

# Introduction to GAMS

Zeliha ERGÜL AYDIN

[zergul@eskisehir.edu.tr](mailto:zergul@eskisehir.edu.tr)

Department of Industrial Engineering  
Eskisehir Technical University

May 2, 2021

- ▶ The General Algebraic Modeling System (GAMS) is a high-level modeling system for mathematical programming problems. It consists of a language compiler and a stable of integrated high-performance solvers.
- ▶ Different types of optimization models can be solved using GAMS.

# GAMS Model Types

GAMS Model Type	Model Type Description	Requirements and Comments
LP	Linear Program	Model with no nonlinear terms or discrete (i.e. binary, integer, etc) variables.
NLP	Nonlinear Program	Model with general nonlinear terms involving only smooth functions, but no discrete variables. For a classification of functions as to smoothness, see section Functions.
QCP	Quadratically Constrained Program	Model with linear and quadratic terms, but no general nonlinear terms or discrete variables.
DNLP	Discontinuous Nonlinear Program	Model with non-smooth nonlinear terms with discontinuous derivatives, but no discrete variables. This is the same as NLP, except that non-smooth functions may appear as well. These models are more difficult to solve than normal NLP models and we strongly advise not to use this model type.
MIP	Mixed Integer Program	Model with binary, integer, SOS and/or semi variables, but no nonlinear terms.
RMIP	Relaxed Mixed Integer Program	Like MIP, except that the discrete variable requirement is relaxed. See the note below on relaxed model types.
MINLP	Mixed Integer Nonlinear Program	Model with both nonlinear terms and discrete variables.
RMINLP	Relaxed Mixed Integer Nonlinear Program	Like MINLP except that the discrete variable requirement is relaxed. See the note below on relaxed model types.
MIQCP	Mixed Integer Quadratically Constrained Program	Model with both quadratic terms and discrete variables, but no general nonlinear term.
RMIQCP	Relaxed Mixed Integer Quadratically Constrained Program	Like MIQCP except that the discrete variable requirement is relaxed. See the note below on relaxed model types.
MCP	Mixed Complementarity Problem	A square, possibly nonlinear, model that generalizes a system of equations. Rows and columns are matched in one-to-one complementary relationships.
CNS	Constrained Nonlinear System	Model solving a square, possibly nonlinear system of equations, with an equal number of variables and constraints.
MPEC	Mathematical Programs with Equilibrium Constraints	A difficult model type for which solvers and reformulations are currently being developed.
RMPEC	Relaxed Mathematical Program with Equilibrium Constraints	A difficult model type for which solvers and reformulations are currently being developed. See the note below on relaxed model types.
EMP	Extended Mathematical Program	A family of mathematical programming extensions.
MPSGE	General Equilibrium	Not actually a model type but mentioned for completeness, see MPSGE.

# GAMS Solver-Model Matrix

	LP	MIP	NLP	MCP	MPEC	CNS	DNLP	MINLP	QCP	MIQCP	Stoch.	Global
ALPHAECIP								✓		✓		
ANTIGONE			✓			✓	✓	✓	✓	✓		✓
BARON	✓	✓	✓			✓	✓	✓	✓	✓		✓
BDMIP	✓	✓										
BONMIN								✓		✓		
CBC	✓	✓										
CONOPT 3	✓		✓			✓	✓		✓			
CONOPT 4	✓		✓			✓	✓		✓			
COUENNE			✓			✓	✓	✓	✓	✓		✓
CPLEX	✓	✓							✓	✓		
DECIS	✓										✓	
DICOPT								✓		✓		
GLOMIGO									✓	✓		✓
GUROBI	✓	✓							✓	✓		
GUSS	✓	✓	✓	✓		✓	✓	✓	✓	✓		
IPOPT	✓		✓			✓	✓		✓			
KESTREL	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
KNITRO	✓		✓		✓	✓	✓	✓	✓	✓		
LGO	✓		✓				✓		✓			✓
LINDO	✓	✓	✓				✓	✓	✓	✓	✓	✓
LINDOGLOBAL	✓	✓	✓				✓	✓	✓	✓		✓
LOCALSOLVER		✓	✓			✓	✓	✓	✓	✓		
MILES				✓								
MINOS	✓		✓			✓	✓		✓			
MOSEK	✓	✓	✓				✓	✓	✓	✓		
MSNLP			✓						✓			✓
NLPEC				✓	✓							
ODHCPLEX		✓								✓		
PATH				✓		✓						
SBB								✓		✓		
SCIP		✓	✓			✓	✓	✓	✓	✓		✓
SHOT								✓		✓		
SNOPT	✓		✓			✓	✓		✓			
SOLVEENGINE	✓	✓										
SOPLEX	✓											
XA		✓										
XPRESS	✓	✓	✓			✓	✓	✓	✓	✓		

<https://www.gams.com/download/>



G A M S

[Products](#) ▾ [Documentation](#) [Download](#) [Support](#) [Sales](#) ▾ [Community](#) ▾ [About Us](#) ▾ 

## Download GAMS Release 32.2.0 (August 26, 2020)

Please consult the [release notes](#) before downloading a system. Here are the [detailed platform descriptions](#) and [installation notes](#).  
The GAMS distribution includes the [documentation](#) in electronic form.

### Platform

MS Windows	Microsoft Desktop and Server Operating Systems (x86_64 architecture)	<a href="#">Download</a>
Linux	GNU/Linux System (x86_64 architecture)	<a href="#">Download</a>
Mac OS X	Macintosh System (x86_64 architecture)	<a href="#">Download</a>

### Request a Free Demo License

GAMS will not work without a valid license. Please use the form below to request a demo license.

- ▶ Demo (Default) :For linear models (LP, RMIP, and MIP) GAMS will generate and solve models with up to 2000 constraints and 2000 variables.
- ▶ Community: Users can request a free community license from sales@gams.com. The community license lets you generate and solve linear models (LP, MIP, and RMIP) that do not exceed 5000 variables and 5000 constraints.
- ▶ Purchase license.

If your model exceeds Demo/Community license limits and you don't have money: **Neos Solver**.

The NEOS Server offers the IBM ILOG CPLEX Optimizer for the solution of linear programming (LP), mixed-integer linear programming (MILP), and second-order conic programming (SOCP) problems. Acceptable input formats for CPLEX on the NEOS server include AMPL, GAMS, LP, MPS, and NL formats.

# The basic components of a GAMS model

Inputs	<b>Sets</b> Declaration Assignment of members
	<b>Data</b> (Parameters, Tables, Scalars) Declaration Assignment of values
	<b>Variables</b> Declaration Assignment of type
	<b>Assignment of Variable Bounds and/or Initial Values</b> (optional)
	<b>Equations</b> Declaration Definition
	<b>Model and Solve Statements</b>
	<b>Display Statements</b> (optional)



# The basic components of a GAMS model

Outputs	<b>Echo Prints</b>
	<b>Reference Maps</b>
	<b>Equation Listings</b>
	<b>Status Reports</b>
	<b>Solution Reports</b>

# General Rules For Formating and Ordering Statements

- ▶ GAMS is case insensitive. The user-defined name "const" is identical to "CONSt" and "CONST".
- ▶ Individual GAMS statements may be formatted in almost any style. Multiple lines may be used for a statement, blank lines may be embedded and any number of spaces or tabs may be inserted.
- ▶ Each statement should terminated with a semicolon
- ▶ Statements must be ordered such that identifiers are declared before they are used.

# Example

(Diet Problem ,Winston 3.4., p. 69)

$$\text{Min } z = 50x_1 + 20x_2 + 30x_3 + 80x_4$$

s.t

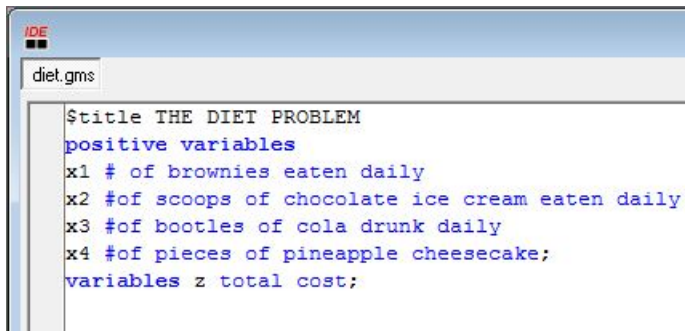
$$400x_1 + 200x_2 + 150x_3 + 500x_4 \geq 500$$

$$3x_1 + 2x_2 \geq 6$$

$$2x_1 + 2x_2 + 4x_3 + 4x_4 \geq 10$$

$$2x_1 + 4x_2 + x_3 + 5x_4 \geq 8$$

$$x_1, x_2, x_3, x_4 \geq 0$$



```
IDE
diet.gms

$title THE DIET PROBLEM
positive variables
x1 # of brownies eaten daily
x2 #of scoops of chocolate ice cream eaten daily
x3 #of bottles of cola drunk daily
x4 #of pieces of pineapple cheesecake;
variables z total cost;
```

- ▶ \$title sets the title of the pages in the file. Optional.
- ▶ x1, x2, x3, x4, z are variable names.
- ▶ "# of brownies eaten daily", "# of scoops of chocolate ice cream eaten daily", "#of bottles of cola drunk daily", "# of pieces of pineapple cheesecake", "total cost" are explanatory text. Explanatory text is optional.

# Equations Declaration

```
equations  
obj objective function  
const1 calorie constraint  
const2 chocolate constraint  
const3 sugar constraint  
const4 fat constraint;
```

- ▶ obj, const1, const2, const3, const4 are equations names.
- ▶ "objective function", "calorie constraint", "chocolate constraint", "sugar constraint", "fat constraint" are explanatory text.  
Explanatory text is optional.

# Equations Definition

```
obj.. z=e=50*x1+20*x2+30*x3+80*x4;  
const1.. 400*x1+200*x2+150*x3+500*x4=g=500;  
const2.. 3*x1+2*x2=g=6;  
const3.. 2*x1+2*x2+4*x3+4*x4=g=10;  
const4.. 2*x1+4*x2+x3+5*x4=g=8;
```

Note:

=e=	Equality: right-hand side must equal left-hand side.
=g=	Greater than: left-hand side must be greater than or equal to right-hand side.
=l=	Less than: left-hand side must be less than or equal to right-hand side.

# Model Statement

```
model diet /all/;
```

or

```
model diet /obj,const1,const2, const3, const4/;
```

- ▶ "diet" is model name.
- ▶ "all" is a shorthand notation to include all known (declared) equations a model.

# Solve Statement

```
solve diet using lp minimizing z;
```

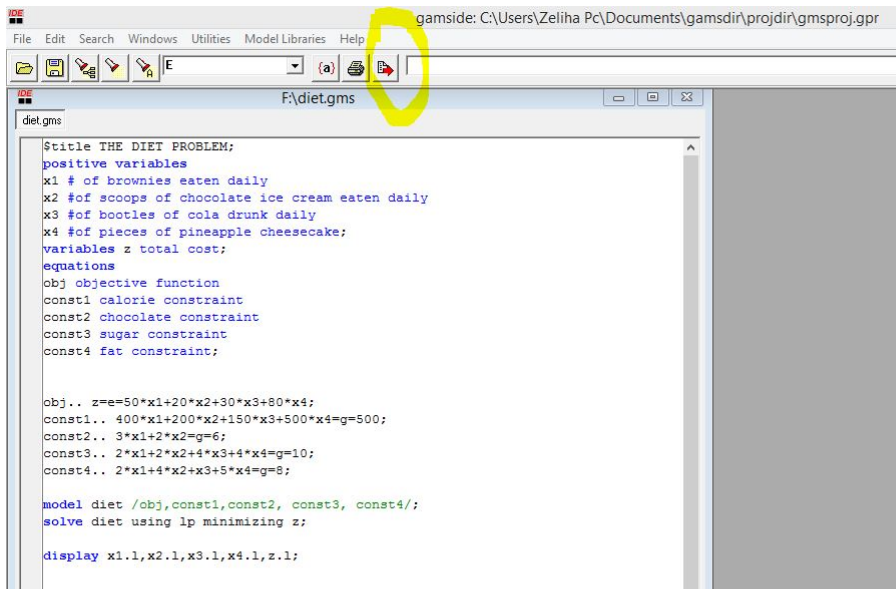
- ▶ "Solve" and "using" are reserved words,
- ▶ "diet" is the model name,
- ▶ "lp" is the model type(linear programming),
- ▶ "minimizing" is the direction of optimization
- ▶ z is the objective variable



# Display Statement

```
display x1.1,x2.1,x3.1,x4.1,z.1;
```

# Running a model



The screenshot shows the GAMS IDE interface. The title bar indicates the file path: `gamside: C:\Users\Zeliha Pc\Documents\gamsdir\projdir\gmsproj.gpr`. The menu bar includes File, Edit, Search, Windows, Utilities, Model Libraries, and Help. The toolbar contains icons for file operations, and a yellow circle highlights the 'Run' button (a red arrow icon). The main window displays the contents of `F:\diet.gms`.

```
$title THE DIET PROBLEM;
positive variables
x1 # of brownies eaten daily
x2 #of scoops of chocolate ice cream eaten daily
x3 #of bottles of cola drunk daily
x4 #of pieces of pineapple cheesecake;
variables z total cost;
equations
obj objective function
const1 calorie constraint
const2 chocolate constraint
const3 sugar constraint
const4 fat constraint;

obj.. z=e=50*x1+20*x2+30*x3+80*x4;
const1.. 400*x1+200*x2+150*x3+500*x4=g=500;
const2.. 3*x1+2*x2=g=6;
const3.. 2*x1+2*x2+4*x3+4*x4=g=10;
const4.. 2*x1+4*x2+x3+5*x4=g=8;

model diet /obj,const1,const2, const3, const4/;
solve diet using lp minimizing z;

display x1.l,x2.l,x3.l,x4.l,z.l;
```

# Output

The screenshot shows the GAMS IDE interface. The main window displays the output of a GAMS model named 'diet.gms'. The output is divided into several sections: a report summary, execution details, and a file summary. The left pane shows the project structure, and the right pane shows the log of the CPLEX solver.

```
**** REPORT SUMMARY :      0  NONOPT
                        0  INFEASIBLE
                        0  UNBOUNDED

GAMS Rev 233 WEX-WEI 23.3.3 x86_64/MS Windows
THE DIET PROBLEM;
Execution

|--- 25 VARIABLE x1.L = 0.000
      VARIABLE x2.L = 3.000
      VARIABLE x3.L = 1.000
      VARIABLE x4.L = 0.000
      VARIABLE z.L = 90.000

EXECUTION TIME = 0.000 SECONDS 2 Mb WE

USER: School of Civil Aviation G100
      Anadolu University
      License for teaching and research at degree grantin

**** FILE SUMMARY
```

The right pane shows the log of the CPLEX solver, indicating that the model was solved successfully. The log includes information about the solver version, the problem size, and the solution status.

```
--- diet.gms(23) 3 Mb
--- Executing CPLEX: elapsed 0:00:00.067
--- diet.gms(23) 3 Mb

ILOG CPLEX Nov 1, 2009 23.3.3 WEX 13908.15
Cplex 12.1.0, GAMS Link 34

Reading data...
Starting Cplex...
Tried aggregator 1 time.
LP Presolve eliminated 1 rows and 1 columns.
Reduced LP has 4 rows, 4 columns, and 14 nonzeros
Presolve time = 0.00 sec.

Iteration Dual Objective In Varia
1 60.000000
2 90.000000
LP status(1): optimal

Optimal solution found.
Objective : 90.000000

--- Restarting execution
--- diet.gms(23) 2 Mb
--- Reading solution for model diet
--- diet.gms(23) 2 Mb
--- Executing after solve: elapsed 0:00:00.118
--- diet.gms(25) 2 Mb
*** Status: Normal completion
--- Job diet.gms Stop 10/31/20 12:52:46 elapsed 0
```

# Sources for GAMS

- ▶ User's Guide: [https://www.gams.com/latest/docs/UG\\_MAIN.html](https://www.gams.com/latest/docs/UG_MAIN.html)
- ▶ A GAMS Tutorial by Richard E. Rosenthal:  
[https://www.gams.com/latest/docs/UG\\_Tutorial.html](https://www.gams.com/latest/docs/UG_Tutorial.html)
- ▶ Teach Yourself GAMS, Deniz AKSEN,