

Index



A

- Absorbing states, 344
- Acceleration, 70, 208–212, 427–428
- Accelerometers, 49–50, 390–391
- Accuracy versus precision, 109–111
- Acoustic localization, 504–516
 - cones of confusion, 220, 510–511
 - continuous sound source and, 515
 - cues, 505
 - head related transfer functions (HRTF), 505–506
 - interaural level differences (ILD), 505
 - interaural time differences (ITD), 505, 507–510
 - Nic_3 robot design, 508–516
 - spatial direction of sound, 510–512
- Adaptive (learning by doing) programs, 346, 363–368
- Aelianus, Claudius, 256
- Agent embodiment and disembodiment, 259–260
- Agents (function and program), 114
- Algorithms, 116–117, 120–123, 127–129, 140–146, 148–172, 296–302, 412–424, 430–433, 464–467
 - Babylonian, 128–129
 - Boolean logic and, 120
 - CORDIC (coordinate rotation digital computer), 156–162
 - Dijkstra, 140–142, 166, 169–172
 - divide and conquer, 143–146, 148–156, 164
 - exponential growth and, 155–156
 - “15” puzzle using, 169–172
 - foraging problem, 296–302
 - formal verification of, 140
 - fuel consumption problem, 162–163
 - Heron’s, square root calculation using, 128–129, 286–288
 - heuristic (informed guess), 121–122, 172
 - Kalman filter (KF), 412–424, 427–433, 521–526
 - LEGO challenges using, 162–172
 - maze solutions from, 120–123, 127, 141–142
 - naïve sort, 163–164, 165
 - Pledge, 121–123, 127, 162
 - predict-correct, 430–433
 - problem solver exercises using, 148–162
 - programmer’s use of, 116–117, 120–123, 127–129
 - proof by infinite descent, 167–168
 - quicksort, 164–165
 - random walk algorithms, 297–302
 - rational and irrational numbers, 167–168
 - recursive functions, 148–155
 - recursive predictor/corrector, 418–420
 - sine/cosine calculation efficiency, 156–162
 - spigot, 464–467
 - study of for problem-solving, 116–117
 - sweep scan algorithm, 297
 - Toepler’s, 129
 - Tower of Hanoi, 148–155
 - Turing machine and, 116, 119
 - turning by a constant angle algorithm, 300–302
 - Z-series computers and, 116–117

- Al-Jazari's musical "robot band," 113–114
 - Al-Khwāizmī robot navigation behavior, 310–315
 - Analog audio, 492–493
 - Angle (phase), 79
 - Angle sensor, 65–69
 - Animal mimicry, 36–38, 50–51, 244, 249–251, 481, 488–491
 - animal as a machine, 36
 - animal mimicry (cognition), 244, 249–251, 481
 - cricket sound creation, 488–491
 - Jansen mechanism applications, 37, 51
 - robot designs inspired by nature, 50–51
 - sensors for, 8
 - synchronization of locomotion, 36–37
 - walking mechanisms, 37
 - Animation of Braitenberg vehicles, 260
 - Antikythera mechanism, 117–118
 - Apollo guidance computer (AGC), 412–415
 - Archimedes, 35–36, 131, 464
 - Archimedes' palimpsest, 35
 - Archimedes' planetarium, 118
 - Aristotle, 384
 - Arithmetic and Logic Unit (ALU) operations, 157–160
 - Arkin, Ronald, 249
 - Artificial intelligence (AI), 119, 245–248
 - challenges of, 248
 - robotics and, 119, 245–248
 - Turing test, 245
 - Automata, 76–79, 117–120, 244–245
 - Antikythera mechanism, 117–118
 - Archimedes' planetarium, 118
 - chess player of Maelzel, 76–78, 244
 - computers and robots compared to, 117–120
 - Mechanical Turk, 77–78
 - pantograph, 78–79
 - Pascaline, 118
 - robotics and, 244–245
 - Avoid-to-the-left behavior, 313
 - Azimuth (bearing), 70, 79
- B**
- Babylonian algorithm, 128–129
 - Bach, Johann Sebastian, 183
 - Bearing (azimuth), 70
 - Bee "waggle dance," 5, 123, 182–183, 201–202
 - code, 182–183, 201–202
 - maze algorithm strategies and, 123
 - robot design based on, 182–183, 201–202
 - Behavior of robots, 248–251, 263, 292–341, 357–372
 - animal mimicry and, 249–251
 - avoid-to-the-left, 313
 - dead reckoning navigation, 310–315
 - decision-making functions, 357–363
 - effects of actions, 249
 - emergent, 292–294
 - environmental interaction and, 248–249, 292–296
 - event-driven, 323–327
 - foraging problem, 296–302
 - goal-oriented, 292–293, 302
 - intelligence and, 251, 292–294, 357–372
 - intelligent robots, 292–341
 - learning programs, 346, 363–372
 - morphology and, 292–293
 - mysterious, 303–304
 - priorities of multiple functions, 357–363
 - repetitive "buggy," 304
 - semaphore algorithm signals and messages, 315–323
 - sensors and, 295–296, 304, 311–315, 317–321, 333–334
 - suspension of tasks, 357–363
 - transition probability, 366, 368–372
 - turning, 302–304
 - virtual bee robot, 296–302
 - wait functions, 304, 323, 361
 - Bell, Alexander Graham, 492
 - Bell, Eric Temple, 2
 - Bellman, Richard E., 366
 - Benedettelli, Daniele, 471
 - Benzoni, Girolamo, 3
 - Berliner, Emile, 492
 - Berners-Lee, Timothy John, 176
 - Bernoulli, Jacob, 306
 - Bernoulli robot behavior, 302–306
 - Binary number implementation, 126–127

Binary phase-shift keying (BPSK), 181–182
 Bionic mimicry of nature, *see* Animal mimicry
 Bitmap conversion to curves, 476
 Blind Juggler robot, 216–218
 Bohr, Niels, 2–3
 "Bombe" cypher, 184
 Bonaparte, Emperor Napoleon I, 142–143
 Boolean logic, algorithms and, 120
 Boulette's Institute of Technology (BIT), 14
 Boy Scout and Follower program, 123
 Brainstorming, 387
 Braitenberg, Valentino, 260, 262, 268
 Braitenberg vehicles, 259–265, 268–269, 285
 animation of, 260
 data-logging with, 285
 design of, 259–265
 dual sensor reactions, 268–269, 270–271
 Light-dependent resistor (LDR), 263–264
 monosynaptic reflex arc, 261–262
 NXT programming, 260–261
 open-loop trajectory of, 264–265
 sensor values, 259–265
 steady state of, 263
 Vehicle 1 design and behavior, 259–265
 Vehicle 2 behavior, 268
 Vehicle 3 behavior, 268, 269
 Brake mode, 281
 Bratzel, Barbara, 11
 Bricoleurs, 57, 59–60
 Brightness, adjustment of, 472–473
 Brooks, Rodney A., 248–249, 355, 506
 Brown, J. P., 150
 Bucket brigade problems, 303–304, 311–315
 Bugs, 254–291, 296–302
 Braitenberg vehicle designs, 259–265, 268–269
 Cauchy robot design, 269–271
 convergence of a sequence, 268–271, 286–288
 Fermat problem, 272–278
 foraging problem, 296–302
 geotaxis, 259, 271
 Helmholtz vehicle design, 266–267
 honeycomb pattern characteristics, 256–259
 insect behavior, 254–255, 269, 271–272

LEGO challenges, 282–289
 magnetotaxis, 259, 271
 perception and reactions, 259
 phototaxis, 269, 271–272
 polarotaxis, 288
 problem solver exercises, 272–282
 pulse-width modulation (PWM), 278–281
 random walk algorithms, 297–302
 robot intelligence and, 296–302
 robotics and, 254–291
 sweep scan algorithm, 297
 synthetic psychology, 259–260, 268–269
 thinking capabilities of, 254–255, 296
 transistors, 281–282
 turning by a constant angle algorithm, 300–302

C

Calibration of paper encoder, 65–69
 Capacitors, 102–103
 Cardan suspension (gimbal), 89–90
 Cartesian coordinates, 54–55
 Catching a free-falling ball, model for, 239–240
 Cauchy robot design, 269–271
 Cauchy, Augustin Louis, 269–270
 Cavalieri, Bonaventura Francesco, 272–273
 Cesar's cipher, 199–201
 Champollion, Jean-François, 183
 Chebyshev, Pafnuty L., 62
 Checksum, 181, 194, 437–438
 Chess player of Maelzel, 76–78
 Chongzhi, Zu, 131
 Choosing between tasks, 362–363
 Church's thesis, 116
 Circuit board etching, 96–99
 Clarke, Arthur C., 129
 Clean-up diagram, LabVIEW, 147
 Clock models, 232–238
 data-logging subroutines, 234–235
 pendulum, 232–237
 rolling ball, 237–238
 Closed-loop control, 215–216, 218–219
 Coast mode, 281

- Code breakers, 174–205
 - bee “waggle” dance, 182–183, 201–202
 - Cesar’s cipher, 199–201
 - checksum, 181, 194
 - cryptograms, 183–184, 199–204
 - decryption, 183–184, 199–201
 - encoding, 181–183
 - endianness, 204
 - LEGO challenges, 195–204
 - Mars Rover telerobotic project, 174–182, 184–185
 - Morse code, 202–203
 - problem solver exercises, 185–194
 - problem-solving methods by, 175, 183–184
 - rules for, 182–183
 - telerobotics, 174–182, 184–185, 195–199
- Cohen, Danny, 204
- Collision avoidance, 76–77
- Colossus Mark 1 computer, 117
- Compass design, 57–58, 83–85, 87–90, 91–96, 103, 434–435 . *See also* Dead reckoning; Direction master
 - Earth’s magnetic field measurement, 83–85
 - electronic sensors, 87–88
 - electronic, 83–85, 87–90
 - gimbal (Cardan suspension), 89–90
 - Hall-effect sensors, 83, 87–88
 - magnetic, 57–58
 - magneto-resistive permalloy sensors, 84–85
 - needle, 83–84
 - NXT configuration as, 103
 - RCX control program for, 88–89
 - Serendipity boat navigation, 434–435
 - tilt error, 89–90
- Computer science, LEGO RIS for education of, 12–13
- Computers, 116–120, 131
 - automata and robots compared to, 117–120
 - Colossus Mark 1, 117
 - Electronic Numerical Integrator and (ENIAC), 117
 - read/write capabilities, 119
 - sine and cosine calculations using, 131
 - Turing machine, 116, 118–119
 - Z-series, 116–117
- Concorde TSP solver, 296
- Conditional probability, 376–377
- Cones of confusion, 220, 510–511
- Conquerors, 140–173
 - CORDIC (coordinate rotation digital computer), 156–162
 - Dijkstra algorithm, 140–142
 - divide and conquer algorithm, 143–146, 148–156
 - emotions expressed by GASTON, 143–146
 - exponential growth and, 155–156
 - Napoleon’s theorem, 142–143
 - problem-solving by, 140–147
 - sine/cosine calculation efficiency, 156–162
 - spaghetti code, 146–147
 - Tower of Hanoi algorithm, 148–155
- “Consul,” the educated monkey, 73–74
- Continuous-sound locator design, 219–222
- Contrast, adjustment of, 472–473
- Control, 215–219, 252
 - closed-loop, 215–216, 218–219
 - comparison of methods, 215–216
 - open-loop, 215–218
 - robotic development, methods for, 252
- Convergence sequence, 268–271, 286–288
- Conversion functions, 70–71
- Coordinates, 54–55, 79–81
- CORDIC (coordinate rotation digital computer) algorithm, 156–162
- Counting pulses, 46–48
- Course, 70
- Create Sub.vi tools, LabVIEW, 147
- Cremonini, Cesare, 385
- Crenshaw, Jack W., 129, 413
- Crickets, sound creation, 488–491
- Cross-correlation function, 495–498
- Cruise task, 311–313, 324, 358–360
- Cryptograms, 181–184, 199–204
 - “Bombe” cypher, 184
 - Bach motif, 183
 - bee “waggle” dance code, 182–183, 201–202
 - binary phase-shift keying (BPSK), 181–182
 - Cesar’s cipher, 199–200, 201
 - decryption, 183–184, 199–201
 - DNA code, 183
 - encoding, 181–183
 - LEGO challenges, 195–204
 - Morse code, 202–203

- Rosetta stone, 183
 - safe or puzzle box design, 199
 - WWII "Ultra Secret," 183–184
- Cues, sound, 505
- Cyclops robot design, 288–289
- D**
- Da Vinci robot, 346, 368–372, 378. *See also*
 - Dynamic learning programs
- da Vinci, L eonardo, 6, 36, 51–53
- Data communication, 179–181
- Data-logging subroutines, LabVIEW, 234–235, 285, 329
- Data Viewer for upload and comparison, LabVIEW, 329–330
- David, Brian, 22, 433
- de Mestral, George, 36
- de Solla Price, Derek J., 118
- Dead reckoning, 44, 57, 310–315
- Deadlock situation, 317
- Decibel, dB, 487–488
- Decision graphs, 368–369
- Decision tables, 318–321
- Decryption, 183–184, 199–201
- Deep Blue, 245–246
- Deformation theorem for simple closed curves, 121
- Degrees of freedom (DOF), 220–222
- Delay function as memory, 302–303
- Descartes, Ren e, 6, 36, 54–55, 244
- Dewey, John, 11, 366
- Dewey robot, 346, 363–368. *See also* Adaptive learning programs
- dGPS unit, 439
- Differential equations for rotation, 39–41
- Differential gear train, 34
- Digital numbers, programming and, 136–138
- Digital signal processing (DSP), 493–495
- Dijkstra, Edsger Wybe, 140–141, 316
- Dijkstra algorithm, 140–142, 167, 169–172
- Dinsmore 1655 electronic compass, 87, 91–96
- Direction master robot, 32–34, 39–41, 57–58
 - kinematics of design, 39–41
 - magnetic compass for, 57–58
 - shaft-encoder design, 33–34
 - sensors used for, 39–41
 - transmission design, 33
- Discoverers, 31–56
 - Archimedes' palimpsest, 35–36
 - bionic mimicry and, 36–38
 - direction master robot design, 32–34, 39–41
 - importance of, 31–32
 - Internet use and, 32–35
 - LEGO challenges, 45–55
 - lightning vehicle robot design, 38, 42–43
 - problem solvers exercises, 39–44
 - robot motion and, 36–38, 39–55
 - south-pointing chariot, 34
- Disembodiment, 259–260
- Dispatching function, 326
- Distance (magnitude), 79
- Distance, robot driving of, 45–46
- Divide and conquer algorithm, 143–146, 148–156, 164
- DNA code, 183
- Don't think-react! design, 274–275
- Downloading HTML files, 196–199
- Drawing, 99–100, 445–480
 - adjustment of picture values, 472–477
 - Euler's identity and, 463
 - histograms for values of, 472–475
 - LEGO parts used for, 99–100
 - Pi-writer, 463–470
 - plotter control, 445–448, 477
 - polar (RR) plotter, 454, 471–479
 - Pythagoras' theorem and, 450–451
 - robotic calligraphy, 467–479
 - robotic hand for, 471–479
 - rule of thumb, 449–451
 - Spigot algorithm for, 464–467
 - trigonometry for, 448–454
 - xy-plotter, 454–463
- Drunken sailor problem, 335–337, 344–346, 363–365
- Dual-drive kinematics and path integration, 328–333
- Dual-drive robot program, 394–397
- Duct tape, inventive use of, 58
- Dudeney, Henry Ernest, 4
- Duty cycle, 278
- Dynamics of physical systems, 207

E

Earth's magnetic field measurement, 83–85
 Edge detection, 474–475
 Edison, Thomas Alva, 58–59, 492
 Education, 6–16, 383–389. *See also* Learning programs
 brainstorming, 387
 computer science, 12–13
 Hole-in-the-Wall project, 9
 instruction versus dictation, 383–385
 LEGO Robotics Invention System (RIS), 10–16
 Montessori methods, 9, 11
 open-ended projects and, 383–389
 physics by design method, 11
 project-based learning, 386–388
 rational versus empirical methods, 6–7
 real-world applications and, 7–8
 resistance to change and, 383–386
 robotics clubs and organizations, 13–16
 science, technology, engineering, and mathematics (STEM), 7
 trigonometry, 7–8
 “Turtle graphics,” 12
 Egg of Columbus, 2–5. *See also* Problem solving
 Einstein, Albert, 390
 Electronic devices, 83–85, 87–103, 410–411
 capacitors, 102–103
 compass design, 83–85, 87–90, 103
 connecting Dinsmore 1655 sensor to NXT, 91–96
 Ohm's law, 88, 100–101
 operational amplifiers (op-amps), 410–411
 printed circuit board etching, 96–99
 resistors, 100–101
 sensors, 87–88
 Electronic Numerical Integrator and Computer (ENIAC), 117
 Electronic sensors, compass design using, 87–88
 Embodiment, 259
 Emotions expressed by GASTON, 143–146
 Empirical density function, 228–230
 Empirical methods, 6–7, 206–207
 Encoder devices, 33–34, 39–41, 46–48, 65–69
 angle sensors from, 65–69
 calibration of, 65–69

 counting pulses, 46–48
 differential equations for, 39–41
 gradual gray-toned, 65–69
 hysteresis technique for, 47–48
 light sensors and, 65–69
 paper (transmission) shaft, 33–34, 65–69
 rotation sensors and, 39–41
 Encoding, 181–183
 Endianness, 204
 Engelberger, Joseph F., 119
 Enumerator list, LabVIEW, 358–359
 Environmental interaction, robot behavior and, 248–249, 292–294
 Epistemology, 7
 Eppler, John W., 162–163
 Equilibrium (Nash), 206
 Error, 89–90, 103–105, 407–409, 423–424
 estimation variance, 423–424
 measurement, 407–409
 nonorthogonality, 103–104
 nonparallelism, 103
 propagation, 407–409
 tilt, 89–90
 Estimation, 411–424. *See also* Prediction and estimation
 Etching printed circuit boards, 96–99
 Euler's identity, 463
 Event-driven (flags) behavior, 323–327, 352
 Event handling, 326
 Event status variable, 324, 353
 Exhaustion, method of, 273
 Exponential growth, 155–156

F

Fermat problem, 272–278
 calculation methods for, 272–274
 LabVIEW program for, 276–277
 omnidirectional holomorphic platform (OHP), 275
 robot design for, 273–276
 Steiner points, 273–274
 time difference of arrival (TDOA), 276
 Fermat's Last Theorem, 2
 Fermi, Enrico, 5
 Ferrari, Mario, 72, 467–468
 Fibonacci, Leonardo, 116, 164

"15" puzzle challenge, 169–172
 Filtering property, 415–416
 Finite state machines (FSM), 248, 346–363, 377
 disjunctive, 356
 event flags, 352
 event variables, 353
 functionality graphs, 347–350
 LabVIEW programs for, 351–354, 360–363
 Moore robot demonstration of, 347–354
 Moore-state control variable, 351–354
 robotics use of, 248, 346, 348
 Rodney robot, 346, 355–363
 state transitions, 353–354
 subsumption architecture, 346, 355–363
 transition tables, 351
 Firmware, 16, 18–19
 First-come, first-served (FCFS) queue, 316
 Fix, 70
 Flags, event status and, 323–327
 Floating-point numbers, 136–137
 Flory, Paul, 36
 Flow charts, 319
 Flying machines, robot re-creation of, 51–54
 For Inspiration and Recognition of Science and
 Technology (FIRST), 14
 Foraging problem, 296–302, 334
 Fourier transform, 495, 500–501
 Frey, Karl, 386–387
 Fuel consumption problem, 162–163
 Functionality graphs, 347–350

G

Galilei, Galileo, 208, 232–233, 383
 Gasperi, Michael, 38, 60, 120, 406, 489
 GASTON, 14–19, 59–60, 143–146
 commission of, 14
 emotions expressed by, 143–146
 inventors use of teamwork for, 59–60
 ROBOLAB software for, 16–19
 rules for development of, 14–15
 sensors for, 15
 sensor multiplexer "mood-meter" design,
 60
 Ultimate ROBOLAB (UR) software for,
 18–19

Gauss, Carl Friederich, 83–84, 109–110, 163,
 212–214, 223–228, 407
 Gauss' linear fit method, 223–232
 Gaussian (normal) distribution, 212–214, 227–228
 Geomagnetism, 83–84
 Geotaxis, 259, 271
 Gimbal (Cardan suspension), 89–90
 Global positioning system (GPS), 397–310,
 428–430, 434–441
 indoor (iGPS), 397–310, 428–430
 National Marine Electronics Association
 (NMEA), 439
 navigation problem, 397–310
 Serendipity boat navigation, 434–441
 Gradient pad generation, 71–73
 Gray, Elisha, 492
 Gyroscope design, 104

H

HAL9000, 246–247
 Hales, Thomas C., 256
 Hall-effect sensors, 83, 87–88
 Hamming code, 194
 Harrison, John, 107–108
 Head related transfer functions (HRTF), 505–506
 Helmholtz vehicle design, 266–267
 Heron of Alexandria, 128
 Heron's algorithm, 128–129, 286–288
 Hestenes, David, 207
 Heuristics (informed guess), 106–107, 121–122,
 127, 172
 High-Altitude Long Endurance (H.A.L.E.), 26–28
 Histograms, 201, 231, 472–475
 Hoare, Tony, 164–165
 Hole-in-the-Wall Education project, 9
 Honeycomb conjecture (pattern), 256–259
 HTML files, downloading, 196–199
 Hughes, David E., 492
 Hurbain, Philippe (Philo), 22, 92, 106, 145,
 207–211, 275, 279
 Huygens, Christiaan, 233
 Hypertext Transfer Protocol (HTTP), 175–179
 Hysteresis, method of, 47–48

- I**
- Indoor global positioning system (iGPS),
 - 397–310, 428–430
 - dual-drive robot program, 394–397
 - infrared/ultrasonic indicators, 398–410
 - Mars simulation use if, 428–430
 - Pythagoras' theorem, 394
 - Thales' theorem, 394
 - thunderbolt signal concept, 398–404
 - trilateration, 398–403
 - Inertial navigation, 107–109
 - Infinite descent, proof by, 167–168
 - Infrared/ultrasonic indicators, 398–410
 - Inhibitors, 356
 - Insect behavior, *see* Bugs
 - Intelligence, 244–245, 342–381. *See also* Artificial intelligence (AI)
 - adaptive learning program, 346, 369–368
 - choosing between tasks, 362–372
 - cruise and course correction, 358–360
 - Da Vinci robot, 346, 368–372, 378
 - Dewey robot, 346, 363–368
 - Drunken sailor problem, 344–346
 - dynamic learning programs, 346, 368–372
 - finite state machine (FSM), 346–363, 377
 - GASTON "wow" effect, 342–343
 - LabVIEW programs for, 358–362, 367–369, 373–375
 - LEGO challenges, 375–380
 - line detection/following, 356–358
 - Markov chain, 344–345, 347, 373–375
 - Moore robot, 346, 347–354
 - object avoidance, 360–362
 - probability, 344–346, 366, 368–372, 376–378
 - problem-solver exercises, 373–375
 - robot functions, 342–381
 - robot state, 343–346
 - robotics and, 244–245, 292–294
 - Rodney robot, 346, 355–363
 - sampling problems, 375–376, 379–380
 - sensors and, 357–363, 366–368
 - subsumption architecture, 346, 355–363, 377
 - transition probability, 366, 368–372
 - Intelligent robots, 251, 292–341
 - Al-Khwāzmi navigation behavior, 310–315
 - behavior and, 251, 292–306
 - Bernoulli behavior, 302–306
 - bucket brigade problems, 303–304, 311–315
 - computational perspective of, 294–296
 - dead reckoning navigation, 310–315
 - decision tables and, 318–321
 - delay function as memory, 302–303
 - drunken sailor problem, 335–337
 - dual-drive kinematics and path integration, 328–333
 - emergent behavior of, 292–294, 302–303
 - environment interaction and, 292–296
 - event-driven behavior, 323–327
 - flags, event status and, 323–327
 - foraging problem, 296–302, 334
 - LabVIEW programs for, 302, 304–305, 311–313, 317–318, 320–323, 324–325, 329–332
 - LEGO challenges, 333–340
 - Nelson semaphore behavior, 315–323
 - NXT-G 2.0 program for, 303–305
 - Piccard event driven behavior, 323–327
 - problem-solving exercises, 328–333
 - random numbers and, 307–310
 - random walk algorithms, 297–302
 - reaction delays, 335
 - rotation sensor values and, 311–315
 - semaphore algorithm signals and messages, 315–323
 - Seven bridges of Königsberg problem, 302, 337–339
 - Star Wars problem, 339–340
 - sweep scan algorithm, 297
 - touch sensors, 295–296, 304, 311–312, 317–321, 333–334
 - travelling salesman problem (TSP), 296
 - turning by a constant angle algorithm, 300–302
 - virtual bee, 296–302
 - wait-for-pressed functions and, 304
 - walking the grid problem, 335
 - Intensity graphs, 71–72, 472–474
 - Interaural level differences (ILD), 505
 - Interaural time delay (ITD), 220, 512
 - Interaural time differences (ITD), 505, 507–510
 - Internet use, 32–35, 175–181, 196–199
 - binary phase-shift keying (BPSK), 181–182

- data communication, 179–181
- discovery and, 32–35
- Hypertext Transfer Protocol (HTTP), 175–179
- POST command, 178–179
- robot control over the web, 196–199
- TCP listener, 177–178
- telerobotics, 175–181, 196–199
- Transmission Control Protocol/Internet Protocol (TCP/IP), 175–178
- two's complement numbers, 179–180
- Interrupt service routine, 326
- Inventors, 57–82
 - bricoleurs, 57, 59–60
 - direction master robot design, 39–41, 65–69
 - gradual grey paper shaft encoder calibration, 65–69
 - LEGO challenges, 70–81
 - linkages, 61–63, 70–81
 - magnetic compass design, 57–58
 - navigation, 69–70
 - never-give-up philosophy, 59, 63
 - problem solver exercises, 65–70
 - problem-solving methods by, 57–64
 - sensors, 39–43, 65–69
 - tinkering by, 57–63
 - trial-and-error approach, 57, 59, 63
- Inverse-square law, 285–286
- Inversor (Peaucellier) linkage, 62–63, 74–75
- Inverted pendulum design, 218
- Investigators, 83–112
 - accuracy versus precision, 109–111
 - compass design, 83–85, 87–90
 - electronic devices, 83–85, 87–103
 - gyroscope design, 104
 - heuristic vehicle path, 106–107
 - inertial navigation, 107–109
 - LEGO challenges, 99–111
 - navigation of a mobile robot, 104
 - nonorthogonality error, 103–105
 - nonparallelism error, 103
 - NXT as a compass, 103
 - problem solver exercises, 91–99
 - problem-solving methods by, 85–86
 - tire grip evaluation, 104–106
- J**
 - Jacquard, Joseph-Marie, 114
 - Jansen, Theo, 37
 - Jansen mechanism applications, 37, 51
 - Jones, William, 131
- K**
 - Kalman filter (KF), 412–424, 521–526
 - Kalman gain (K), 420
 - Kalman, Rudolf, 413–414
 - Keith, Mike, 463
 - Kekulé, Friedrich August, 36
 - Kilpatrick, William H., 386–387
 - Kinetics, *see* Motion
 - KMZ51 sensor, 84–85
 - Kneip, Laurent, 18, 219
 - Kolmogorov, Andrey, 213
 - Kremer, Gerhard, 55
 - Kroll, William Justin, 86
- L**
 - LabVIEW for LEGO MINDSTORMS (LVLM), 22–24, 42–43, 65–69, 70–72, 147, 152–154, 196–197, 230–232, 234–235, 267, 270, 276–277, 280, 302, 304–305, 311–313, 317–318, 320–323, 324–325, 329–332, 351–354, 358–362, 367–369, 373–375, 498–499, 501
 - Al-Khwäizmi navigation robot, 311–313
 - Bernoulli robot, 302, 304–305
 - calibration using, 65–69
 - Cleanup diagram, 147
 - Create Sub.vi tools, 147
 - Data Viewer, upload and comparison using, 329–330
 - data-logging subroutines, 234–235
 - delay function, 302–303
 - density function subroutines, 231–232
 - Driver Sub.vi, 196–197
 - enumerator list, 358–359
 - Fermat problem solution using, 276–277
 - finite state machine (FSM), 351–354, 360–363
 - Histogram.vi, 231
 - intelligent robot programs, 302, 304–305, 311–313, 317–318, 320–323, 324–325, 329–332

- LabVIEW for LEGO MINDSTORMS (LVLM)
(continued)
 intensity graphs, 71–72
 Linear Fit.vi, 230–232
 Main Application Instance, 42–43
 modeling subroutines 230–232, 234–235
 Nelson semaphore robot program, 320–323
 NXT test program, 65–69
 Piccard event-driven program, 324–325
 pseudorandom generator, 309–310
 robot design using, 22–24
 robot intelligence, 351–354, 358–362,
 367–369, 373–375
 shift registers, 42–43
 Solve Linear Equations.vi, 70–71
 sound2.llb library, 498–499, 501
 Std Deviation and Variance.vi, 231–232
 Tower of Hanoi program, 152–154
 tracking objects using, 196–197
 wait-for-pressed functions, 304
- Landmark for positioning, 71
 Laplace transform, 234
 Latitude, 54
 Leaky integrator, 366, 488–489
 Learn state, 366
 Learning programs, 346, 363–372
 adaptive (learning by doing), 346, 366–368
 dynamic, 346, 368–372
 reinforcement, 363–368
 robot intelligence and, 346, 363–372
- Least significant byte (LSB), 204
 Least squares, method of, 212–213, 230, 421
 Legacy motor, 38, 41–42, 145, 187, 279, 283–285
 LEGO Code Pilot, 114–116
 LEGO MINDSTORMS, 10, 13, 16–25. *See also*
 LabVIEW for LEGO MINDSTORMS
 (LVLM)
- LEGO Robotics Invention System (RIS), 10–25
 BIT rules for design using, 14–15
 development of, 10
 education using, 13–25
 LabVIEW for LEGO MINDSTORMS
 (LVLM), 22–24
 LOGO software and, 12
 MINDSTORMS, 10, 13, 16–25
 NXT design, 10, 21–23
 RCX design, 10, 13, 16, 17–21
- ROBOLAB software and, 16–21
 sensors for, 15
 Ultimate ROBOLAB (UR) software and,
 18–19
- Leine, Remco, 218–219, 390
 Light-dependent resistor (LDR), 263–264
 Light sensors, 33, 46–48, 65–69, 266–267, 270–271
 Lightning vehicle robot, 38, 41–43
 Limkilde, Peter, 75
 Line detection/following, 125–126, 356–363, 389,
 394–395
- Linkages, 61–63, 70–81
 automatons, 76–79
 Chebyshev, 62–63
 collision avoidance, 76–77
 “Consul,” the educated monkey, 73–74
 conversion functions, 70–71
 inventor solutions, 61–63
 inversor (Peaucellier), 62–63, 74–75
 Jansen, 37, 51
 pantograph, 77–79
 parabola points as a vehicle course, 75–76
 parallel motion, 61–62
 polynomial fit (conversion functions), 70–71
 RR plotter, 454–455
 teleoperation, 76–79
 Watt, 61–62
 way integration, 76–77
- Loyd, Sam, 169
 Locomotion, *see* Animal mimicry
 LOGO software, 12
 Longitude, 54, 107–108
 Longitude Act, 107–108
 Look-up tables, 132–133, 318–319
 Loudness, 485
 Louis XI, King of France, 143
 Lucas, Edouard, 149
 Ludolphine number, 131
 LUXPAK weather balloon payload, 26–28
- M**
- Machina Speculatrix, 119–120, 130
 Magnetic compass design, 57–58, 83–91
 Magneto-resistive permalloy sensors, 84–85

- Magnetotaxis, 259, 271
- Magnitude (distance), 79
- Main Application Instance, LabVIEW, 42–43
- Markov chain, 344, 347, 373–375
- Mars environment simulation, 174–175, 424–433
- Mars Rover telerobotic project, 174–182, 184–185
 - design process, 174–175
 - RX infrared communication for, 179–181
 - ROBOLAB for, 175
 - web server for, 175–179
- Martin, Fred, 13, 264–265
- Massey, Walter, 6
- Matarić, Maja J., 119, 243, 252, 343
- Maybeck, Peter S., 414–415
- Maze strategies, 120–123, 124–127, 141–142
 - algorithms for, 120–123, 141–142
 - Boyscout and Follower program, 123
 - deformation theorem for simple closed curves, 121
 - Dijkstra algorithm for, 141–142
 - perfect maze, 121
 - Pledge algorithm for, 121–123, 127
 - right- and left-hand rules, 121
 - robot project for, 124–126
 - Theseus mouse robot, 120
 - topology and, 121, 123–124, 141–142
- McCarthy, John, 245
- Mechanical Turk, 77–78
- Melde’s experiment, 482–485
- Michaelis, M. M., 218
- Microphones, 492–393
- Milios, E., 220–221, 506
- Mindsensors vision subsystem v3 for NXT (NXTCam3), 196–197
- MINDSTORMS, 10, 13, 16–25. *See also* LabVIEW for LEGO MINDSTORMS (LVLN)
- Mitra, Sugata, 9
- Möbius strip environment, 293–294, 389
- Modelers, 206–242
 - Blind Juggler robot, 216–218
 - catching a free-falling ball, model for, 239–240
 - closed-loop control, 215–216, 218–219
 - continuous-sound locator design, 219–222
 - data-logging subroutines, 234–235
 - degrees of freedom (DOF), 220–222
 - dynamics of physical systems, 207
 - Gaussian (normal) distribution and, 212–214, 227–228
 - Gauss’ linear fit method, 224–232
 - inverted pendulum design, 218–219
 - LabVIEW subroutines, 230–232, 234–235
 - LEGO challenges, 237–240
 - method of least squares, 212–213
 - Nash equilibrium, 206
 - open-loop control, 215–218
 - pendulum clock model, 232–238
 - PID controller model, 238–239
 - problem solver exercises, 223–237
 - problem-solving methods by, 206–207, 212–215
 - rational versus empirical methods, 206–207
 - residuals, calculation of, 225–232
 - rolling ball clock model, 237–238
 - seesaw robotic design, 207–212, 237
 - Segway PT design, 218–219
 - system and control theory for, 207
 - system state and, 211–212
 - trajectory calculations, 213–214
- Modulo function, 65
- Monosynaptic reflex arc, 261–262
- Montessori, Maria, 9
- Montessori education methods, 9, 11
- Moore robot, 346, 347–354. *See also* Finite state machines (FSM)
- Morphology, 292–294
- Morse, Samuel, 202
- Morse code, 202–203
- Most significant byte (MSB), 204
- Motion, 36–38, 44–55, 213–214, 278–281. *See also* Sensors
 - accelerometer functions, 49–50
 - bionically inspired designs, 50–51
 - brake mode, 281
 - Cartesian coordinates for, 54–55
 - coast mode, 281
 - counting pulses, 46–48
 - da Vinci’s flying machines, 51–54
 - dead reckoning, 44
 - direction master robot design, 39–41
 - driving the distance, 45–46

Motion (*continued*)

- driving the straight line, 45
- duty cycle, 278
- Jansen mechanism applications, 37, 51
- lightning vehicle robot design, 41–43
- light sensor, 33, 46–48
- multiple stops, 46
- pulse-width modulation (PWM), 278–281
- reaching the target, 46
- rotation sensor, 41, 49–50
- south-pointing chariot, 34, 51
- tachometer values, 48–49
- trajectory calculations, 213–214
- travel distance calculations, 48–49

Motors, 278–285

- Braitenberg vehicles, 285
- characteristic curve of transistor, 283–285
- data-logging configurations, 285
- LEGO challenges, 282–289
- pulse-width modulation (PWM), 278–281
- speed-regulated design, 282–283
- transistors, 281–282, 283–285

Multiple stops by a robot, 46

Mysterious behavior, 303–304

N

- Nahin, Paul, 463
- Naïve sort, 163–166
- Nakamura, Shuji, 85–86
- Napoleon's theorem, 142–143
- Nash, John, 206
- Nassi-Shneiderman-Diagram (NSD), 319–320
- National Marine Electronics Association (NMEA), 439
- Navigation, 44, 57, 69–70, 104, 107–109, 310–315, 389–444
 - accelerometer, 390–391
 - Al-Khwāizmi robot behavior, 310–315
 - Apollo guidance computer (AGC), 412–415
 - dead reckoning, 44, 57, 310–315, 393
 - indoor global positioning system (GPS), 397–470
 - inertial, 107–109
 - Kalman filter (KF), 412–424

- Mars environment simulation, 424–433
- mobile robot design for, 104, 310–315
- operational amplifiers (op-amps), 410–411
- perspective and, 389–392
- prediction and estimation, 411–424
- Serendipity robot boat, 433–442
- Stonehenge circle problem, 392–397
- terminology for, 69–70
- zero gravity, 390–391

Needles, compass use of, 83–84

Nehmzow, Ulrich, 292

Nelson, Lord Horatio, 323

Nelson robot semaphore behavior, 315–323

Never-give-up philosophy, 59, 63

Newton's inverse-square law, 285–286

Nic_3 robot design, 22, 508–516

Nicholas, Adrian, 53

Noble, David, 31

Nonorthogonality error, 103–104

Nonparallelism error, 103

Normal (Gaussian) distribution, 212–214, 227–228

Numerical derivation, 41

NXT hardware and software, 10, 21–23, 91–96, 103, 131–135, 195–197, 279–281, 303–305

Bernoulli robot using NXT-G 2.0 program, 303–305

compass configuration by, 103

design of, 10, 21–23

Dinsmore 1655 sensor connected to, 91–96

Mindsensors vision subsystem v3 for (NXTCam3), 196–197

optical connection, 195–196

pulse-width modulation (PWM), 279–281

serial communication to/from RCX/NXT, 195–196

sine and cosine calculation using, 131–135

O

Object avoidance, 76–77, 313, 360–362

Ohm, George Simon, 206–207

Ohm's law, 88, 100–101, 206–207

Omnidirectional holonomic platform (OHP), 275

- Open-ended projects, 383–519
 - brainstorming, 387
 - drawing, 445–480
 - education, instruction versus dictation, 383–385
 - navigation, 389–444
 - project-based learning, 386–388
 - sound, 481–519
- Open-loop control, 215–218
- Operational amplifiers (op-amps), 410–411
- Osborn, Alex, 387

- P**
- Pantograph, 78–79
- Papert, Seymour, 12, 27, 385
- Pappus of Alexandria, 256
- Parabola points as a vehicle course, 75–76
- Parallel motion, 61–62
- Parity check, 194
- Pascal, Blaise, 118
- Pascaline, 118, 130–131
- Path (track), 70
- Pauli, Wolfgang, 2–3
- Peaucellier, Charles Nicolas, 62–63
- Peirce, Benjamin, 463
- Pendulum clock model, 232–237, 238
- Pendulums, accelerometer functions using, 49–50
- Perception and reactions, 259, 482–488
- Perfect maze, 121
- Perspective, navigation and, 389–392
- Phase (angle), 79
- Photoresistor-based sensors, 15
- Phototaxis, 269, 271–272
- PHP-hypertext preprocessor, 197–198
- Physics by design method, 11
- Pi (π), calculation of, 131
- Piazzzi, Giuseppe, 213–214
- Piccard family achievements, 327
- Piccard robot, event-driven behaviour, 323–327
- PID controller model, 238–239
- Pierre-Simon, Marquis de Laplace, 234
- Piscopia, Elena Lucrezia Cornaro, 383
- Pitch, 485, 516–517
- Pi-writer, 463–470
- Pixels, 472–477
- Pledge algorithm, 121–123, 127, 162
- Plotter control, 445–448, 456–463, 477
- Poe, Edgar Allan, 77, 118
- Polar coordinates, 79–81
- Polar (RR) plotter, 454, 471–479
- Polarotaxis, 288
- Polya, George, 5–6
- Polynomial fit (conversion functions), 70–71
- Position, 44, 46, 54, 70, 71, 389–444, 454–471, 484–485, 505, 510–515
- POST command, 178–179
- Precedence effect, 15, 501–502
- Precision versus accuracy, 109–111
- Predict-correct algorithm, 415–424, 477
- Prediction and estimation, 411–424
 - Apollo guidance computer (AGC), 412–415
 - estimation error variance, 422–424
 - filtering property, 415–416
 - Kalman filter (KF), 412–424
 - recursive predictor/corrector algorithm, 418–420
 - samples, 416
 - state estimates, 418
- Primary Guidance, Navigation and Control System (PGNCS), 412–413
- Printed circuit board etching, 96–99
- Probability, 344–346, 348–349, 364–366, 368–378
 - conditional, 376–377
 - decision graphs, 368–369
 - distribution, 369–371, 373–374
 - random determination, 348–349
 - transition, 344–346, 364–366, 368–372, 378
 - vectors, 373–375
- Problem solving, 1–242
 - bee “waggle dance,” 5
 - challenges of, 25–29
 - code breakers, 174–205
 - communication and, 5
 - conquerors, 140–173
 - discoverers, 31–56
 - education and, 6–16
 - Egg of Columbus, 2–5
 - electronic devices, 83–85, 87–103
 - Fermat’s Last Theorem, 2
 - Hole-in-the-Wall Education project, 9
 - inventors, 57–82
 - investigators, 83–112

Problem solving (*continued*)

- LEGO robotics used for, 1
 - LEGO Robotics Invention System (RIS), 10–25
 - mathematics and, 5–9
 - modelers, 206–242
 - never-give-up philosophy, 6, 59, 63
 - programmers, 113–139
 - puzzles and, 3–4
 - researcher role in, 5, 9
 - science and engineering for, 6–7
 - strategy, tactics, and tools for, 2–6
 - thinking outside the box, 3–5
 - tinkering, 57–64
 - Tippe top, 2–3
 - trial-and-error approach, 57, 59, 63
- Program for International Student Assessment (PISA), 9
- Programmers, 113–139
- agents (function and program), 114
 - algorithms, 116–117, 120–123, 127–129
 - Al-Jazari’s musical “robot band” design, 113–114
 - artificial intelligence and, 114, 120
 - comparison of automata, computers, and robots, 117–120
 - digital numbers, 136–138
 - Jaquard loom design, 114
 - LEGO challenges, 127–138
 - LEGO Code Pilot, 114–116
 - maze strategies, 120–123, 124–126
 - problem solver exercises, 124–127
 - robot requirements of, 113–114
 - robots and, 113–114, 117–120, 122–124, 124–126, 127–138
 - visualization (imagination) of, 120–121, 124
- Programming, 113, 124–136
- binary number implementation, 126–127
 - digital numbers and, 136–138
 - Heron’s algorithm, 128–129
 - LEGO challenges, 127–136
 - look-up tables, 132–133
 - Machina Speculatrix, 130
 - maze strategy for robot, 124–126
 - NXT calculation of sine and cosine, 131–135

- Pascaline, 130–131
 - Pledge algorithm for robots, 127
 - Shannon’s Ultimate Machine, 129–130
 - square root calculation, 128–129
 - Turing machine, 126–127, 130
- Project-based learning, 386–388
- Pseudorandom generator, 309–310
- Pulse width modulation (PWM), 278–281
- Pythagoras, 60–61
- Pythagoras’ theorem, 394, 450–451

Q

- Quicksort, 164–165

R

- Rabinowitz, Stanley, 464–465
- Rádl, Emanuel, 269
- RampPattern.vi, LabVIEW, 231–232
- Random numbers, 307–310
- Random walk algorithms, 297–302
- Rational and irrational numbers, 167–168
- Rationalist methods, 6–7, 206–207
- RCX controller device, 10, 13, 16, 17–21, 88–89, 179–182, 195, 501–504
 - compass driver, 88–89
 - development of, 10
 - firmware for, 501–504
 - infrared communication, 179–182
 - LEGO RIS design, 13, 16, 17–21
 - serial communication to/from NXT, 195
- Reaction delays, 335
- Recursive functions, 148–155
- Recursive predictor/corrector algorithm, 418–420
- Reid, G. L., 220–221, 506
- Reinforcement learning, 363–372
- Reist, Philipp, 216–218
- Repetitive “buggy” behavior, 304
- Researcher role in problem solving, 5, 9
- Residuals, calculation of, 225–232
- Resistors, 100–101
- Resource starvation, 316
- Reverse engineering, 88
- Rhombic dodecahedron, 256, 272
- Ries, Adam, 116, 164
- ROBOLAB software, 16–21, 150–151, 175
 - development of, 16–21
 - GASTON use of, 16–19

- Mars Rover telerobotic project, 175
 - Tower of Hanoi problem using, 150–151
 - ultimate (UR), 18–19, 150–151, 501
 - Robot Institute of America (RIA), 113
 - Robotics, 207–212, 243–381, 471–479
 - animal cognition and, 244
 - artificial intelligence (AI) and, 245–248
 - automata and, 244–245
 - behavior of robots, 248–251, 263, 292–341
 - Braitenberg vehicle designs, 259–265, 268–269
 - bugs, 254–291
 - calligraphy using, 467–479
 - control methods for, 252
 - convergence of a sequence, 268–271, 286–288
 - Deep Blue, 245–246
 - development of robots, 243–247
 - drawing hand design, 471–479
 - Fermat problem, 272–278
 - HAL9000, 246–247
 - Helmholtz vehicle design, 266–267
 - intelligence and, 244–245, 292–294
 - intelligent robots, 292–341
 - problem-solving perspective of, 243–253
 - seesaw design, 207–212
 - self-awareness and, 250–251
 - sensors and, 260–271
 - Steiner points, 257–258, 273–274
 - Turing test, 245
 - Watson, 246
 - Robots, 113–114, 116–120, 122–124, 124–126, 143–146, 168–169, 196–199, 213–221, 243–251, 263, 274–276, 288–289, 292–341, 342–348
 - architecture of, 116
 - artificial intelligence (AI) and, 119, 245–247
 - automata and computers compared to, 116–120
 - behavior of, 248–251, 263, 292–294
 - closed-loop control, 215–216, 218–219
 - complex, design of, 168–169
 - continuous-sound locator design, 219–222
 - control of over the web, 196–199
 - degrees of freedom (DOF), 220–222
 - development of, 243–247
 - don't think-react! design, 274–275
 - emotions expressed by, 143–146
 - Fermat problem-solving designs, 273–276
 - intelligence of, 342–381
 - intelligent, 292–341
 - invention of, 119–120
 - inverted pendulum design, 218–219
 - maze strategies for, 124–126
 - modelers and, 213–222
 - open-loop control, 215–218
 - polarotaxis functions, 288–289
 - programmers and, 113–114, 116–120
 - Robot Institute of America (RIA) definition, 113
 - steady state of, 263
 - think-hard-act-later design, 273–274, 296–297
 - trajectory calculations, 213–214
 - Rodney robot, 346, 355–363. *See also* Finite state machine (FSM)
 - Rogers, Chris, 11, 16, 18, 343, 390
 - Rolling ball clock model, 237–238
 - Rosetta Stone, 183
 - Rotation sensors, 41, 49–50, 311–315
 - Rudder control, 440–441
 - Rule of thumb, 449–451
- S**
- Samples, 416
 - Sampling problems, 375–376, 379–380
 - Sampling theorem, 379
 - Satellite perspective, 426
 - Schlitt, Herbert, 213
 - Schmidt, Stanley F., 413
 - Science, technology, engineering, and mathematics (STEM), 7
 - Seesaw robotic design, 207–212, 237
 - Segway PT design, 218–219
 - Self-awareness, robotics and, 250–251
 - Semaphore algorithm, 315–323
 - Sense-act processing, 255
 - Sense-think-act processing, 255, 294
 - Sensor watcher task, 323–325

- Sensors, 15, 33, 38–43, 46–50, 60, 65–69, 83–85, 87–88, 260–271, 295–296, 304, 311–315, 317–321, 333–334, 357–363, 366–368, 489–491
 - angle, 65–69
 - Braitenberg vehicles, 260–265, 268–269
 - calibration of, 65–69
 - combined light and touch, 357–363
 - compass design using, 83–85, 87–88
 - differential equations for, 39–41
 - dual, reactions to, 260–265, 270–271
 - electronic, 87–88
 - Hall-effect, 83, 87–88
 - Helmholtz vehicle design, 266–267
 - intelligent robot behavior and, 295–296, 304, 311–315, 317–321, 333–334
 - KMZ51, 84–85
 - legacy motor, 38, 42–43
 - light, 33, 46–48, 65–69, 266–267, 270–271
 - multiplexer “mood-meter” design, 60
 - photoresistor-based, 15
 - precedence effect from, 15
 - robot intelligence and, 357–363, 366–368
 - robotics and, 260–267
 - rotation, 39–41, 49–50, 311–315
 - shaft encoders and, 39–41, 65–69
 - sound, 489–491
 - touch, 295–296, 304, 311–312, 317–321, 333–334, 366–368
 - values, reactions to, 260–265
- Serendipity robot boat navigation, 433–442
- Sessa, 155
- Seven bridges of Königsberg problem, 302, 337–339
- Shaft encoders, *see* Encoders
- Shannon, Claude, 120, 129, 216, 379
- Shannon’s Ultimate Machine, 129–130
- Shift registers, LabVIEW, 42–43
- Signal semaphore, 323
- Signals and messages, 202–203, 315–323, 398–410
 - infrared/ultrasonic indicators, 398–410
 - Morse code, 202–203
 - semaphore algorithm, 315–323
- Simon, Herbert A., 248
- Simpson, Thomas, 273
- Sine/cosine calculation, 131–135, 156–162
- SmartBird, 53
- Sound, 219–221, 481–519
 - acoustic localization, 504–516
 - analyzing, 492–504
 - animal mimicry and, 481, 488–491
 - animats, 488–491
 - crickets, 488–491
 - cross-correlation function, 495–498
 - cues, 505
 - digital signal processing (DSP), 493–495
 - ear, system of, 482–483, 485
 - interaural time differences (ITD), 505, 507–510
 - LabVIEW sound2.llb library, 498–499, 501
 - location of sources in space, 219–221
 - Melde’s experiment, 482–485
 - microphones, 492–494, 510–512
 - Nic_3 robot design, 508–516
 - perception, 482–488
 - RCX firmware for, 501–504
 - robot implementation of, 506
 - sensors, 489–491
 - spatial direction of, 510–512
 - spectral analysis, 498–500
 - time delay (lag), 501–503
- Sound pressure levels (SPL), 488
- South-pointing chariot, 34, 51
- Spaghetti code, 146–147
- Spatial direction of sound, 510–512
- Spectral analysis, 498–500
- Speed (velocity), 70
- Sperling, Walter, 150
- Spigot algorithm, 464–467
- Square root calculation, 128–129
- Star Wars problem, 339–340
- State estimates, 418
- State space, 343–344
- State transitions, 344–346, 353–354
- State variables, 343–344
- Steady-state errors, 239
- Steady state of vehicles, 263
- Steiner points, 257–258, 273–274
- Stigmergy, 255, 259
- Stochastic matrix, 373
- Stochastic process, 230
- Stonehenge circle problem, 392–397
- Straight line, robot driving of, 45

Strategy for problem solving, 6
 Structogram, 319–320
 Stuart, David, 183
 Subsumption architecture, 248, 346, 355–363, 377
 Sun Tzu, 143
 Sweep scan algorithm, 297
 Synchronization of locomotion, 36–37
 Synthetic psychology, 259–260, 268–269
 System and control theory, 207
 System state, 211–212

T

Tachometers, 41–43, 48–49
 Tactics, 6
 Target, reaching goal of, 16
 Taxicab geometry, 172
 TCP listener, 177–178
 Teleoperation, 76–79. *See also* Automata
 Telerobotics, 174–182, 184–185, 195–199
 data communication, 179–181
 downloading HTML files, 196–199
 Mars Rover project, 174–182, 184–185
 Mindsensors vision subsystem v3 for NXT (NXTCam3), 196–197
 NXT to RCX serial communication, 195
 optical NXT to NXT connection, 195–196
 PHP-hypertext preprocessor, 197–198
 RCX to NXT serial communication, 195
 robot control over the web, 196–199
 tracking objects using LabVIEW, 196–197
 web server for, 175–179
 Thales' theorem, 394
 Theory of indivisibles, 273
 Theseus mouse robot, 120
 Think and act separately robot design, 275
 Think-hard-act-later robot design, 273–274, 296–297
 Think-the-way-you-act robot design, 275
 Thomson, Sir William (Lord Kelvin), 63
 Thresholding, 46–48, 472–476
 Thunderbolt signal concept, 398–404
 Tilt error, 89–90
 Timbre, 485
 Time delay (lag), 427–433, 501–503, 512
 Time difference of arrival (TDOA), 276
 Tinkering, 57–63. *See also* Inventors

Tippe top, 2–3
 Tire grip evaluation, 104–106
 Toepler, August, 129
 Tools, 6
 Topology and maze strategy, 121, 123–124
 Torricelli, Evangelista, 272–273
 Touch sensors, 295–296, 304, 311–312, 317–321, 333–334
 Tower of Hanoi algorithm, 148–155
 Track (path), 70
 Tracking objects using LabVIEW, 196–197
 Trajectory calculations, 213–214
 Trammel, 99–100
 Transistors, 281–282
 Transition probability, 344–346, 366, 368–372
 Transition tables, 344, 351
 Transmission Control Protocol/Internet Protocol (TCP/IP), 175–178
 Transmission shaft-encoder, 33–34
 Travel distance calculations, 48–49
 Travelling salesman problem (TSP), 296
 Trial-and-error approach, 57, 59, 63
 Trigonometry, 7–8, 448–454
 drawing programs use of, 448–454
 educational methods for, 7–8
 Tuft's Center of Engineering Educational Outreach (CEEEO), 19–21
 Turing, Alan, 118–119, 245
 Turing machine, 116, 118–119, 126–127, 130
 Turing test, 245
 Turning by a constant angle algorithm, 300–302
 "Turtle graphics," 12
 Two's complement numbers, 179–180

U

Ultimate Machine (Shannon's), 129–130
 Ultimate ROBOLAB (UR) software, 18–19, 150–151, 501
 Uniform Resource Locators (URL), 196–199

V

Valéry, Paul, 6
 Valin, J.-M., 506
 van Ceulen, Ludolph, 131, 464
 van Soest, Johannes Leendert, 220–221
 Vectors, 81

Vehicle course of five parabola points, 75–76
Vehicle path, heuristics of, 106–107
Velocity (speed), 70
Vernier, David, 105–106
Viète, François, 464
Virtual bee robot, 296–302
Visualization (imagination) of programmers,
120–121, 124
Volder, Jack E., 157–159
von Békésy, Georg, 482
von Frisch, Karl, 5, 182–183
von Helmholtz, Hermann, 265, 482
von Kempelen, Johann Wolfgang Baron de
Pázmánd, 77
von Lindemann, Ferdinand, 463
von Neumann, John, 117, 319
von Zach, Franz Xaver, 214

W

Wagon, Stan, 464–465
Wait functions, 304, 323, 361
Walking mechanisms, 37
Walking the grid problem, 335
Wallach, H., 220

Walter, W. Grey, 119–120
Wang, Eric, 21
Water-rockets, 53
Watson, 246
Watt, James, 61–62
Way integration, 76–77
Weather balloon payload, 27–28
Web, *see* Internet use
Web server for Mars Rover project 175–179
Webb, Barbara, 249–250, 488–489
Wiles, Andrew, 2
WWII "Ultra Secret," 183–184

X

xy-plotter robot, 19–20
xy-plotter, 454–463

Z

Z-series computers, 116–117
Zeitz, Paul, 6, 28
Zero gravity, 390–391
Zig-zag scan algorithm, 297
Zuse, Konrad, 116–117