

---



---

# Index

- `..` *see* arithmetic progression
- `;` *see* semicolon statement terminator
- `"..."`, `'...'` *see* quotes
- `>>>` `<<<` *see* error messages
- `#`, `/*...*/` comment 10, 453
- `^`, `**`, exponentiation 111, 456
- `+`, `-` in set data table 156–157, 475
- `>`, `>>` output redirection 220, 239, 251, 273, 481
- `:=` attribute 131, 398, 466, 468
- `=` attribute *see also* defined variables
- `=` attribute 74, 118–120, 211, 462, 465
- `$` column name abbreviation 230
- `~` column name operator 177, 184–185, 188, 190, 478
- `-?` command-line argument 499–500
- `<...>` data specification 196–197, 479
- `..`, default symbol 120, 153, 156–157, 228, 477
- `{}`, empty set 74, 146, 162, 262, 264, 341, 456–457, 462
- `*` in option command 205, 481
- `<` input redirection 163, 165, 481, 485
- `<<...>>`, piecewise-linear expression 367, 369, 376, 380–382, 460
- `%` `printf` conversion character 239, 482
- `->` read/write status 189, 193, 478
- `<-` read/write status 175, 177, 182, 193
- `<->` read/write status 193, 478
- `-` standard input filename 165, 283, 485, 495
- `&` string concatenation operator 270, 459
- `:=` versus `=` initial values 211
  
- `abs` function 113, 140, 380, 458
- Access database 170, 174–175, 178, 192, 194
- Access filename extension, `.mdb` 199–200
- Ackermann's function 126
- `acos` function 113, 458
- `acosh` function 113, 458
- active-set methods 279, 486
- activity model 43
- advertising model 21, 450
- aggregated model 55, 88
- Ahuja, Ravindra K. 347
- airline crew scheduling 362
- Alatorre, Jaime 20
- algebraic modeling language xvii, xxi
- alias function 458
- `AMPL`
  - `AMPL`
    - command-line 203, 499
    - data prompt 144
  - `AMPL:` prompt 5, 165, 204, 485
  - `AMPL?` prompt 165, 204, 485
  - AMPL web page *see* [www.ampl.com](http://www.ampl.com)
  - `AMPL_include` option 479
  - `_AMPL_time` parameter 251, 496
  - AMPLFUNC option 498, 500
  - and operator 114, 456
- `arc`
  - declaration 334–341, 353, 467
  - loss along 326
- argument *see* command-line argument
- arithmetic
  - expression 61, 76, 111–114, 456
  - functions, table of 113, 458
  - logical, set operators, table of 456
  - operators 456
  - operators, table of 112
  - progression 59, 75, 457
  - reduction operators 456
  - to logical conversion 123, 457
- arity function 462, 464
- Aronofsky, Julius S. 20
- array *see* one-dimensional set, parameter
- ASCII filename extension, `.tab` 201, 477
- ASCII set 87, 464

- asin function 113, 458
- asinh function 113, 458
- assignable expression 185
- assignment model 49–51, 53, 135, 330–333, 348
- .astatus suffix 294–295
- astatus\_table option 294
- atan, atan2 function 113, 458
- atanh function 113, 458
- attribute
  - := 131, 398, 466, 468
  - = 74, 118–120, 211, 462, 465
  - binary 117, 122, 131, 441, 465–466
  - circular 83, 98, 146, 463
  - circular by 86, 463
  - coeff 356, 362, 466
  - cover 361, 467
  - default 74, 119–120, 160–161, 462, 465–466, 468, 476–477
  - dimen 93, 96, 462
  - from 334, 336, 338, 341, 353, 361, 363, 467
  - integer 36, 116, 131, 438, 441, 465–466
  - obj 334, 339, 341, 356, 363, 466
  - ordered 83, 98, 146, 463
  - ordered by 86, 463
  - ordered, inheritance of 86, 463
  - pipe 498
  - random 497
  - reversed 86, 463
  - suffix 466, 469–470, 489
  - symbolic 123, 465, 476, 497
  - to 334, 336, 338, 341, 353, 361, 363, 467
  - verbose 200
- auxiliary files, table of 488
  
- bad\_subscripts option 253
- basic solution 287, 289–290, 292, 298, 473
- Beale, E. M. L. xxi
- Benders decomposition 301
- Bentley, Jon L. xxi
- Bertsekas, Dimitri P. 347
- Beta function 459
- bimatrix game 426, 433
- binary filename extension, .bit 201, 477
- binary attribute 117, 122, 131, 441, 465–466
- bipartite graph 330
- Bisschop, Johannes xxi
- .bit binary filename extension 201, 477
- Bixby, Robert E. xxi
- blending
  - model 37–39
  - nonlinearity in 38
- .body *see* constraint body
- bounded variable complementarity 423–424
- bounds on variables 10, 14, 17, 32, 35, 130, 139, 241–242, 276, 279, 466, 470–471, 486
- Braess’s paradox 414
- break statement 266, 493
- breakpoint *see* piecewise-linear expression
- built-in function *see* function
- built-in
  - table handler 174, 201
  - timing parameters, table of 496
- by clause 75, 457
  
- card function 79, 98, 462, 464
- cardinality restrictions 445
- Cartesian product *see* cross product
- caterer problem 347, 350
- Cauchy function 459
- cd command 495
- \_cd current directory 479, 495, 500
- ceil function 212, 237, 458
- char function 459
- check statement 46, 56, 117, 465
- Cherry, Lorinda L. xxi
- Chvátal, Vašek 20, 317
- circular
  - attribute 83, 98, 146, 463
  - by attribute 86, 463
- close command 251, 481, 485
- cmdprompt1, cmdprompt2 option 494
- coeff attribute 356, 362, 466
- collections of
  - data columns, indexed 196
  - sets, indexed 100–104, 161–162
  - tables, indexed 193
- column
  - key 170, 172, 175, 178, 180, 183, 193
  - name, valid 171–172, 177, 185
- column name
  - abbreviation, § 230
  - operator, ~ 177, 184–185, 188, 190, 478
- comma separator in data 144
- command *see also* statement
- command mode *see* model mode
- command
  - language 479–496
  - null 494
- command
  - \* in option 205, 481
  - cd 495
  - close 251, 481, 485

- cont 495
- csvdisplay 484
- data 11, 143–144, 206, 214, 256, 473, 479
- delete 213, 488
- delete check 488
- display 12, 58, 67, 81, 104, 124, 219–238, 482–484
- drop 214, 489
- end 204, 496
- environ 490
- exit 496
- expand 13, 247–248, 492
- fix 215, 489
- include 255–256, 479
- let 36, 210–212, 405, 491
- load 402, 477, 498
- model 11, 206, 214, 256, 479
- next 269, 495
- objective 135–136, 206, 489
- option 204, 481
- print 238–239, 482–484
- printf 124, 239–240, 261, 482–484
- problem 301, 307, 311, 489
- purge 213, 488
- quit 5, 204, 214, 496
- read 163, 484
- read table 175, 177, 180, 477
- redeclare 213, 488
- redeclare check 488
- reload 498
- remove 482
- reset 17, 214, 481, 490
- reset data 122, 209–210, 490
- reset options 205, 490
- reset problem 491
- restore 214, 489
- shell 214, 495
- show 246, 491
- skip 270, 495
- solexpand 248, 492
- solution 487
- solve 206–207, 485–486
- solve problem 315
- step 268, 495
- suffix 296, 299, 302
- unfix 215, 489
- unload 498
- update data 209, 491
- write 487
- write table 178, 186, 477
- xref 213, 247, 492
- command-line
  - AMPL 203, 499
  - argument, -? 499–500
  - argument, -i 500
  - argument, -o 485
  - argument, -R 500
  - argument, -s 458
  - argument, -v 500
- commands
  - stepping through 268–270, 495
  - table of 480
- commands statement 256, 479
- comment, #, /\*...\*/ 10, 453
- comments in data 47, 453, 473
- comparison operators *see* relational operators
- compl\_warn option 487
- complementarity
  - bounded variable 423–424
  - constraint 427–428, 469, 471, 487
  - constraint, suffixes for 428, 431, 471
  - generic names and 431
  - presolve and 429
  - problem 420–423
- complements operator 469
- computed parameters 118–121
- concatenation *see* string concatenation
- operator
- concave functions 383, 393
- conditional declaration *see* if indexing expression
- conditional expression *see* logical expression, if-then-else
- constant expression 455, 457, 461
- constraint
  - body 35, 242, 280, 470
  - complementarity 427–428, 469, 471, 487
  - declaration xv, 2, 5, 9, 16, 137–139, 340, 461, 468–469
  - defining 386, 399, 467
  - double-inequality 32, 139, 470
  - either/or 59
  - equality 21, 30, 130
  - nonlinear 59, 140, 492
  - redundant 276–278
- constraint declaration, to\_come 356, 362, 468
- cont command 495
- continuation *see* line continuation
- continue statement 266, 493
- continuity assumption 38, 391
- control flow statements 492
- conversion from
  - arithmetic to logical 123, 457
  - number to string 271, 459
- conversion characters, table of printf 483
- convex functions 383, 393, 396

- cos function 113, 458
- cost functions, nonlinear 394
- Cottle, Richard W. 432
- Coullard, Collette xxi, 136, 141, 166, 287
- cover attribute 361, 467
- covering problem, set 362, 364
- CPLEX solver 36, 50, 208, 281
- Cramer, Gary I. xxi
- credit scoring model 386
- crew scheduling, airline 362
- cross product 97–98, 456
- cross operator 98, 456
- csvdisplay command 484
- ctime function 458
- cut.mod, file 306
- cutting-stock problem 41, 305, 364, 452
  
- Dantzig, George B. xxi, 39, 52
- .dat file *see* file
- data reset *see* reset data command
- data value, omitted *see* default symbol
- data
  - columns, indexed collections of 196
  - comma separator in 144
  - comments in 47, 453, 473
  - for indexed collection 101, 161
  - format 473–477
  - mode 143, 206, 256, 479
  - multidimensional list 148–151
  - multidimensional tabular 156–160
  - one-dimensional list 145–146
  - parameter 477
  - set 473–475
  - set and parameter 151–154
  - table, set and parameter 476
  - two-dimensional list 146–148
  - two-dimensional tabular 154–156
  - unformatted 163
  - validation 12, 46, 93, 118
  - value, default 160–161
- data specification, < . . . > 196–197, 479
- data command 11, 143–144, 206, 214, 256, 473, 479
- database handler *see* table handler
- data-model separation 7, 11, 74, 143
- dataprompt1, dataprompt2 option 494
- declaration *see also* statement
- declaration
  - arc 334–341, 353, 467
  - constraint xv, 2, 5, 9, 16, 137–139, 340, 461, 468–469
  - node 334–340, 353, 468
  - objective 134–137, 470
  - parameter 8, 110–111, 465–466
  - problem 489
  - set 8, 74, 461–465
  - table 174, 178, 180, 186, 477
  - variable 5, 8, 466–467
- declaration
  - function 402, 497
  - to\_come constraint 356, 362, 468
- declarations, suffix 471
- default read/write status *see* INOUT
- default data value 160–161
- default symbol . 120, 153, 156–157, 228, 477
- default
  - attribute 74, 119–120, 160–161, 462, 465–466, 468, 476–477
  - versus = initial values 119
- defined variables 466–468
- defining constraint 386, 399, 467
- delete
  - check command 488
  - command 213, 488
- derivative 407
- diet model 27–37, 39, 77–78, 135
  - integer solutions in 36, 439
- diff operator 76, 98, 456–457
- differentiability assumption 391
- dimen attribute 93, 96, 462
- directed network graph 320
- .direction suffix 296
- Dirkse, Steven P. 433
- \_display command 484
- display
  - command 12, 58, 67, 81, 104, 124, 219–238, 482–484
  - formatting options 227–230
  - formatting options, table of 227
  - numeric options 232–238
  - numeric options, table of 232
- Display set 87, 464
- display\_1col option 63, 228, 231
- display\_eps option 235
- display\_precision option 233, 263, 406, 483
- display\_round option 234–235, 483
- display\_transpose option 229
- display\_width option 204, 229
- displaying
  - indexing expression 224–227
  - parameters 220–224
  - sets 220
- div operator 111, 456
- double-inequality constraint 32, 139, 470
- drop command 214, 489

- Drud, Arne xxi
- .dual *see* dual value
- dual
  - value 17, 22, 40, 66–67, 162, 243–245, 259, 262, 266, 470, 482
  - variable 398
  - variables 162, 243, 468, 472, 485
  - variables, initial values of 185, 468, 472, 477, 486
- dual\_initial\_guesses option 486
- dummy index 31, 47, 79, 92, 94–95, 99, 138, 184, 188, 190–191, 197, 455, 463
  - scope of 47, 321
- Dutta, Goutam 20
- Dutton, John M. 20
  
- EBCDIC set 87, 464
- economic
  - analysis 42, 415
  - equilibrium model 419, 422
- eexit option 252
- either/or constraint 59
- Emlin, Grace R. xxi
- empty
  - key specification 182
  - set, indexing over 82, 146
  - string 205, 234, 252, 272
  - table entry 173, 187
- empty set { } 74, 146, 162, 262, 264, 341, 456–457, 462
- end effect 67, 379
- end command 204, 496
- environ command 490
- environment
  - options 490
  - variables 205, 481
  - variables, table of 484
- environment, Initial 316, 490
- environments, named 316–317
- equality constraint 21, 30, 130
- equilibrium stress model 388
- Erikson, Warren 39
- error messages 12, 83, 85, 94, 116, 119, 146, 160, 207, 211, 224, 252, 338
  - \ escape sequences, printf 482
- evaluation, order of 100, 111, 132, 467, 473
- Excel spreadsheet 174, 176, 196
- Excel filename extension, .xls 199–200
- exists operator 115, 456–457
- exit command 496
- exp function 113, 458
- expand command 13, 247–248, 492
- expand\_precision option 484
- expand\_round 484
- Exponential function 459
- exponentiation  $^$ ,  $**$  111, 456
- expression
  - arithmetic 61, 76, 111–114, 456
  - assignable 185
  - constant 455, 457, 461
  - indexing 9, 79–82, 99–100, 110, 138, 455
  - linear 132–134, 139, 383
  - logical 80, 99, 114–116, 455, 457
  - metacharacters, regular 460
  - nonlinear 114, 133, 400–403
  - regular 273, 460
  - set 455, 457
  - simplification of 46, 130, 134, 139
  - string 194, 197, 273, 459
- expression <<...>>, piecewise-linear 367, 369, 376, 380–382, 460
  
- Fabian, Tibor 20
- factorial 126
- feasibility tolerance 281
- feasible solutions 4, 35
- Ferris, Michael C. 433
- Fibonacci numbers 126
- file inclusion *see* model, data, include, commands
- file output *see* output redirection
- file, temporary 485
- file
  - blend.mod 38
  - cut2.mod 313
  - cut.dat 309
  - cut.mod 306
  - diet2.dat 34
  - diet.dat 33
  - diet.mod 32
  - dietu.dat 82
  - dietu.mod 78
  - econ2.mod 424
  - econ.dat 421
  - econmin.mod 420
  - econ.mod 423
  - econnl.mod 425
  - fence.mod 411
  - iocol1.mod 356
  - iocol2.mod 358
  - iorow.mod 355
  - minmax.mod 381
  - multic.mod 103
  - multi.dat 58
  - multi.mod 57
  - multmpl.dat 443

- multmip1.mod 442
- multmip3.dat 447
- multmip3.mod 446
- net1.mod, net1.dat 322
- net1node.mod 335
- net2.dat 325
- net2.mod 324
- net3.dat 327
- net3.mod 326
- net3node.mod 336
- netasgn.mod 333
- netfeeds.mod 344
- netmax3.mod 339
- netmax.mod 329
- netmcol.mod 363
- netmulti.mod 342
- netshort.mod 331
- netthru.mod 346
- nltrans.mod, nltrans.dat 404
- prod.dat 10
- prod.mod 8
- sched.dat 361
- sched.mod 361
- steel2.dat 14
- steel3.mod, steel3.dat 15
- steel4.dat 17
- steel4.mod 16
- steel.mod, steel.dat 11
- steelP.dat 65
- steelpl1.dat 374
- steelpl1.mod 373
- steelP.mod 64
- steelT0.mod 59
- steelT2.dat 85
- steelT2.mod 84
- steelT3.dat 102
- steelT3.mod 101
- steelT.mod, steelT.dat 62
- transp2.mod 93
- transp3.dat 95
- transp3.mod 95
- transp.dat 48
- transpl1.mod 368
- transpl2.dat 371
- transpl2.mod 370
- transp.mod 47
- filename syntax 11, 144, 165, 204, 479
- filename
  - standard input 165, 283, 485, 495
  - extension, .bit binary 201, 477
  - extension, .mdb Access 199–200
  - extension, .tab ASCII 201, 477
  - extension, .xls Excel 199–200
- files, table of auxiliary 488
- first function 84, 463
- Fisher, Marshall L. 317
- fix command 215, 489
- fixed
  - costs 440–444
  - variables 131, 241, 402
- Fletcher, Roger 410
- floating-point representation 76, 87, 144, 233, 235, 263, 271, 279, 453–454, 483
- floor function 212, 457–458
- flow *see* network flow
- for statement 258–262, 493
- forall operator 115, 456–457
- Ford, Lester R. Jr. 347
- formatting options
  - display 227–230
  - table of display 227
- Fourer, Robert xxi, 20, 318, 384
- from attribute 334, 336, 338, 341, 353, 361, 363, 467
- Fulkerson, D. R. 347
- function
  - abs 113, 140, 380, 458
  - acos 113, 458
  - acosh 113, 458
  - alias 458
  - arity 462, 464
  - asin 113, 458
  - asinh 113, 458
  - atan, atan2 113, 458
  - atanh 113, 458
  - Beta 459
  - card 79, 98, 462, 464
  - Cauchy 459
  - ceil 212, 237, 458
  - char 459
  - cos 113, 458
  - ctime 458
  - exp 113, 458
  - Exponential 459
  - first 84, 463
  - floor 212, 457–458
  - Gamma 459
  - gsub 273, 460
  - ichar 459
  - indexarity 462, 464
  - Irand224 459
  - last 84, 463
  - length 271, 459
  - log, log10 113, 458
  - match 271, 285, 460
  - max 113, 380, 458
  - member 463
  - min 113, 140, 458

- next 84–85, 463
- nextw 84–85, 463
- Normal, Normal01 209, 459
- num, num0 460
- ord, ord0 85, 463
- Poisson 459
- precision 458
- prev, prevw 83–84, 463
- round 212, 458
- sin 113, 458
- sinh 113, 458
- sprintf 271, 459, 483
- sqrt 113, 458
- sub 273, 460
- substr 271, 459
- tan 113, 458
- tanh 113, 458
- time 458
- trunc 212, 458
- Uniform, Uniform01 459
- function declaration 402, 497
- functions
  - concave 383, 393
  - convex 383, 393, 396
  - imported 497–499
  - of sets 83–85
  - of sets, table of 464
  - piecewise-linear 365, 379–384, 393, 395
  - separable 381, 383
  - smooth 391, 400–402
  - table of arithmetic 113, 458
  - table of random number 459
  - table of regular expression 459
  - table of rounding 212
  - table of string 459
- Gamma function 459
- Garille, Susan Garner 39
- Gass, Saul I. 39
- Gassmann, H. I. xxi
- Gay, David M. 318
- generic names 249, 291, 429, 492
  - and complementarity 431
  - table of 493
- gentimes option 250
- Gill, Philip E. 410
- Gilmore-Gomory algorithm 305–306
- global optimum 408
- Glover, Fred 347
- graph
  - bipartite 330
  - directed network 320
  - transitive closure of 462
- Grosse, Eric H. xxi
  - gsub function 273, 460
  - gutter\_width option 230
- Haessler, Robert W. 318
- Hager, W. W. 318
- handler *see* table handler
  - \_handler\_lib 477
  - \_HANDLERS 477
- Hearn, D. W. 318
- Hilal, Said S. 39
- i command-line argument 500
- ichar function 459
- IEEE arithmetic *see* floating-point representation
- if indexing expression 138, 338, 341, 362, 460–461, 467
- if-then-else
  - nested 266
  - operator 115, 117, 133, 264, 457
  - statement 264–266, 493–494
- .iis suffix 299
- iisfind, iis\_table option 299
- imported functions 497–499
- IN, INOUT suffix attribute 472
- in operator 78, 98, 114, 455–457, 465
- IN read/write status 174, 177, 180, 182, 186, 478
- include *see also* model, data include
  - command 255–256, 479
  - search path 479
- index *see* dummy index
- indexarity function 462, 464
- indexed
  - collection, data for 101, 161
  - collections of data columns 196
  - collections of sets 100–104, 161–162
  - collections of tables 193
  - objectives 134
- indexing
  - expression 9, 79–82, 99–100, 110, 138, 455
  - expression, displaying 224–227
  - expression, scope of 80, 94–95, 455–456, 482
  - over empty set 82, 146
- indexing expression, if 138, 338, 341, 362, 460–461, 467
- infeasibility in presolve 279
- infeasible solutions 34, 299, 374
- infinite
  - loop 262, 267

- sets 86, 463
- Infinity 242, 276, 338, 456, 466
- inheritance of ordered attribute 86, 463
- Initial
  - environment 316, 490
  - problem 311, 315, 489
- initial values
  - of dual variables 185, 468, 472, 477, 486
  - of variables 131, 162, 398, 405, 408–409, 466, 471, 477, 486
- initial values
  - := versus = 211
  - default versus = 119
- initialization, suffix 466, 469–470, 489
- INOUT read/write status 186, 193, 195
- input redirection, < 163, 165, 481, 485
- input-output model 37, 42, 354–358
- insertprompt option 268, 479
- integer
  - program, relaxation of 309, 448
  - programming xvi, 36, 437–438, 448–449
  - solution, rounding to 35, 309, 360, 439, 448
  - solutions in diet model 36, 439
  - solutions in network model 344
  - solutions in scheduling model 360
  - solutions in transportation model 51, 439–448
- integer
  - attribute 36, 116, 131, 438, 441, 465–466
  - set 86, 463–464
- integer\_markers option 486
- Integers set 87, 464
- inter operator 76, 98, 103, 456–457
- interior point algorithm 287
- intersection, set *see* inter
- interval set 86, 463–464
- Irand224 function 459
- irreducible infeasible subset 299
- iterated operator *see* operator
  
- Jacobs, Walter 347, 351
- Johnson, Ellis L. 449
  
- Kahan, Gerald 363
- Kelly, Paul xxi
- Kendrick, David A. 20
- Kernighan, Mark D. xxi
- key
  - column 170, 172, 175, 178, 180, 183, 193
  - specification, empty 182
- keyword *see* reserved word
- Klingman, Darwin 347
- knapsack problem 23, 306, 451
- Kontogiorgis, Spyros 384
- Kostreva, Michael M. 449
- Kuhn, Harold W. xxi
- Kuip, C. A. C. xxi
  
- Lagrange multiplier *see* dual variables
- large scale optimization xvi
- Lasdon, Leon 318, 414
- last function 84, 463
- .lb lower bound 35, 241–242, 279, 470, 486
- .lb0, .lb1, .lb2 lower bound 279
- least squares 388, 412
- length function 271, 459
- Lenstra, Jan Karel xxi
- less operator 111, 456
- let command 36, 210–212, 405, 491
- library *see* imported functions
- Liebman, Judith 414
- line continuation \ 205, 453
- linear
  - expression 132–134, 139, 383
  - program, network 319, 343, 353
  - programming xvi, 1–5, 129
- linearity of costs, yield 38, 393
- linelim option 467
- linked linear programs 55, 60
- list data
  - multidimensional 148–151
  - one-dimensional 145–146
  - two-dimensional 146–148
- literal
  - number 75, 111, 144, 453–454
  - set 74, 454
  - string 74, 144, 453–454, 473
- load command 402, 477, 498
- local optimum 407–408
- LOCAL suffix attribute 472
- log, log10 function 113, 458
- log\_file option 251, 496
- log\_model, log\_data option 252, 496
- logical
  - arithmetic, set operators, table of 456
  - conversion, arithmetic to 123, 457
  - expression 80, 99, 114–116, 455, 457
  - operators 114, 457
  - parameters 122–123
- loop
  - infinite 262, 267
  - name 268, 494
  - nested 261, 267
- loss in network flow 326, 336
- Lowe, Todd xxi

- lower bound
  - .1b 35, 241–242, 279, 470, 486
  - .1b0, .1b1, .1b2 279
- Magnanti, Thomas L. 347
- marginal value *see* dual value
- Marsten, Roy E. 384
- match function 271, 285, 460
- mathematical programming xv–xvi
- matrix *see* two-dimensional set, parameter
- matrix generator xvii, xxi
- max
  - function 113, 380, 458
  - operator 114, 456
- maximize *see* objective declaration
- maximum flow model 328–329, 337–339, 412
- max-min objective 382
  - .mcb Access filename extension 199–200
- Meeraus, Alexander xxi, 20
- member, dummy *see* dummy index
- member function 463
- membership test *see* in, within
- membership operators, set 78–79
- memory use 88, 97, 448
- messages *see* error messages
- metacharacters, regular expression 460
- min
  - function 113, 140, 458
  - operator 114, 456
- minimize *see* objective declaration
- min-max objective 382
- MINOS solver 6, 208, 281, 398
- mixed-integer programming 441
  - .mod file *see* file
- mod operator 111, 456
- model file *see* file
- model
  - activity 43
  - advertising 21, 450
  - aggregated 55, 88
  - assignment 49–51, 53, 135, 330–333, 348
  - blending 37–39
  - credit scoring 386
  - diet 27–37, 39, 77–78, 135
  - economic equilibrium 419, 422
  - equilibrium stress 388
  - input-output 37, 42, 354–358
  - maximum flow 328–329, 337–339, 412
  - mode 143, 206, 256, 479, 496
  - multicommodity transportation 56–59, 450
  - multi-period planning 67–69, 452
  - multi-period production 59–63
  - network flow 319, 333–339, 416
  - nonlinear transportation 403–410
  - oil refinery 24–25
  - piecewise-linear production 370–377
  - piecewise-linear transportation 366–369
  - production 6–12, 15, 420
  - production and transportation 63–67
  - scheduling 37, 359–362, 451
  - shortest path 329–330
  - steel production 2–7, 10–19
  - stress analysis 388
  - transportation 44–49, 52, 330–333
  - transshipment 319–327, 334–337, 347
- model command 11, 206, 214, 256, 479
- model-data separation 7, 11, 74, 143
- modeling language xvi–xviii
- MPEC system 426
- MPS file 486
- multicommodity transportation model 56–59, 450
- multidimensional
  - list data 148–151
  - tabular data 156–160
- multi-period
  - planning model 67–69, 452
  - production model 59–63
- multiple
  - objectives 134–137, 206, 240
  - solutions 6, 49, 137, 407
- Murray, Walter 410
- Murty, Katta G. 347
- name
  - loop 268, 494
  - syntax 453
- named
  - environments 316–317
  - problems 309–316
- names
  - generic 249, 291, 429, 492
  - table of generic 493
- Nash equilibrium 426, 433
- Nemhauser, George L. 450
- NEOS xx
- nested loop 261, 267
- nested if-then-else 266
- net\_in, net\_out 334, 340, 469
- network
  - flow, loss in 326, 336
  - flow model 319, 333–339, 416
  - graph, directed 320
  - linear program 319, 343, 353
  - model, integer solutions in 344

- newline 144, 453, 473, 498
- \n newline in printf 239, 261, 482
- next
  - command 269, 495
  - function 84–85, 463
- nextw function 84–85, 463
- nl\_permute option 487
- Nocedal, Jorge 410
- node declaration 334–340, 353, 468
- non-existent data value *see* default symbol
- nonlinear
  - constraint 59, 140, 492
  - cost functions 394
  - expression 114, 133, 400–403
  - programming xvi, 391–397, 403–410
  - transportation model 403–410
  - variables 397–400
- nonlinearity in
  - blending 38
  - physical models 397
- nonnegative variables 3, 10, 45, 130, 377
- Normal, Normal01 function 209, 459
- not
  - in operator 79, 456
  - operator 114, 456
  - within operator 79, 456
- null command 494
- num, num0 function 460
- number representation *see* floating-point representation
- number
  - literal 75, 111, 144, 453–454
  - to string, conversion from 271, 459
- numbers, set of 75–76
- numeric to logical conversion 457
- numeric options, display 232–238
  
- o command-line argument 485
- obj attribute 334, 339, 341, 356, 363, 466
- objective 3, 9
  - declaration 134–137, 470
  - function xv, 31, 240, 341
  - max-min, min-max 382
- objective function, to\_come 362, 470
- objective command 135–136, 206, 489
- objective\_precision option 484–485
- objectives
  - indexed 134
  - multiple 134–137, 206, 240
- ODBC table handler 169, 174, 194, 198
- oil refinery model 24–25
- omit\_zero\_cols option 66, 232
- omit\_zero\_rows option 50, 66, 231
  
- omitted data value *see* default symbol
- one-dimensional
  - list data 145–146
  - parameter 110
  - set 110, 454
  - set data 473
- O'Neill, Richard P. 411
- operator
  - precedence 9, 77, 111–114, 116, 133, 456, 459
  - synonyms 454
- operator
  - ~ column name 177, 184–185, 188, 190, 478
  - & string concatenation 270, 459
  - and 114, 456
  - complements 469
  - cross 98, 456
  - diff 76, 98, 456–457
  - div 111, 456
  - exists 115, 456–457
  - forall 115, 456–457
  - if-then-else 115, 117, 133, 264, 457
  - in 78, 98, 114, 455–457, 465
  - inter 76, 98, 103, 456–457
  - less 111, 456
  - max 114, 456
  - min 114, 456
  - mod 111, 456
  - not 114, 456
  - not in 79, 456
  - not within 79, 456
  - or 114, 456
  - prod 74, 114, 456
  - setof 98, 272, 457
  - sum 74, 113, 456
  - syndiff 76, 98, 456–457
  - union 76, 98, 103, 456–457
  - within 78, 93, 97–98, 114, 456–457, 462
- operators
  - arithmetic 456
  - arithmetic reduction 456
  - logical 114, 457
  - relational 114, 465
  - set 76–78
  - set membership 78–79
  - table of arithmetic 112
  - table of arithmetic, logical, set 456
  - tuple 98–100
- optimal basis *see* basic solution
- optimum
  - global 408
  - local 407–408
- option

- ampl\_include 479
- AMPLFUNC 498, 500
- astatus\_table 294
- bad\_subscripts 253
- cmdprompt1, cmdprompt2 494
- compl\_warn 487
- dataprompt1, dataprompt2 494
- display\_lcol 63, 228, 231
- display\_eps 235
- display\_precision 233, 263, 406, 483
- display\_round 234–235, 483
- display\_transpose 229
- display\_width 204, 229
- dual\_initial\_guesses 486
- eexit 252
- expand\_precision 484
- gentimes 250
- gutter\_width 230
- iisfind, iis\_table 299
- insertprompt 268, 479
- integer\_markers 486
- linelim 467
- log\_file 251, 496
- log\_model, log\_data 252, 496
- nl\_permute 487
- objective\_precision 484–485
- omit\_zero\_cols 66, 232
- omit\_zero\_rows 50, 66, 231
- pl\_linearize 383
- presolve 139, 278, 290
- presolve\_eps, presolve\_epsmax 281
- presolve\_fixeps, presolve\_fixepsmax 282
- presolve\_inteps, presolve\_intepsmax 279
- presolve\_warnings 252, 282
- print\_precision 239, 482–483
- print\_round 239, 482–483
- print\_separator 238, 482
- prompt1, prompt2 204, 494
- randseed 122, 458
- relax\_integrality 215, 217, 309, 486
- reset\_initial\_guesses 291, 297, 486
- send\_statuses 288
- send\_suffixes 486
- SHELL 495
- shell\_env\_file 495
- show\_stats 250, 275, 282
- single\_step 268
- solution\_precision 236, 238, 264, 484, 486
- solution\_round 236, 238, 484, 486
- solve\_exitcode, solve\_exitcode\_max 285
- solve\_message 285
- solve\_result, solve\_result\_num 283–284
- solve\_result\_table 283
- solver 37, 207
- solver\_msg 260
- sstatus\_table 289
- substout 399, 414, 467–468
- times 250
- TMPDIR 285, 485
- var\_bounds 279, 486
- version 500
- option command 204, 481
  - \* in 205, 481
- option-name pattern 481
- options 204–205, 481
  - environment 490
- options
  - display formatting 227–230
  - display numeric 232–238
  - table of display formatting 227
  - table of display numeric 232
- options, reset 205, 490
- OPTIONS\_IN, OPTIONS\_OUT, OPTIONS\_INOUT 490, 499
- \$option value 460, 481
- \$\$option default value 481
- or operator 114, 456
- ord, ord0 function 85, 463
- order of evaluation 100, 111, 132, 467, 473
- ordered sets 82–87, 98, 223, 261, 463–464
- ordered
  - attribute 83, 98, 146, 463
  - attribute, inheritance of 86, 463
  - by attribute 86, 463
- Orlin, James B. 347
- OUT
  - read/write status 177, 186, 190, 192, 478
  - suffix attribute 472
- output redirection, >, >> 220, 239, 251, 273, 481
- pairs
  - sets of ordered 91–92
  - slices of 93–96
  - subsets of 93–96
- Pang, Jong-Shi 432–433
- parameter

- and set data table 476
- data 477
- data statement 475
- data table 154
- data template 147–148, 150, 475
- declaration 8, 110–111, 465–466
- definition, recursive 120, 122, 465
- one-dimensional 110
- two-dimensional 110
- parameters
  - computed 118–121
  - displaying 220–224
  - logical 122–123
  - randomly generated 121–122, 209
  - restrictions on 116–118
  - symbolic 123–124, 323, 328
  - table of built-in timing 496
- Pardalos, P. M. 318
- PATH solver 422
- pattern, option-name 481
- penalty function 369–377, 385, 396
- piecewise-linear 372, 375–376
- Phillips, Nancy 347
- physical models, nonlinearity in 397
- `_pid` process ID 495
- piecewise-linear
  - function at zero 367, 376
  - functions 365, 379–384, 393, 395
  - penalty function 372, 375–376
  - production model 370–377
  - transportation model 366–369
- piecewise-linear expression `<<...>>` 367, 369, 376, 380–382, 460
- Pike, Ralph 411
- pipe attribute 498
- `pl_linearize` option 383
- planning model, multiperiod 67–69, 452
- Poisson function 459
- `pre` status 276
- precedence, operator 9, 77, 111–114, 116, 133, 456, 459
- precision function 458
- predefined words 74, 112, 454
- redefinition of 213, 454
- presolve 60, 65, 207, 241, 275–282, 486
  - and complementarity 429
  - infeasibility in 279
- presolve option 139, 278, 290
- presolve\_eps, presolve\_epsmax option 281
- presolve\_fixeps, presolve\_fixepsmax option 282
- presolve\_inteps, presolve\_intepsmax option 279
- presolve\_warnings option 252, 282
- `prev`, `prevw` function 83–84, 463
- price, shadow *see* dual value
- primal variables 243, 398, 485
- `print` command 238–239, 482–484
- `print_precision` option 239, 482–483
- `print_round` option 239, 482–483
- `print_separator` option 238, 482
- `printf`
  - command 124, 239–240, 261, 482–484
  - conversion character, % 239, 482
  - conversion characters, table of 483
  - \ escape sequences 482
  - string quotes, %q, %Q 483
- `.priority` suffix 296
- problem *see also* model
- problem
  - caterer 347, 350
  - complementarity 420–423
  - cutting-stock 41, 305, 364, 452
  - declaration 489
  - knapsack 23, 306, 451
  - set covering 362, 364
- problem, Initial 311, 315, 489
- problem command 301, 307, 311, 489
- problems, named 309–316
- prod operator 74, 114, 456
- production
  - and transportation model 63–67
  - model 6–12, 15, 420
  - model in algebraic form 7
  - model, multiperiod 59–63
  - model, piecewise-linear 370–377
- programming
  - integer xvi, 36, 437–438, 448–449
  - linear xvi, 1–5, 129
  - mathematical xv–xvi
  - mixed-integer 441
  - nonlinear xvi, 391–397, 403–410
  - stochastic 69–71
- prompt
  - `ampl:` 5, 165, 204, 485
  - `ampl?` 165, 204, 485
  - `ampl` data 144
- `prompt1`, `prompt2` option 204, 494
- purge command 213, 488
- %q, %Q `printf` string quotes 483
- qualified name *see* suffix
- query, SQL 184, 199
- quit command 5, 204, 214, 496
- quotes 74, 144, 165, 204–205, 453, 473

- R command-line argument 500
- random number
  - functions, table of 459
  - generator seed 122, 210, 458
- random attribute 497
- randomly generated parameters 121–122, 209
- randseed option 122, 458
- .rc *see* reduced cost
- read
  - command 163, 484
  - table command 175, 177, 180, 477
- reading files *see* model, data
- reading scalar from table 182
- read/write status
  - > 189, 193, 478
  - <- 175, 177, 182, 193
  - <-> 193, 478
  - IN 174, 177, 180, 182, 186, 478
  - INOUT 186, 193, 195
  - OUT 177, 186, 190, 192, 478
- Reals set 87, 464
- recursive
  - parameter definition 120, 122, 465
  - set definition 462
- redeclare
  - check command 488
  - command 213, 488
- redefinition of predefined words 213, 454
- redirection
  - >, >> output 220, 239, 251, 273, 481
  - < input 163, 165, 481, 485
- reduced cost 17, 22, 40, 67, 243–245
- reduction operators, arithmetic 456
- redundant constraint 276–278
- regular expression 273, 460
  - functions, table of 459
  - metacharacters 460
- relational
  - operators 114, 465
  - table 170–171, 173, 178
- .relax suffix 486
- relax\_integrality option 215, 217, 309, 486
- relaxation of integer program 309, 448
- reload command 498
- remove command 482
- repeat statement 262–264, 493
- representation, floating-point 76, 87, 144, 233, 235, 263, 271, 279, 453–454, 483
- reserved
  - word synonyms 454
  - words 74, 112
  - words, table of 454
- reset
  - command 17, 214, 481, 490
  - data command 122, 209–210, 490
  - options command 205, 490
  - problem command 491
- reset\_initial\_guesses option 291, 297, 486
- restore command 214, 489
- restrictions on parameters 116–118
- .result suffix 287
- reversed attribute 86, 463
- reversible activities 377–379
- Richton, Robert E. xxi
- Rinnooy Kan, Alexander H. G. xxi
- roll trim problem *see* cutting-stock problem
- round function 212, 458
- rounding 76, 87, 233, 237, 263, 271, 279, 454, 457, 483
  - functions, table of 212
  - to integer solution 35, 309, 360, 439, 448
- s command-line argument 458
- Saunders, Michael A. xxi
- scalar from table, reading 182
- scaling along arc *see* loss in network flow
- scaling of variables 410
- scheduling
  - airline crew 362
  - model 37, 359–362, 451
  - model, integer solutions in 360
- Schrage, Linus 414
- Schrijver, Alexander xxi, 450
- scope *see also* dummy index
- scope of
  - dummy index 47, 321
  - indexing expression 80, 94–95, 455–456, 482
- script 18, 179, 195, 255, 304, 485, 495
- script
  - cut.run 308
  - diet.run 284
  - dietu.run 256
  - steelT.sa1 258
  - steelT.sa2 259
  - steelT.sa3 260
  - steelT.sa4 263
  - steelT.sa5 266
  - steelT.sa7 267
  - steelT.tab1 261
- search path, include 479
- seed, random number generator 122, 210, 458
- Seip, Robert H. xxi
- self-reference 117

- semicolon statement terminator 9, 144, 204, 256, 453, 473, 494
- send\_statuses option 288
- send\_suffixes option 486
- sensitivity analysis 194, 256, 262, 304
- separable functions 381, 383
- separator in data, comma 144
- set difference *see* diff, symdiff
- set intersection *see* inter
- set of sets *see* indexed collection
- set
  - and parameter data 151–154
  - and parameter data table 476
  - arithmetic, logical operators, table of 456
  - covering problem 362, 364
  - data 473–475
  - data, one-dimensional 473
  - data statement 474
  - data table 474
  - data template 147, 150, 474
  - data, two-dimensional 475
  - declaration 8, 74, 461–465
  - definition, recursive 462
  - expression 455, 457
  - literal 74, 454
  - member rules 454
  - membership operators 78–79
  - of numbers 75–76
  - one-dimensional 110, 454
  - operators 76–78
  - two-dimensional 110, 455
- set
  - ASCII 87, 464
  - data table, +, - in 156–157, 475
  - Display 87, 464
  - EBCDIC 87, 464
  - integer 86, 463–464
  - Integers 87, 464
  - interval 86, 463–464
  - Reals 87, 464
- setof operator 98, 272, 457
- sets
  - displaying 220
  - functions of 83–85
  - indexed collections of 100–104, 161–162
  - infinite 86, 463
  - of ordered pairs 91–92
  - of tuples 96–98
  - ordered 82–87, 98, 223, 261, 463–464
  - unordered 73–74
- shadow price *see* dual value
- shell command 214, 495
- SHELL option 495
- shell script 500
- shell\_env\_file option 495
- Shepherd, J. R. xxi
- shortest path model 329–330
- show command 246, 491
- show\_stats option 250, 275, 282
- side
  - constraint 342, 345, 363
  - variable 343
- significant figures 233
- simplex algorithm xvi, xxi, 279, 287, 344, 486
- simplification of expression *see also* presolve
- simplification of expression 46, 130, 134, 139
- sin function 113, 458
- single\_step option 268
- single-step mode 495
- singularity 406–407
- sinh function 113, 458
- Sinha, Lakshman P. xxi
- skip command 270, 495
- .slack suffix 81, 241–242, 244, 472
- slice 58, 91, 95, 97, 147–149, 455
- slices of pairs 93–96
- slope *see* piecewise-linear expression
- smooth functions 391, 400–402
- SNOPT solver 48, 208
- social accounting matrix 415
- soft constraints *see* penalty function
- solexpand command 248, 492
- solution, basic 287, 289–290, 292, 298, 473
- solution command 487
- solution\_precision option 236, 238, 264, 484, 486
- solution\_round option 236, 238, 484, 486
- solutions
  - feasible 4, 35
  - infeasible 34, 299, 374
  - multiple 6, 49, 137, 407
- solve
  - command 206–207, 485–486
  - problem command 315
- solve\_exitcode,
  - solve\_exitcode\_max option 285
- solve\_message option 285
- solve\_result, solve\_result\_num option 283–284
- solve\_result\_table option 283
- \_solve\_time parameter 251, 496
- solver
  - CPLEX 36, 50, 208, 281
  - MINOS 6, 208, 281, 398
  - PATH 422

- SNOPT 48, 208
  - status suffixes 286, 288
- solver status, `.status` 473
- solver option 37, 207
- `solver_msg` option 260
- solver-defined suffixes 298–304
- space requirement *see* memory use
- sparse
  - set 111
  - subset 94
- spreadsheet 238
  - Excel 174, 176, 196
- `sprintf` function 271, 459, 483
- SQL query 184, 199
- SQL SELECT statement 199
- `sqrt` function 113, 458
- square system 422, 425, 427, 487
- `.sstatus` suffix 290, 292, 295, 473
- `sstatus_table` option 289
- `s.t.` *see* constraint declaration
- standard input filename, `-` 165, 283, 485, 495
- starting guess *see* initial values of variables
- statement *see also* command, declaration
- statement
  - parameter data 475
  - set data 474
  - terminator, semicolon 9, 144, 204, 256, 453, 473, 494
- statement
  - break 266, 493
  - check 46, 56, 117, 465
  - commands 256, 479
  - continue 266, 493
  - for 258–262, 493
  - if-then-else 264–266, 493–494
  - repeat 262–264, 493
- statements, control flow 492
- `.status`
  - solver status 473
  - suffix 295
- steel production model 2–7, 10–19
- step command 268, 495
- stepping through commands 268–270, 495
- stochastic programming 69–71
- Stone, Richard E. 432
- stress analysis model 388
- string
  - conversion from number to 271, 459
  - expression 194, 197, 273, 459
  - functions, table of 459
  - literal 74, 144, 453–454, 473
- string
  - concatenation operator, `&` 270, 459
  - quotes, `%q`, `%Q printf` 483
- sub function 273, 460
- subject to *see* constraint declaration
- subscript 8, 16, 110, 455
- subset test *see* in, within
- subsets of pairs 93–96
- substitution of variables 399–400, 414
- `substout` option 399, 414, 467–468
- `substr` function 271, 459
- such-that condition *see* logical condition
- suffix declarations 471
- suffix
  - `.astatus` 294–295
  - attribute, IN, INOUT, OUT, LOCAL 304, 472
  - `.direction` 296
  - `.iis` 299
  - `.priority` 296
  - `.relax` 486
  - `.result` 287
  - `.slack` 81, 241–242, 244, 472
  - `.sstatus` 290, 292, 295, 473
  - `.status` 295
- suffix
  - attribute 466, 469–470, 489
  - command 296, 299, 302
  - initialization 466, 469–470, 489
- suffixes
  - for complementarity constraint 428, 431, 471
  - solver status 286, 288
  - solver-defined 298–304
  - table of 472
  - user-defined 296–297, 302
- Suhl, Uwe H. 449
- sum of squares 388, 412
- sum operator 74, 113, 456
- symbolic parameters 123–124, 323, 328
- symbolic attribute 123, 465, 476, 497
- `symdiff` operator 76, 98, 456–457
- symmetric difference *see* `symdiff`
- synonyms, table of 454
- syntax error *see* error messages
- `.tab` ASCII filename extension 201, 477
- table
  - declaration 174, 178, 180, 186, 477
  - entry, empty 173, 187
  - handler 169, 197–202, 477
  - handler, built-in 174, 201
  - handler, ODBC 169, 174, 194, 198
  - of arithmetic functions 113, 458
  - of arithmetic, logical, set operators 456
  - of arithmetic operators 112

- of auxiliary files 488
- of built-in timing parameters 496
- of commands 480
- of environment variables 484
- of functions of sets 464
- of generic names 493
- of random number functions 459
- of regular expression functions 459
- of reserved words 454
- of rounding functions 212
- of string functions 459
- of suffixes 472
- of synonyms 454
- parameter data 154
- reading scalar from 182
- relational 170–171, 173, 178
- set data 474
- table
  - of display formatting options 227
  - of display numeric options 232
  - of printf conversion characters 483
  - transposition (`tr`) 32, 155, 157, 228–229, 475–476
- tables, indexed collections of 193
- tabular data
  - multidimensional 156–160
  - two-dimensional 154–156
- `tan` function 113, 458
- `tanh` function 113, 458
- Tayyabkhan, Michael T. 20
- template
  - parameter data 147–148, 150, 475
  - set data 147, 150, 474
- temporary file 485
- `time` function 458
- `times` option 250
- timing parameters, table of built-in 496
- `TMPDIR` option 285, 485
- `to` attribute 334, 336, 338, 341, 353, 361, 363, 467
- `to_come`
  - constraint declaration 356, 362, 468
  - objective function 362, 470
- tolerance, feasibility 281
- (`tr`), table transposition 32, 155, 157, 228–229, 475–476
- traffic flow *see* maximum flow model
- transitive closure of graph 462
- transportation model 44–49, 52, 330–333
  - integer solutions in 51, 439–448
  - multicommodity 56–59, 450
  - nonlinear 403–410
  - piecewise-linear 366–369
  - production and 63–67
  - transposition *see* table transposition
- transshipment model 319–327, 334–337, 347
- `trunc` function 212, 458
- tuple operators 98–100
- tuples 73, 91, 96, 146
  - sets of 96–98
- two-dimensional
  - list data 146–148
  - parameter 110
  - set 110, 455
  - set data 475
  - tabular data 154–156
- `.ub` upper bound 35, 240–242, 279, 470, 486
- `.ub0`, `.ub1`, `.ub2` upper bound 279
- `unfix` command 215, 489
- unformatted data 163
- Unicode 453, 459
- `Uniform, Uniform01` function 459
- union operator 76, 98, 103, 456–457
- unload command 498
- unordered sets 73–74
- unspecified data value *see* default symbol
- `until` condition 262, 267, 494
- update data command 209, 491
- user-defined suffixes 296–297, 302
- 
- `-v` command-line argument 500
- valid column name 171–172, 177, 185
- validation, data 12, 46, 93, 118
- Van Wyk, Chris J. xxi
- Vanderbei, Robert J. 20
- `var_bounds` option 279, 486
- variable
  - data 162, 398, 477
  - declaration 5, 8, 466–467
- variables xv, 2, 129–132
  - bounds on 10, 14, 17, 32, 35, 130, 139, 241–242, 276, 279, 466, 470–471, 486
  - defined 466–468
  - dual 162, 243, 468, 472, 485
  - environment 205, 481
  - fixed 131, 241, 402
  - initial values of 131, 162, 398, 405, 408–409, 466, 471, 477, 486
  - initial values of dual 185, 468, 472, 477, 486
  - nonlinear 397–400
  - nonnegative 3, 10, 45, 130, 377
  - primal 243, 398, 485
  - scaling of 410
  - substitution of 399–400, 414
  - zero-one 439–448

- vector *see* one-dimensional set, parameter
- verbose attribute 200
- version option 500
- viewing data *see* display command
- Vignali, Juliana xxi
- Vukhac, Thong xxi
  
- Waren, Allan 414
- while condition 262, 267, 494
- white space 144, 453, 473
- wild card \* 205, 481
- Wilkins, Bert 411
- Williams, H. Paul 67
- Williard, Mark 411
- within operator 78, 93, 97–98, 114, 456–457, 462
- Wolsey, Laurence A. 450
- words
  - predefined 74, 112, 454
  - reserved 74, 112
- Wright
  - Margaret H. xxi, 410
  - Stephen J. 410
- write
  - command 487
  - table command 178, 186, 477
- www.ampl.com xviii, xx, 19, 453
  
- .xls Excel filename extension 199–200
- xref command 213, 247, 492
  
- zero
  - piecewise-linear function at 367, 376
  - suppression 231–232
- zero-one programming *see* integer programming
- zero-one variables 439–448
- zero-or-minimum restrictions 444