Index

ADEPT-50, 163, 166
Advanced Research Projects Agency. *See* ARPA (Advanced Research Projects Agency)
Allaire, Frank, 127, 130, 132, 132–133, 135–136, 139–140, 142
AM (automated mathematician), 15, 95–98, 279
Anantharam, Siva, 289
Anderson, James P., 160
 Anderson panel, 160, 163, 166, 174
Anti-ballistic missile systems, 57, 214, 299–300
Apollo, 26–29, 42, 57, 59
Appel, Kenneth, 15–16, 101–102, 117, 119, 121, 127, 129–149, 317, 321
Argonne National Laboratory, 77–78, 80, 88–89, 98, 290, 294, 314
ARPA (Advanced Research Projects Agency), 16, 51, 82, 155, 163, 168, 174–175, 177, 285, 299
ARPAnet, 168–169, 174, 183
Artificial intelligence, 4–5, 14–15, 47–48, 63–71, 74, 79–83, 85–98, 133, 238, 262–264, 266–269, 276, 278, 308, 310–311
Atiyah, Sir Michael, 317–319
Authority, 2–3, 20, 303–306
Autodin II, 169–170, 180
Automated mathematician (AM), 15, 95–98, 279
AUTOMATH, 98–99, 285, 322
Axiom of choice, 203–282
Backus, John, 47, 49
Baker, F. Terry, 39
Ballistic Missile Early Warning System (BMEWS), 23–24
Barton, Robert S., 35
Barwise, Jon, 232–324
Basic security theorem, 165–166, 182
Batch operation, 153
Bates, Joseph L., 283–284
Bauer, Friedrich, 34, 36
Beck, Ulrich, 7–8, 301
Bell, David Elliott, 16–17, 163–165, 170–171, 174, 183–185
 Basic security theorem, 165–166, 182
 Bell-LaPadula model, 16–17, 163–166, 169–172, 181–185, 189–190, 195–196
Berkeley, Edmund, 64
Bernhart, Frank, 129, 130, 139
Beth, Evert W., 75, 265
Bevier, William, 215, 217, 244
Bieberbach, Ludwig, 260
Binary decision diagram methods, 252–253, 272
Birkhoff, George, 117, 121
 "diamond," 121–131
 number, 122, 135
Bishop, Errett, 261, 281
BLACKER, 181, 190–191
Bledsoe, Woodrow W., 86–88, 167, 237, 262, 264, 310
Bloor, David, 2, 184–185, 286, 313, 332

BMEWS (Ballistic Missile Early Warning System), 23–24
Boehm, Barry, 32–33
Böhm-Jacopini theorem, 57
Bonsall, Frank, 102
Boyer-Moore prover, 168, 179, 232–233, 238, 240–242, 252, 254, 262–270, 272–273, 280, 291–294, 296, 313, 322
Boyer, Robert S., 168, 236–237, 241–243, 254–255, 263–271, 291–294, 313–315. *See also* Boyer-Moore prover; Computational Logic, Inc.
 attitude to decision procedures, 295
 and complexity of theorem proving, 86
 critique of "design proof," 178–179
 and SIFT, 19, 223, 232–233, 238–241, 254–255
Brock, Bishop C., 242–243, 249–251, 254
Brooks, Frederick P., Jr., 32, 42, 56
Brouwer, L. E. J., 91, 257–261, 265, 280–282, 319
Brun's sum, 219–220
Bryant, Randal E., 252–253
Bundy, Alan, 265, 315, 330
Burstall, Rod, 263, 273, 276
Butler, Ricky W., 234
Buxton, John, 34–35
Byzantine faults, 18, 229–230

Cauchy, Augustin-Louis, 108–112
Cayley, Arthur, 104, 111, 147
Certainty trough, 333–334
Charter Technologies, Ltd., 247, 249
Chessboard, problem of mutilated, 325–326
Chiaraviglio, Lucio, 199
Chief programmer teams, 39, 41
Choice, axiom of, 203, 281–282
Chomsky, Noam, 48
Church, Alonzo, 73, 75, 93, 199, 206, 274, 279, 282–283
 Church-Turing thesis, 206, 274–275
Clarke, Edmund, 234, 251–252, 254
Classical logic, 258, 279, 281, 283, 285–287, 313, 327
Cleanroom, 20, 57–59, 303, 306–307, 310
CLInc stack, 243–244, 268
Clock convergence algorithm, 18, 229–230, 272
Coe, Tim, 221
Cohen, Daniel I. A., 144, 147, 317–318
Cohen, Paul, 203, 282
Cohn, Avra, 214–215, 217, 247–251, 276–277, 279, 314, 334
Collins, Harry, 12, 16, 139, 302–303, 321, 332, 334
Colmerauer, Alain, 82–83
Combinatorial explosion, 68–69, 76, 79, 83–85, 264. *See also* Complexity theory
Complexity theory, 83–86, 252, 283
Computational Logic, Inc., 196, 215, 217, 241–245, 249, 254, 272
Computer Security Evaluation Center, 175–176, 180, 194, 234. *See also* National Computer Security Center
Constable, Robert L., 282–288, 290, 295–296, 314
Constructivism, 261, 265, 280–287, 313–314, 327, 329
Continuum hypothesis, 203, 282
Cook, Steven, 84–85
Core set, 15, 139, 147
Covert channels, 16, 166, 171–172
Craigen, Dan, 288
C-reducibility, 122–124, 132, 134, 136
Cultures of proving, 13, 20–21, 306–316, 319, 321–322, 327–328
Cyberwar, 152–153

Davis, Martin, 73, 76–78, 84–85, 93, 98
 Davis-Putnam procedure, 76–78
de Bruijn, N. G., 98, 285
Decision procedures, 73–75, 85, 267–270, 272
 controversy over unverified, 294–295, 313–314

Deduction, 2–4, 11–12, 17, 305–306. *See also* Formal proof; Formal verification; Logic
DeMillo, Richard A., 17–18, 197–210, 212–214, 216, 218, 232, 272, 284, 309, 320, 323–324, 326, 328
 and SIFT peer review, 234–235
De Morgan, Augustus, 72, 103–104
Denning, Peter J., 206, 216–217
Deskilling of programming, 39–40
Dijkstra, Edsger W., 26, 53, 56, 241, 273, 313
 attitude to proof, 20, 60, 208–210, 311, 327
 controversy with DeMillo, Lipton, and Perlis, 17–18, 197–198, 201, 208–210
 critique of testing, 3, 45, 197
 and Garmisch meeting, 14, 36, 197
 and **go to**, 38–39, 204
 and handwriting, 311–312
 and structured programming, 37–41, 302
 and synchronization primitive, 200
Discharging algorithms, 121, 129, 134–135, 140, 142, 147
Divide bug, 219–223, 251, 253, 315, 331
Dobson, John, 217–218
D-reducibility, 122–127, 132, 134, 136
Dreyfus, Hubert L., 90, 330
Dunham, Bradford, 74–75
Dürre, Karl, 123–126, 132, 136

EDSAC (Electronic Delay Storage Automatic Calculator), 25–26, 273, 276
Edwards, Daniel, 164
EHDM (Enhanced Hierarchical Development Methodology), 271–272, 288
Einstein, Albert, 18, 228
Encrypted Packet Interface, 168, 177–178
ε-operator, 280–281, 286
Euler's polyhedral formula, 110–111, 119, 121, 146, 184
Excluded middle, 258–259, 261, 280–281, 283, 286–287, 313, 319
Explosion, combinatorial, 68–69, 76, 79, 83–85, 264. *See also* Complexity theory

Fagan, Michael, 45–47
Fault tolerance, 223–232, 236
FDM (Formal Development Methodology), 179–181, 288
Feiertag, Richard J., 168
Feigenbaum, Edward A., 63, 96
Fermat's last theorem, 85, 135, 141, 148, 322
Fetzer, James H., 18, 205, 210–218, 241, 244, 247, 272, 309, 323
First-order logic, 72, 263, 270, 274, 331
Floyd, Robert W., 14, 48, 51–55, 60–61, 167, 208
Formal Development Methodology (FDM), 179–181, 288
Formal proof, 12–13, 17–18, 20–21, 41, 91, 93, 98–99, 201–204, 207, 209, 213, 218, 248–251, 261, 307–318, 321–328, 330–332
Formal semantics, 47, 51–54, 199, 274–275, 278
Formal specification, 4, 170, 178, 180–181, 196, 248, 270, 283, 290
Formal verification, 4, 56, 140, 166–167, 170, 172, 195–199, 201, 204–208, 210, 214, 225, 232, 248–251, 255, 262, 283, 300, 303, 305–306
Four-color problem, 4, 15–16, 85, 99, 101–149, 317, 321
Fourman, Michael, 287
Frege, Gottlob, 72, 93, 309
Friedman, Andrew, 40–41

Galison, Peter, 18, 228, 328
Garmisch conference, 14, 20, 34–37, 55, 197, 299
Gelernter, Herbert, 69–70, 72, 75, 80, 88, 308

Geometry machine, 69–70, 90
Gerhart, Susan, 56, 205, 210, 234
Giddens, Anthony, 7, 10
Gilmore, Paul, 75–77
Gödel, Kurt, 199, 206
 incompleteness theorems, 15, 52, 90–94, 129, 268, 293, 318, 322
 and Platonism, 91
Good, Donald I., 167–168, 177, 179, 184, 234, 241–243, 288
Goodstein, R. L., 265
Gordon, Michael J. C., 214–215, 217, 276, 278–279, 290, 295–297
Go to instruction, 38–40, 57, 204
Graph theory, 101, 106–111, 120–121, 126–127, 148, 200, 209, 285
Gries, David, 36, 140–141, 311, 313
Grove, Andrew, 221, 223
Gruenberger, Fred, 42
Guthrie, Francis, 103
Gypsy Verification Environment, 167–168, 179–180, 288

Haken, Wolfgang, 15–16, 101–102, 117, 119–121, 124–149, 317, 321
Hales, Thomas, 146
Hall, Eldon, 29
Halmos, Paul, 102, 147
Hamilton, William Rowan, 103–104
Handwriting, 311–312
Hanna, Keith, 96–97, 279
Hardy, G. H., 96
Hartmanis, Juris, 83, 283
Hayes, Pat, 76, 81–83, 86, 93
HDM (Hierarchical Development Methodology), 179, 270–271
Heawood, Percy, 114–120
Heesch, Heinrich, 120–125, 127, 129–131, 133, 136–137
Herbrand procedure, 72, 76
Heuristics, 15, 66, 74–75, 87–90, 95, 264, 266, 268, 270, 272
Hewitt, Carl, 82, 87
Heyting, Arend, 261, 265, 281
Hierarchical Development Methodology (HDM), 179, 270–271
Higher-order logic. *See* HOL (higher-order logic)
High-water mark model, 163
Hilbert, David, 91, 146, 251, 258–259, 280, 308, 319, 328–329
Hoare, C. A. R. [Tony], 14, 20, 52–56, 167, 201, 299–306, 311
 Hoare paradox, 20, 301–306
Hoffman, Daniel, 304
HOL (higher-order logic), 20, 247, 249–250, 252, 262, 277–281, 284–287, 289–291, 293–298, 313–314, 316, 327
Horgan, John, 145
Horning, Jim, 205–206
"Horseshoe" configuration, 125–127, 130
Human-computer interaction, 300, 303, 334
Hume, David, 11–12, 14
Hunt, Warren A., Jr., 242–243, 249–251

IBM, 14, 23, 26, 34, 74–75, 211, 221
 cleanroom, 20, 57–59, 303, 306–307, 310
 Federal Systems Division, 57, 59
 OS/360, 31–32
 and program review, 45–46
 and structured programming, 39–41
 Time Sharing System, 156
ICL (International Computers, Ltd.), 280, 297–298
Impredicative type theory, 286
Incompleteness theorems, 15, 52, 90–94, 129, 268, 293, 318, 322
Induction, 2–3, 11, 14, 20, 303–306. *See also* Testing
 mathematical, 263, 266–267, 322
Integrity, 195
Intel Corporation, 220–223, 253. *See also* Pentium™ processor
Intelligent exhaustive simulation, 246, 248–249
Interactive convergence algorithm, 18, 229–230, 272

International Business Machines. *See* IBM
International Computers, Ltd. (ICL), 280, 297–298
Intuitionism, 91, 258–261, 265, 282, 284

Jaffe, Arthur, 319
Jelen, George F., 173–174
Jones, Cliff, 315
Jones, Roger, 297–298, 327

Kaiser, Andreas, 220–221
Kemmerer, Richard A., 288–289
Kempe, Alfred Bray, 104, 106, 111–121, 137
 Kempe chains, 112–114, 118
Kepler conjecture, 146
Kernel. *See* Security kernel
Kershaw, John, 251, 255
King, James Cornelius, 14, 61, 167
Kleene, Stephen C., 199, 274, 282–283
Kling, Rob, 212
Knot problem, 128–129
Knuth, Donald, 95, 200
Koch, John, 101, 132, 134–136, 139–140
Kowalski, Robert, 82–83, 262–263
Kraft, Philip, 39–40, 57
Krantz, Steven G., 145, 149
Kunen, Kenneth, 269, 322
Kusch, Martin, 321

Lakatos, Imre, 2, 17, 108–111, 146, 184–185, 189, 206, 209, 323, 330
Lam, Clement, 144–146
λ-calculus, 94, 186, 274–277, 279, 282, 285, 287, 331
Lamport, Leslie, 18, 208, 216, 228–229, 232, 272
Lampson, Butler, 171
Landin, Peter, 53, 273–274
LaPadula, Leonard J., 16–17, 163–165, 171, 174, 183–184. *See also* Basic security theorem; Bell-LaPadula model
LCF (Logic for Computable Functions), 246, 262, 264, 276–279, 281, 284, 287, 293–296, 313
LCF_LSM, 246, 278–279
Lederberg, Joshua, 96
Leibniz, Gottfried, 15, 17–18, 71–72, 209, 237
Lenat, Douglas B., 15, 95–97, 279
Levitt, Karl, 168, 288
Licklider, J. C. R., 30, 33, 299–300
Lighthill, Sir James, 83, 98, 262
Linger, Richard C., 57–59
Lipner, Steven, 191
Lipton, Richard J., 17–18, 197–198, 200–210, 212–214, 216, 218, 232, 272, 284, 309, 320, 323–324, 326, 328
 and SIFT peer review, 234–235
Livingston, Eric, 3, 306, 308, 317, 319–320
Logic
 classical, 258, 279, 281, 283, 285–287, 313, 327
 and computer science, 329–330
 constructivist, 261, 281, 283, 285–287, 313, 327
 as a discipline, 73–74, 308–309, 329–330
 first-order, 72, 263, 270, 274, 331
 higher-order, 20, 278–280 (*see also* HOL [higher-order logic])
 predicate, 72–82, 87, 275, 279, 318
 programming, 82–83
 propositional, 4, 14, 65–68, 70–71, 73–76 (*see also* Binary decision diagram methods; Davis-Putnam procedure; *NP*-complete problems; Truth tables)
 relevance, 286, 313
 temporal, 251–252, 278, 285–286, 329
Logic Theory Machine, 14, 63–70, 73–74, 78, 89–91, 96–97, 275, 307
London, Ralph, 56, 167, 202
Loveland, Donald, 69, 80, 234
Lucas, John, 15, 52, 92–93

Mahoney, Michael S., 39, 48
Manin, Yuri I., 321
Mannheim, Karl, 2
Map
 definition of, 104, 147
 minimal, 118–120, 123
 normal, 117–119, 121, 125
Mariner space probe, 25
Martin-Löf, Per, 281, 285
Mathisen, Terje, 220
Maurer, W. D., 207–208
May, Kenneth, 103
Mayer, Jean, 135, 142
McCarthy, John, 14, 47–48, 51, 60, 69, 80–82, 263, 275, 282
McCullough, Daryl, 186–188
McCune, William, 88–89, 290
McIlroy, Doug, 36, 39, 77
McLean, John, 17, 181–186, 189
Melliar-Smith, Michael, 229–230, 232, 235, 269–272
Meltzer, Bernard, 80
Millennium bug, 1, 8, 300, 302
Miller, William F., 78
Mills, Harlan, 14
 and chief programmer teams, 39, 41
 and cleanroom, 56–59, 306
 and **go to**, 57
 view of proof, 14, 58–59, 307
 Milner, Robin, 273–279, 284, 287, 331, 334. *See also* LCF (Logic for Computable Functions); ML
 attitude to decision procedures 295, 314
Mimeomorphic action, 321
Minimal map, 118–120, 123
Minsky, Marvin, 48, 69–70, 80–83, 90, 238
MITRE Corporation, 16, 162–163, 170, 184, 195
MIZAR, 99, 322
mk_thm, 20, 296–298, 316
ML, 276–277, 285, 293–295
Model checking, 19, 251–255, 285, 301, 315, 328–329, 333
Modus ponens, 67–68, 307, 317
Moore, Edward F., 117
Moore, J Strother, 168, 237–238, 241–244, 254–255, 263–271, 291–293, 296, 313–316. *See also* Boyer-Moore prover; Computational Logic, Inc.
 and Apollo software, 29, 238
 attitude to decision procedures, 295
 and complexity of theorem proving, 86
 critique of "design proof," 178–179
 and SIFT, 19, 223, 232–233, 235–236, 238–241, 255
Moral entrepreneurship, 20, 301–302
Mutation testing, 204–205
Mutilated chessboard problem, 325–326

NASA (National Aeronautics and Space Administration), 44, 59, 224, 233, 272, 292
National Computer Security Center, 175–176, 178, 181–183, 190–191, 193–194, 270–271, 288
National Security Agency. *See* NSA (National Security Agency)
Naur, Peter, 14, 38, 48–51, 53, 311
Naval Research Laboratory, 181–182, 189
"Neats," 82, 96, 264
Neumann, Peter, 168, 179, 299–300
Newell, Allen, 14–15, 63–70, 73–75, 78, 87–90, 94, 97, 264, 268, 275, 307–308
Nibaldi, Grace, 177
Nicely, Thomas, 219–221
Nidditch, Peter, 322, 326–327
Nondeducibility, 186–187, 187
Noninterference, 185–189
"No read up" rule, 159, 163–165
Normal map, 117–119, 121, 125
Normativity, 21, 332
"No write down" rule, 164–165
NP-complete problems, 84–85, 252
NSA (National Security Agency), 16–17, 76, 155–158, 164, 166–167, 173–177, 180–182, 185, 190, 194–196, 210, 296–297, 310, 316

Nuclear early warning, 23–24, 303
Nuprl, 262, 284–286, 288, 290–291, 293

Odyssey Research Associates, 185–186, 196, 234
One way regulator, 280
Orange book, 16–17, 177–181, 191, 193–195, 245, 271, 280, 288
Ordered binary decision diagram methods, 252–253, 272
OS/360, 31–32

Papert, Seymour, 81–82, 238
Paris-Harrington theorem, 269, 322
Pease, Marshall, 229
Penrose, Roger, 15, 92–94, 98, 274
Pentium™ processor, 219–223, 251, 253–254, 315, 331
Perlis, Alan J., 17–18, 35, 197–198, 200–210, 212–214, 216, 218, 232, 234, 272, 284, 309, 320, 323–324, 326, 328
Peters, Bernard, 155–158, 166, 174
Platek, Richard, 186, 234
Poincaré conjecture, 128–129
Polimorphic action, 321
Pollack, Randy, 285, 289–290, 294, 331
Polya, George, 66, 95–96
Porter, Theodore, 9–11, 13
Prawitz, Dag, 75–79
Predicate logic, 72–82, 87, 275, 279, 318
Presburger arithmetic, 73, 85–86, 294
Primitive recursive arithmetic, 265–266, 331
Principia Mathematica, 65–67, 71, 74, 264, 307–308, 329
Professions, 3
Proof. *See* Formal proof; Rigorous argument proof
ProofPower, 280, 327. *See also* HOL (higher-order logic)
Propositional logic, 4, 14, 65–68, 70–71, 73–76. *See also* Binary decision diagram methods; Davis-Putnam procedure; *NP*-complete problems; Truth tables
Proving, cultures of, 13, 20–21, 306–316, 319, 321–322, 327–328
PSOS (Provably Secure Operating System), 168–170
Putnam, Hilary, 76–78, 84, 98
 Davis-Putnam procedure, 76–78
PVS (Prototype Verification System), 271–272, 295, 313

QED, 99
Quinn, Frank, 319

Rabin, Michael, 83, 85–86
Ramanujan, Srinivasa, 96
Randell, Brian, 217–218
Random testing, 42–44, 46
Recursion, 265–267
Reducer, 123, 132
Reducibility, 119, 121, 127, 129–132, 135, 140–142, 147
 C-reducibility, 122–124, 132, 134, 136
 D-reducibility, 122–127, 132, 134, 136
Reduction obstacles, 130–132
Relevance logic, 286, 313
Reliability growth modeling, 42–44
Repair, 302–303
Resolution, 15, 19, 77–88, 237, 262–264, 269, 273–274, 276, 286, 324
Restrictiveness, 189
Review of programs and designs, 45–47, 303–307
Rice, John R., 38–39
Rigorous-argument proof, 12–14, 17, 20, 202–204, 206–209, 218, 289, 310, 316–328
Ritchie, Graeme, 97–97
Robbins problem, 5, 89, 289–290
Robertson, Neil, 142, 148–149
Robinson, Abraham, 72–73, 203
Robinson, Alan, 15, 19, 77–81, 88, 276, 286, 288–289, 310, 325
Robinson, George, 88

Robotics, 5, 86, 98, 262
Rochester, Nathaniel, 69–70
Rome intrusion, 151–152
Ross, Douglas, 34, 36
Rosser, J. Barkley, 93, 199, 203, 206, 274
RSRE (Royal Signals and Radar Establishment), 245–251, 255
Rushby, John, 212, 271–272, 290, 293–296, 307, 310, 324–325
Russell, Bertrand, 65, 91, 258–259
 Principia Mathematica, 65–67, 71, 74, 264, 307–308, 329
 Russell's paradox, 264–265

SAGE (Semiautomatic Ground Environment), 29–31, 41–42, 57, 155, 162
SAM's lemma, 89–90
Sanders, Al, 212
Satisfiability, 76–77
Schaefer, Marvin, 182
Schell, Roger, 160–163, 166, 174, 182, 185
Schwartz, Richard, 230, 232, 235, 269–271
Science wars, 6–7, 94
SCOMP (Secure Communications Processor), 179, 190–191, 193–194
Scott, Dana, 186, 203, 274–275
"Scruffies," 82–83, 96, 264, 266
Security kernel, 16, 158, 161–163, 166, 168–170, 172, 178, 191–192, 195
Semantics, formal, 47, 51–54, 199, 274–275, 278
Semiautomatic Ground Environment (SAGE), 29–31, 41–42, 57, 155, 162
Shankar, Natarajan, 268, 271, 296
Shanker, Stuart, 143–144
Shapin, Steven, 9–11
Shaw, Cliff, 65–66, 73–74, 275, 307. *See also* Logic Theory Machine
Shimamoto, Yoshio, 124–127, 129–130
Shostak, Robert, 229, 232–233, 235, 269–271
SIFT (Software Implemented Fault Tolerance), 18–19, 223–236, 238–241, 243, 267, 269–270, 272, 306
Simon, Herbert, 14–15, 63–71, 73–75, 87–90, 94, 97, 264, 268, 275, 307–308
Simple security property, 164–165, 171. *See also* "No read up" rule
Skolem, Thoralf, 264–265
Slaney, John, 147–148
Smith, Michael, 215, 217, 241, 244
Software crisis, 14, 34–36, 41, 55, 299
Software engineering, 3, 14, 34–41
Software Implemented Fault Tolerance. *See* SIFT (Software Implemented Fault Tolerance)
Specification, formal, 4, 170, 178, 180–181, 196, 248, 270, 283, 290
SRI International, 168–169, 179, 223–225, 228–236, 238, 241, 245, 262, 267, 269–273, 288, 293, 295, 299, 314–315
*-property, 164–165, 170–171, 185
Stewart, Ian, 138, 148
Storage channel, 172
STP (Simple Theorem Prover/Shostak Theorem Prover), 270–271
Strachey, Christopher, 37, 273–275
Stromquist, Walter, 130, 133
Structured programming, 37–41, 56, 58, 141, 302
Sutherland, David, 186
Swart, Edward, 127, 130, 132–133, 139, 147
System Development Corporation, 30, 42, 155, 163, 169–170, 179–181, 288
System Z, 17, 183–184, 187

Tanenbaum, Andrew S., 205, 278
Tar pit, 32, 42, 56
Tarski, Alfred, 65, 309
Tautology, 71, 84–85
Teller, Paul, 143

Temporal logic, 251–252, 278, 285–286, 329
Testing, 2–3, 14, 42–46, 140, 205
 and cleanroom, 57–59
 critique by Dijkstra, 3, 45, 197
 and error guessing, 42
 exhaustive, 3, 46, 246
 mutation, 204–205
 and oracle problem, 44–45
 random, 42–44, 46
 and reliability growth modeling, 42–44
 sociotechnical effectiveness of, 304–306
Thomas, Martyn, 251, 324
Thompson, Ken, 299–300
Time-sharing, 16, 153–154, 156, 191, 195, 275
Timing channel, 172
Trading zone, 328–330
Trojan horse programs, 164–165, 171, 299
Trough, certainty, 333–334
Trust, interpersonal, 2, 10–12, 296, 316
Trust, social, 9–12, 19–20, 296–299, 316, 334
Trusted Computer System Evaluation Criteria. *See* Orange Book
Trusted computing base, 171
Trusted subjects, 16, 166, 170–172
Truth tables, 70–71, 73, 75, 84, 252–253
Turing, Alan, 47, 73, 75, 274–275
Turkle, Sherry, 12, 94, 330
Tutte, W. T., 101, 127, 137–138
Tymoczko, Thomas, 142–144, 320–321
Types, 265, 276–277, 279, 284–286
 Type-checking, 19–20, 276–277, 279, 293, 295–296

UCLA data secure Unix, 169–170
Unavoidable set of configurations, 117, 119–122, 127, 129–132, 134–136, 140, 142
Unification, 77–79, 276

Variable capture, 287, 294
VAX security kernel, 162, 191–192, 194
Verification condition generator, 61, 167
Verification, formal. *See* Formal verification
VIPER (Verifiable Integrated Processor for Enhanced Reliability), 19, 217, 245–251, 255, 262, 279, 324, 334
von Bertalanffy, Ludwig, 16, 163
von Henke, Friedrich, 234, 272, 307, 324
von Neumann, John, 47, 64, 91, 155

Waldinger, Richard, 283
Walker, Stephen T., 175–177, 180, 190, 194, 296
Wang, Hao, 15, 71–75, 77, 84, 90–91, 96, 98
Ware, Willis, 155–160
 Ware task force, 158–160, 163
Watters, Aaron, 218
Weissman, Clark, 163
Wensley, John, 224, 236
Whitehead, Alfred North, 65, 91, 258–259
 Principia Mathematica, 65–67, 71, 74, 264, 307–308, 329
Whitney, Hassler, 126–127
Wiles, Andrew, 135, 141, 148, 322
Wilkes, Maurice, 25–26
Wilkins, Burleigh Taylor, 210–211
Witt, Bernard I., 57–59
Wittgenstein, Ludwig, 144, 296–297, 332
Wos, Larry, 88
Wynne, Brian, 9

Young, William D., 215, 217, 244, 272
Y2K problem, 1, 8, 300, 302

Zero-sum assumption, 6, 320