)) ;
 constraint(i)..   SUM(J,a(i,J) *x(J))  =L= b(i);
MODEL   RESALLOC /ALL/;
SOLVE RESALLOC USING LP MAXIMIZING PROFIT;
```

Note GAMS is not terribly sensitive about capitalization as the example illustrates. Any alternative capitalization sequence can be used. However, GAMS uses the first found capitalization sequence in all displays.

Introduction to GAMS
Dissecting the GAMS formulation -- The set

Above we needed to use the subscripts i and j for addressing the variables equations and data items.

In GAMS subscripts are SETs. In order to use any subscript one must declare an equivalent set.

The set declaration contains
- the set name
- a list of elements contained in the set (up to 31 characters long spaces etc allowed in quotes)
- optional labels describing the whole set
- optional labels defining individual set elements

General format for a set statement is:

```
SET setname   optional defining text
    /  first set element name       defining text
         second set element name  defining text
         ...  /;
```

Examples

```
SETs     j        /x1,x2,x3/
         i        /r1 ,r2/;
SET   PROCESS   PRODUCTION PROCESSES   /X1,X2,X3/;
SET   commodities Crop commodities   /
                      corn       in bushels,
                      wheat      in metric tons,
                      milk       in hundred pounds/    ;
```

Introduction to GAMS

Dissecting the GAMS formulation -- Data entry

Above we needed data for c(j), a(i,j), b(i)

How were they entered

GAMS provides for three forms of data entry. These involve PARAMETER, SCALAR and TABLE formats

SCALAR format is used to enter items which are not defined with respect to sets.

```
scalar
     item1name    optional label text   /numerical value/
     item2name    optional label text   /numerical value/
     ...                                                 ;
```

Examples include

```
scalar dataitem    /100/;
scalar landonfarm   total arable acres /100/;
scalars
   landonfarm /100/
   pricecorn   1992 corn price per bushel   /2.20/;
```

Introduction to GAMS

Dissecting the GAMS formulation -- Data entry PARAMETERs

Parameter format is used to enter items defined with respect to sets. Generally parameter format is used when data items which are one-dimensional (vectors) although multidimensional cases can be entered.

The general format for parameter entry is:

```
Parameter
   itemname(setdependency)  optional text
            /  first set name        appropriate value,
               second set name       appropriate value,
               ...   /;
```

Examples

```
   PARAMETER        c(j)     / x1     3    ,x2   2 ,x3    0.5/
                       b(i)     /r1 10 ,r2 3/;
   PARAMETER
      PRICE(PROCESS)     PRODUCT PRICES BY PROCESS
                                 /X1 3,X2 2,X3 0.5/;
      RESORAVAIL(RESOURCE)  RESOURCE AVAILABLITY
                              /CONSTRAIN1 10 ,CONSTRAIN2 3/;
   Parameter   multidim(i,j,k) three dimensional /i1.j1.k1 100 ,i2.j1.k20 /;
```

Multidimensional parameters particular the useful when bringing data in other programs.

Introduction to GAMS

Dissecting the GAMS formulation -- Data entry

TABLE

TABLE format is used to enter items which are dependent on two more sets.

The general format is

```
Table itemname(setone, settwo ... ) descriptive text
                        set_2_element_1   set_2_element_2
set_1_element_1             value_11          value_12
set_1_element_2             value_21          value_22;
```

Examples include

```
TABLE a(i,j)
                 corn      wheat      cotton
      land         1          1          1
      labor        6          4           8      ;
TABLE RESOURUSE(RESOURCE,PROCESS) RESOURCE USAGE
                 Makechair    Maketable    Makelamp
plantcap               3                2               1.1
salecontrct            1               -1;
Table fivedim(i,j,k,l,m) fivedimensional
                 l1.m1 l2.m2
      i1.j1.k2      11    13
      i2.j1.k11     6     -3
+                i3.m1 i2.m7
      i1.j1.k2      1        3
      i10.j1.k4    7        9;
```

Alignment is important

Introduction to GAMS

Dissecting the GAMS formulation

Bad modeling (simple.gms)

```
VARIABLES                    Z;
POSITIVE VARIABLES     X1 ,    X2,   X3;
EQUATIONS              OBJ,  CONSTRAIN1 ,  CONSTRAIN2;
OBJ..                  Z =E= 3 * X1 + 2 * X2 + 0.5* X3;
CONSTRAIN1..           3*X1 + 2*X2 +1.1*X3=L= 10;
CONSTRAIN2..           X1 - X2 =L= 3;
MODEL     PROBLEM      /ALL/;
SOLVE PROBLEM USING LP MAXIMIZING Z;
```

GAMS permits up to ten character names for variables , the equations and other structural elements. Structural elements can also have explanatory comments attached to them.

GAMS also allows us to treat common structures algebraically. The above example does none of this and is really an example of bad GAMS coding so let us illustrate with a better example (also see appendix I in fixmodel.pdf for discussion)

Introduction to GAMS

A Better Model – better names (resalloc.gms)

```
SET   PROCESS   PRODUCTION PROCESSES
             / makechair  chair manufaxture(X1) ,
               maketable  table maufacture(X2),
               makelamp      Lamp Manufacture(X3)/
      RESOURCE  TYPES OF RESOURCES
                / plantcap    Plant Capacity (CONSTRAIN1)
               salecontrct    Sales agreement limiting production (CONSTRAIN2)/;

PARAMETER
   PRICE(PROCESS)      PRODUCT PRICES BY PROCESS
             /makechair     6.5,  maketable      3,  makelamp       0.5/
   Yield(process)  yields per unit of the process
             /Makechair      2,      maketable      6,   makelamp       3/
   PRODCOST(PROCESS)      COST BY PROCESS
              /Makechair 10 ,Maketable 6, Makelamp 1/
TABLE RESOURUSE(RESOURCE,PROCESS) RESOURCE USAGE
                         Makechair    Maketable    Makelamp
   plantcap                 3              2              1.1
   salecontrct              1             -1;
POSITIVE VARIABLES
   PRODUCTION(PROCESS) ITEMS PRODUCED BY PROCESS;
VARIABLES         PROFIT          TOTALPROFIT;
EQUATIONS
    OBJT                              OBJECTIVE FUNCTION ( PROFIT )
   AVAILABLE(RESOURCE)     RESOURCES AVAILABLE ;
   OBJT.. PROFIT=E=   SUM(PROCESS,(PRICE(PROCESS)*yield(process)
                        -PRODCOST(PROCESS))*PRODUCTION(PROCESS)) ;
AVAILABLE(RESOURCE)..
   SUM(PROCESS,RESOURUSE(RESOURCE,PROCESS)
             *PRODUCTION(PROCESS))  =L= RESORAVAIL(RESOURCE);
MODEL RESALLOC /ALL/;
SOLVE RESALLOC USING LP MAXIMIZING PROFIT;
```

Introduction to GAMS

Dissecting the GAMS formulation
Calculated Data – In the Model

```
PARAMETER
   PRICE(PROCESS)      PRODUCT PRICES BY PROCESS
            /makechair      6.5,  maketable      3,  makelamp       0.5/
   Yield(process)  yields per unit of the process
            /Makechair       2,      maketable      6,   makelamp       3/
   PRODCOST(PROCESS)     COST BY PROCESS
             /Makechair 10 ,Maketable 6, Makelamp 1/
   OBJT.. PROFIT=E=   SUM(PROCESS,(PRICE(PROCESS)*yield(process)
                            -PRODCOST(PROCESS))*PRODUCTION(PROCESS)) ;
```

Terms can be included in a GAMS model which involve calculations. When calculations are included in model specification equations (those identified with the ..), then the calculations are automatically executed every time the model is set up.

Example

```
SOLVE RESALLOC USING LP MAXIMIZING PROFIT;
price(“makechair”)=8;
SOLVE RESALLOC USING LP MAXIMIZING PROFIT;
```

Model is first solved at the original price of 6.5, then the price is changed to equal 8 and model is solved again with the altered price in effect doing a comparative statics analysis of solution sensitivity to price.

Introduction to GAMS
Dissecting the GAMS formulation
Calculated Data – In the Code

Data do not only have to be directly entered in the code as constants. Rather data can be calculated or altered in many different ways.

Example (trnsport.gms)

```
Table distance(Source,Destinaton)  distance in thousands of miles
                      new-york     chicago        topeka
     seattle             2.5             1.7             1.8
     san-diego         2.5             1.8             1.4  ;
Scalar    prmilecst   freight cost in $ per case per 1000 miles /90/
              loadcost    freight loading cost in $ per case /25/        ;
Parameter trancost(Source,Destinaton)  transport cost in dollars per case ;
   trancost(Source,Destinaton) =
        loadcost + prmilecst * distance(Source,Destinaton) ;
Costsum ..  totalcost  =e=  sum((Source,Destinaton),
        trancost(Source,Destinaton)*transport(Source,Destinaton));
```

Thus GAMS allows potentially rather extensive data manipulation.

Watch out for the dynamic vs. the non-dynamic calculation. The calculations such as the one above are only once. Calculations in the model equations are done every time the model is set up.

Introduction to GAMS
Dissecting the GAMS formulation
Displaying Calculated Data

GAMS allows one to display an array of data

General format

display itemname;

Example (trnsport.gms)

Entering

```
display trancost;
```

Results in the following in the trnsport.LST file

```
   ---- 24 PARAMETER TRANCOST transport cost in dollars
                                  per case
             New York     Chicago      Topeka
Seattle       250.000     178.000     187.000
San Diego     250.000     187.000     151.000
```

You can also control precision in displays

```
option decimals=0;
display trancost;
```

Yields

```
---- 26 PARAMETER TRANCOST transport cost in dollars
             New York     Chicago      Topeka
Seattle           250         178         187
San Diego         250         187         151
```

Introduction to GAMS

Looking at your model with LIMROW LIMCOL

GAMS can cause one to lose touch with the exact optimization model being solved. The algebraic model gives a general feel for model structure but because

a) data can be calculated

b) models can be big and

c) unanticipated cases can be covered

modelers often don’t know exactly what is in the model.

Thus, when setting up a model one may need to look at individual equations and variables. GAMS permits this through the use of model element displays stimulated by the LIMROW and LIMCOL options.

When GAMS runs a display of the first three variables and equations in each block is included in the LST file

For equations we get something like (trnsport.gms)

```
 ---- COSTSUM     =E=  total transport cost -- objective function
COSTSUM..- 250*TRANSPORT(Seattle,New York) -178*TRANSPORT(Seattle,Chicago)
             - 187*TRANSPORT(Seattle,Topeka) - 250*TRANSPORT(San Diego,New York)
             - 187*TRANSPORT(San Diego,Chicago)-151*TRANSPORT(San Diego,Topeka)
             + TOTALCOST =E= 0 ; (LHS = 0)

---- SUPPLYBAL   =L=  supply limit at source plants

SUPPLYBAL(Seattle)..  TRANSPORT(Seattle,New York) + TRANSPORT(Seattle,Chicago)
    + TRANSPORT(Seattle,Topeka) =L= 350 ; (LHS = 0)
```

Looking at your model with LIMROW LIMCOL

For variables we get something like (trnsport.gms)

```
---- TRANSPORT  shipment quantities in cases
TRANSPORT(Seattle,New York)
            (.LO, .L, .UP = 0, 0, +INF)
    -250      COSTSUM
      1      SUPPLYBAL(Seattle)
      1      DEMANDBAL(New York)

TRANSPORT(Seattle,Chicago)
            (.LO, .L, .UP = 0, 0, +INF)
    -178      COSTSUM
      1      SUPPLYBAL(Seattle)
      1      DEMANDBAL(Chicago)
REMAINING 3 ENTRIES SKIPPED
---- TOTALCOST  total transportation costs in dollars
TOTALCOST
            (.LO, .L, .UP = -INF, 0, +INF)
      1      COSTSUM
```

If we want more or less variables or equations we can do this by altering the LIMROW and LIMCOL options.

To eliminate place the following in the code

```
Option limrow=0;
option limcol=0;
```

To expand place the following in the code

```
Option limrow=100; (or any other number)
option limcol=100;
```

Unfortunately for large models using this model examination procedure can generate very substantial output files

Introduction to GAMS

Looking at your model with GAMSCHK

I have developed an alternative way of displaying models called GAMSCHK. To run GAMSCHK we insert the following line in the model right before the solve

```
     option lp=gamschk;
```

We also create an another file- the GCK file - which tells GAMSCHK what to do

Example

Given the file gamschkt.gms create the file gamschkt.gck which contains (see gamschk.pdf for GCK file contents description)

```
displaycr
    variables
         transport(se*)
    equations
        dem*(n*)
picture
blockpic
postopt
```

The GAMSCHKT.LST file then contains the output on the following pages

Looking at your model with GAMSCHK

Then we get

```
----#### Executing DISPLAYCR

 ----###  DISPLAYING VARIABLES


 ----## VAR TRANSPORT


 ##  TRANSPORT(Seattle,"New York")
 COSTSUM                                     -250.00
 SUPPLYBAL(Seattle)                           1.0000
 DEMANDBAL("New York")                        1.0000

 ##  TRANSPORT(Seattle,Chicago)
 COSTSUM                                     -178.00
 SUPPLYBAL(Seattle)                           1.0000
 DEMANDBAL(Chicago)                           1.0000

 ##  TRANSPORT(Seattle,Topeka)
 COSTSUM                                     -187.00
 SUPPLYBAL(Seattle)                           1.0000
 DEMANDBAL(Topeka)                            1.0000

 ----###    DISPLAYING EQUATIONS

 ----## EQU DEMANDBAL

 ##  DEMANDBAL("New York")
 TRANSPORT(Seattle,"New York")               1.0000
 TRANSPORT("San Diego","New York")           1.0000
    =G=                      325.00
```

Looking at model with GAMSCHK- PICTURE

```
 ### PICTURE -  COEFFICIENT CODES
   LOWER BOUND    CODE    UPPER BOUND
    (INCLUSIVE)           (LESS THAN)
      100.00000     F      1000.00000
       10.00000     E       100.00000
        1.00000     D        10.00000
        1.00000     C         1.00000
        0.00000     0         0.00000
       -0.50000     1         0.00000
       -1.00000     2        -0.50000
       -1.00000     3        -1.00000
      -10.00000     4        -1.00000
     -100.00000     5       -10.00000
    -1000.00000     6      -100.00000
            | T T T T T T T     R
            | R R R R R R O     H     P     N
            | A A A A A A T     S     O     E     R
            | N N N N N N A     C     S     G     O
            | S S S S S S L     O     I A   A A   W
            | P P P P P P C     E     T I   T I   C
            | O O O O O O O     F     I J     J   N
            | R R R R R R S     F     V ,   V ,   T
            | T T T T T T T     S     E S   E S   S
            | 1 2 3 4 5 6 1
             --------------
   COSTSUM 1| 6 6 6 6 6 6 C   = 0       1     6     7
 SUPPLYBAL 1| C C C           < F       3     0     3
 SUPPLYBAL 2|       C C C     < F       3     0     3
 DEMANDBAL 1| C     C         > F       2     0     2
 DEMANDBAL 2|   C     C       > F       2     0     2
 DEMANDBAL 3|     C     C     > F       2     0     2
             --------------
 POSITIVE   | 2   2   2   1
 COLUMN CTS |   2   2   2
 ----### Dictionary of Variables
TRANSPORT      1: TRANSPORT(Seattle,"New York")
TRANSPORT      2: TRANSPORT(Seattle,Chicago)
TRANSPORT      3: TRANSPORT(Seattle,Topeka)
TRANSPORT      4: TRANSPORT("San Diego","New York")
TOTALCOST      1: TOTALCOST
 ----### Dictionary of Equations
COSTSUM        1: COSTSUM
SUPPLYBAL      1: SUPPLYBAL(Seattle)
SUPPLYBAL      2: SUPPLYBAL("San Diego")
DEMANDBAL      1: DEMANDBAL("New York")
```

Looking at your model with GAMSCHK- Blockpic

```
### A. Aggregate Block Picture -- Strip   1
              |  T  T  |
              |  R  O  |
              |  A  T  |
              |  N  A  |
              |  S  L  |
              |  P  C  |
              |  O  O  |     R
              |  R  S  |     H
              |  T  T  |     S
 ----------------------------
 COSTSUM      |  -  +  |  E  0
 SUPPLYBAL    |  +     |  L  +
 DEMANDBAL    |  +     |  G  +
 ----------------------------
 Variable Typ |  +  u

### B. Picture Giving Number of Coefficients by Block
      |    T     T  |            o
              |    R     O  |            e     o
              |    A     T  |            f     f
              |    N     A  |            f
              |    S     L  |
              |    P     C  |            C     E
              |    O     O  |      R     n     q
              |    R     S  |      H     t     n
              |    T     T  |      S     s     s
 -------------------------------------------------
 COSTSUM      |          1+ | E          1+    1
              |    6-       |            6-
 SUPPLYBAL    |    6+       | L    2+    6+    2
              |             |
 DEMANDBAL    |    6+       | G    3+    6+    3
 -------------------------------------------------
 Coeff Cnts   |   12+    1+ |      5+   13+
              |    6-       |            6-
 # of  Vars   |    6     1  |
 Variable Typ |   >=0   <0>
```

Looking at your model with GAMSCHK- Blockpic

```
### C. Picture Giving Average Number of Coefficients by Column
            |      T       T                   f
            |      R       O                   s       o
            |      A       T                           f
            |      N       A                   P
            |      S       L                   e
            |      P       C                   r       E
            |      O       O           R       E       q
            |      R       S           H       q       n
            |      T       T           S       u       s
 ---------------------------------------------------------
 COSTSUM    |              1+ | E              1+      1
            |      1-         |                6-
 SUPPLYBAL  |      1+         | L      2+      3+      2
            |                 |
 DEMANDBAL  |      1+         | G      3+      2+      3
            |                 |
 ---------------------------------------------------------
 Cfs PerVar |      2+      1+ |
            |      1-         |
 # of  Vars |      6       1
  Var Type  |    >=0     <0>
```

```
### D. Scaling Data - Maximum & Minimum Coefficients by Block
               |                            R         E
               |        T         T         H         q
               |        R         O         S         u
               |        A         T
               |        N         A         M         M
               |        S         L         a         a
               |        P         C         x         x
               |        O         O         M         M
               |        R         S         i         i
               |        T         T         n         n
 ----------------------------------------------------------
 COSTSUM    Max|       250       1                 250
            Min|       151       1                   1
 SUPPLYBAL  Max|         1                 600       1
            Min|         1                 350       1
 DEMANDBAL  Max|         1                 325       1
            Min|         1                 275       1
 ----------------------------------------------------------
  Total Var Max|       250       1       600
            Min|         1       1       275
```

Looking at your model with GAMSCHK Postopt

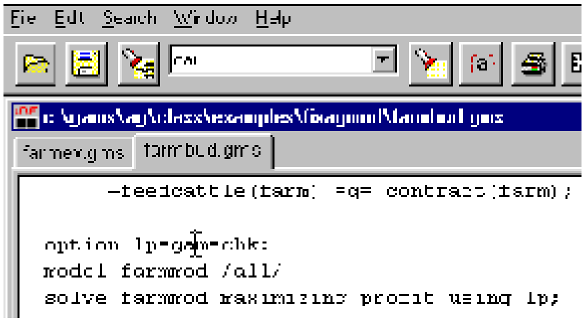
```
 ----###   ROW SUMMING EQUATIONS

----## EQU COSTSUM

##  COSTSUM

  VAR                                Aij         Xj       Aij*Xj
TRANSPORT(Seattle,"New York")    -250.00      50.000     -12500.
TRANSPORT(Seattle,Chicago)       -178.00      300.00     -53400.
TRANSPORT(Seattle,Topeka)        -187.00     0.00000     0.00000
TRANSPORT("San Diego","New York")-250.00      275.00     -68750.
TRANSPORT("San Diego",Chicago)   -187.00     0.00000E    0.00000
TRANSPORT("San Diego",Topeka)    -151.00      275.00     -41525.
TOTALCOST                         1.0000    176180       176180
   =E=                                                     =E=
RHS COEFF                                            0.00000E+00

SHADOW PRICE                                            1.0000
```

```
----###  BUDGETING VARIABLES


----## VAR TRANSPORT


##  TRANSPORT(Seattle,"New York")
  SOLUTION VALUE                                50.0000

  EQN                           Aij         Ui       Aij*Ui
COSTSUM                     -250.00      1.0000     -250.00
SUPPLYBAL(Seattle)           1.0000     0.00000E+00 0.00000E+00
DEMANDBAL("New York")        1.0000      250.00      250.00
TRUE REDUCED COST                                   0.00000E+00
```

Accessing documentation on GAMS through the IDE.

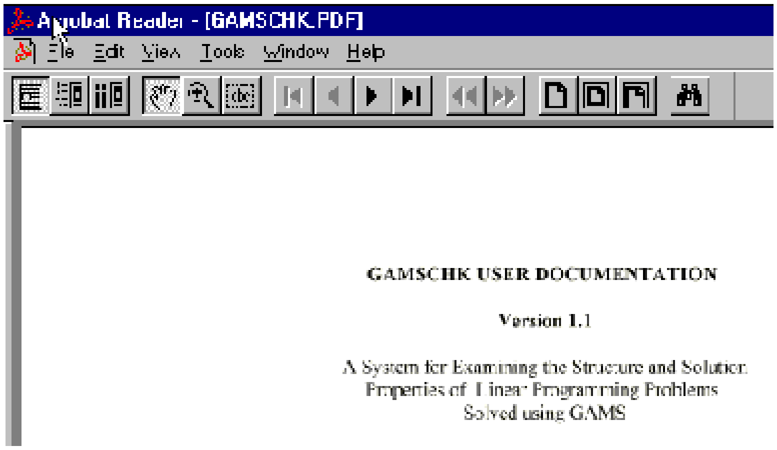
The GAMSIDE has a tie in to documentation. In particular suppose we wish to know about a particular item and there happens to be a file on that item. For example suppose we are going to use GAMSCHK and our source code contains the line
option lp=gamschk;

If we place the cursor over the word GAMSCHK and press the <F1> key as follows



```
        -feedcattle(farm)  =g= contract(farm);

option lp=gamschk;
model farmod /all/
solve farmmod maximizing profit using lp;
```

we get


Acrobat Reader - [GAMSCHK.PDF]
File Edit View Tools Window Help

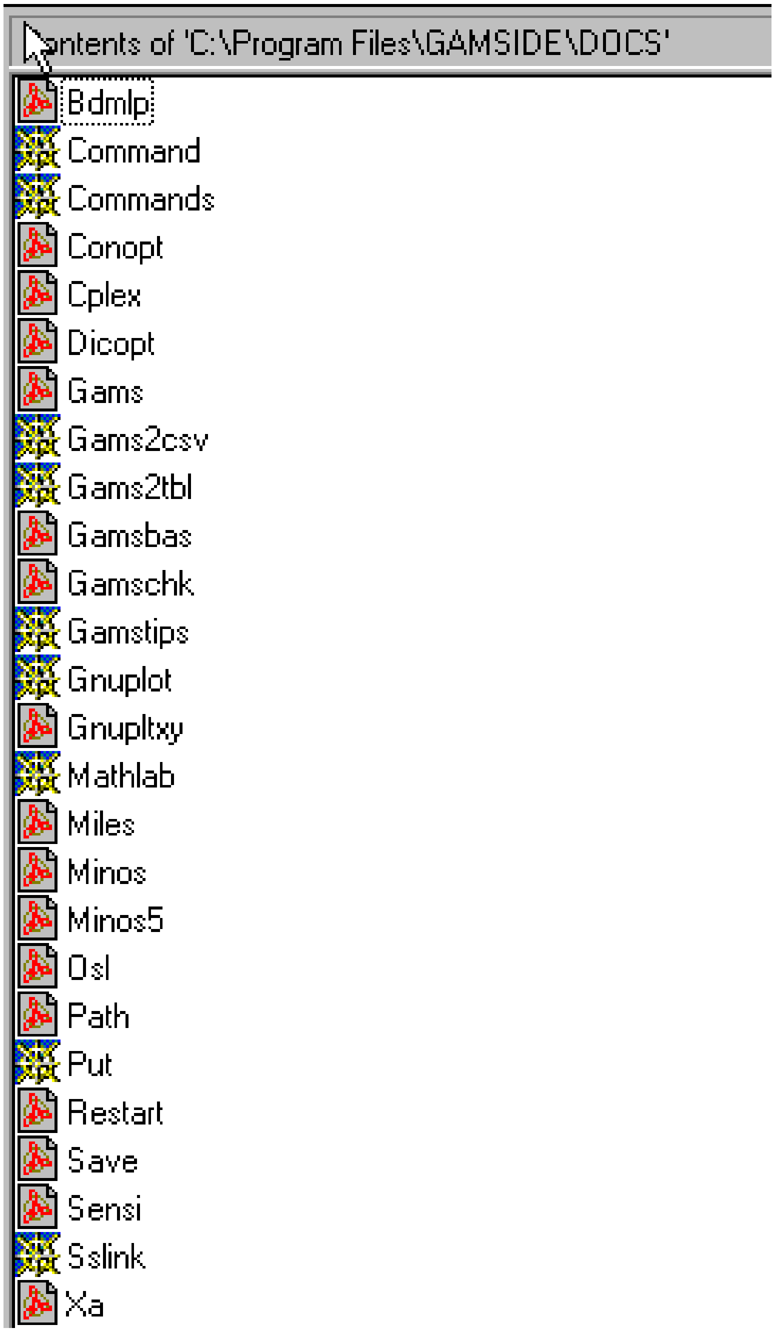
GAMSCHK USER DOCUMENTATION

Version 1.1

A System for Examining the Structure and Solution
Properties of Linear Programming Problems
Solved using GAMS

Accessing documentation on GAMS through the IDE.

In fact we can get any of the following



Contents of 'C:\Program Files\GAMSIDE\DOCS'

Bdmlp
Command
Commands
Conopt
Cplex
Dicopt
Gams
Gams2csv
Gams2tbl
Gamsbas
Gamschk
Gamstips
Gnuplot
Gnupltxy
Mathlab
Miles
Minos
Minos5
Osl
Path
Put
Restart
Save
Sensi
Sslink
Xa

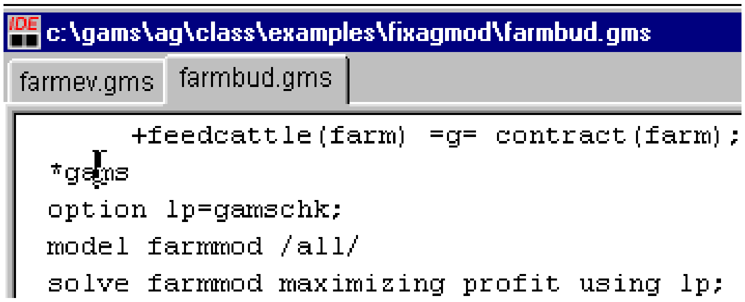
Introduction to GAMS
Accessing documentation on GAMS through the IDE.

The files used are those in the docs directory that were created for this course. You can add more. Any file with a pdf or html extension will work if you add it to docs.

It does not have to be a command GAMS recognizes.

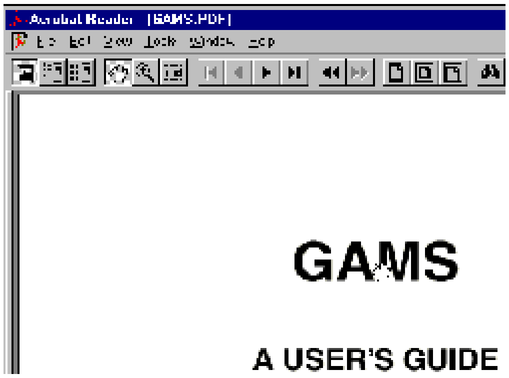
Try putting in the following

*gams



```
      +feedcattle(farm) =g= contract(farm);
*gams
option lp=gamschk;
model farmmod /all/
solve farmmod maximizing profit using lp;
```

In turn you get



GAMS

A USER'S GUIDE