

Index

- adaptive Runge–Kutta method, 271
- arithmetic operators, in Python, 6
- arrays, in Python, 20
- augmented assignment operators, in Python, 7
- augmented coefficient matrix, 32

- banded matrix, 59
- bisection method, for equation root, 148
- boundary value problems, 293
 - shooting method, 294
 - finite difference method, 307
- Brent's method, 183
- Bulirsch–Stoer method, 280
- bulStoer.py, 284

- choleski.py, 50
- Choleski's decomposition, 48
- cmath module, 19
- comparison operators, in Python, 7
- conditionals, in Python, 8
- conjGrad.py, 91
- conjugate gradient method, 89
- continuation character, in Python, 6
- cubic spline, 120

- deflation of polynomials, 175
- diagonal dominance, 70
- docstring, in Python, 29
- Doolittle's decomposition, 45
- Dormand–Prince coefficients, 273
- downhill simplex method, 392

- eigenvals3.py, 364
- eigenvalue problems, 322
 - eigenvalues of tridiagonal matrices, 359
 - Householder reduction, 351
 - inverse power method, 336
 - Jacobi method, 329
 - power method, 338
- elementary operations, linear algebra, 34
- equivalent linear equation, 34
- error control, in Python, 15
- euler.py, 248

- Euler's method, 247
- evalPoly.py, 175
- evaluation of polynomials, 173

- false position method, 152
- finite difference approximations, 183
- finite elements, 232
- functions, in Python, 16

- gaussElimin.py, 37
- Gauss elimination method, 41
 - with scaled row pivoting, 71
- Gaussian integration, 216
 - abscissas/weights for Gaussian quadratures, 221
 - orthogonal polynomials, 217
- Gauss–Jordan elimination, 36
- Gauss–Legendre quadrature over quadrilateral element, 233
- gaussNodes.py, 224
- gaussPivot.py, 72
- gaussQuad.py, 225
- gaussQuad2.py, 235
- gaussSeidel.py, 89
- Gauss–Seidel method, 87
- gerschgorin.py, 363
- Gerschgorin's theorem, 361
- golden section search, 377
- goldSearch.py, 378

- householder.py, 356
- Householder reduction to tridiagonal form, 351
 - accumulated transformation matrix, 355
 - Householder matrix, 352
 - Householder reduction of symmetric matrix, 352–359

- Idle (Python code editor), 3
- ill-conditioning, matrices, 33
- incremental search method, roots of equations, 146
- initial value problems, 246
- integration order, 224

- interpolation/curve fitting, 104
- interval halving method. *See* bisection method
- `inversePower.py`, 339
- inverse power method, 336
- `inversePower3.py`, 366

- `jacobi.py`, 326–327
- Jacobian matrix, 234
- Jacobi method, 324
- Jenkins–Traub algorithm, 182

- knots of spline, 120

- Lagrange’s method, of interpolation, 104
- Laguerre’s method, for roots of polynomials, 176
- `lamRange.py`, 363
- least-squares fit, 129
- linear algebra module, in Python, 24
- linear algebraic equations, 31
- linear regression, 130
- linear systems, 30
- `linInterp.py`, 295
- lists, in Python, 5
- loops, in Python, 9
- LR algorithm, 373
- `LUdecomp.py`, 47
- `LUdecomp3.py`, 61
- `LUdecomp5.py`, 66
- LU decomposition methods, 44
 - Choleski’s decomposition, 48
 - Doolittle’s decomposition, 45
- `LUpivot.py`, 73

- mathematical functions, in Python, 11
- `math` module, 18
- MATLAB, 2
- `matplotlib.pyplot` module, 25
- matrix algebra, 410
- matrix inversion, 84
- midpoint method, 280
- `midpoint.py`, 282
- minimization along line, 376
 - bracketing, 377
 - golden section search, 377
- modules, in Python, 18
- multiple integrals, 232
 - Gauss–Legendre quadrature over quadrilateral element, 233
 - Gauss–Legendre quadrature over triangular element, 239
- multistep methods, for initial value problems, 292

- namespace, in Python, 28
- natural cubic spline, 120
- Nelder–Mead method, 392
- `neville.py`, 110
- Neville’s method, 109

- Newton–Cotes formulas, 200
 - Simpson’s rules, 204
 - trapezoidal rule, 200
- `newtonPoly.py`, 108
- `newtonRaphson.py`, 158
- `newtonRaphson2.py`, 162
- Newton–Raphson method, 156, 161
- norm of matrix, 33
- `numpy` module, 20
- numerical instability, 260
- numerical integration, 199

- operators, in Python
 - arithmetic, 6
 - comparison, 7
 - optimization, 374
 - orthogonal polynomials, 217

- pivoting, 69
- `plotPoly.py`, 133
- `polyFit.py`, 132
- polynomial fit, 131
- polynomial interpolation, 104
 - Lagrange’s method, 104
 - Neville’s method, 109
 - Newton’s method, 106
- polynomials, zeroes of, 173
 - deflation of polynomials, 175
 - evaluation of polynomials, 173
 - Laguerre’s method, 176
- `polyRoots.py`, 177
- `powell.py`, 385
- Powell’s method, 382
- printing, in Python, 12
- `printSoln.py`, 249

- QR algorithm, 373
- quadrature. *See* numerical integration

- rational function interpolation, 115
- reading input, in Python, 11
- relaxation factor, 88
- Richardson extrapolation, 188, 281
- Ridder’s method, 152
- `ridder.py`, 153
- `romberg.py`, 209
- Romberg integration, 207
- `rootsearch.py`, 147
- roots of equations, 145
 - bisection, 148
 - incremental search, 146
 - Newton–Raphson method, 156, 161
 - Ridder’s method, 152
- Runge–Kutta methods, 252
 - fifth-order adaptive, 271
 - fourth-order, 254
 - second-order, 253
- `run_kut4.py`, 255
- `run_kut5.py`, 274

- scaled row pivoting, 71
- shape functions, 234
- shooting method, 294
 - higher-order equations, 299
 - second-order equation, 294
- Shur's factorization, 373
- similarity transformation, 325
- Simpson's 1/3 rule, 204
- Simpson's 3/8 rule, 205
- slicing operator, Python, 3
- `sortJacobi.py`, 331
- stability of Euler's method, 268
- stiffness, in initial value problems, 267–268
- `stdForm.py`, 332
- strings, in Python, 4
- Sturm sequence, 359
- `sturmSeq.py`, 359
- `swap.py`, 72
- symmetric/banded coefficient matrices, 59
 - symmetric coefficient matrix, 62
 - symmetric/pentadiagonal matrix, 63
 - tridiagonal matrix, 60
- synthetic division, 175
- `trapezoid.py`, 203
- trapezoidal rule, 200
- `triangleQuad.py`, 241
- tridiagonal coefficient matrix, 60
- tuples, in Python, 5
- two-point boundary value problems, 293
 - finite difference method, 307
 - shooting method, 294
- type conversion, in Python, 10
- weighted linear regression, 134
- writing/running programs, in Python, 29