INDEX

Note: Page references followed by f, t, and b indicate an illustrated figure, table, and box respectively.

A
Absolute price oscillator (APO), 57
ACCEPT state, 16–18, 20–21
Adaptation in Natural and Artificial Systems (Holland), 230
ADX. See Average Directional Movement Index
ADXR, 29
AFL. See AmiBroker Function Language
Algorithm
   backtesting
      Excel, usage, 145
      Python, usage, 167
   CCI algorithm, trades (generation), 50f
codes. See EasyLanguage; TradeStation.
criteria, 1–2
defining, 1
development, 1
flowchart (FC), stochastic oscillator crossover
   (entry signal usage), 45f
genetic algorithms, identification, 228–230
mean reversion algorithm, equity curve, 108f
parameters
   comparison, 94t
   results, 245t
   sets, 96t–97t
pseudocode, example, 6
ramp-up data, insufficiency, 158f
RSI algorithm, performance, 38t
stochastic algorithm, performance, 47t
   trading algorithm, example, 7b–8b
Algorithmic traders, programmer ability, 2
Algorithmic trading, books (source), 335–336
Algo Testing Software, example, 3b
AmiBroker, 293
debugging, Exploration (usage), 295–298
Exploration, results, 296f
functions, 295
keywords, 293–294
loop programming, 139–140
   code, sample, 140f
Monte Carlo settings dialog, 262f
Monte Carlo simulation, 261–264
position sizing, futures mode, 298–299
walk-forward optimizer, usage, 107f
AmiBroker Function Language (AFL), 113
array programming, 120–129
Automatic Analysis dialog box, 138f
backtest, parameters (setting), 126f
broker integration (feature), 114
Check icon, 125f
code samples, 141–143
Code Wizard
   Add item, 134f
   button, 133f
   Edit Rule dropdown list, 135f
   window, 133f
data, 114
   algorithm, application, 126f
types, 115–116
AmiBroker Function Language (AFL) (Continued)
Editor, 114
   auto-completion tool, 124f
   function helper, 124f
   launching, 122f
   window, example, 123f
Edit Rule pane, parameters (change), 136f
   error warning, 125f
   expressions, 116–117
   results, 117–118
   features, 113–114
   initiation, 113–118
   integrated development environment (IDE), 114
   knowledge, programmer requirements, 115–118
   operators, 116–117
   precedence, 117–118
   parameters, setting, 126f
   portfolio results, 128
   power (feature), 114
   price (feature), 113
   request (sending), exclamation point (clicking), 137f
Send to Analysis window icon, 126f
Set Futures mode, 127f
simple moving average crossover, 120
speed (feature), 114
syntax, 129–133
3-D optimization chart, 130f
trade-by-trade report, 128f
usage, 227–230
variable naming, 116
variables, 115–116
   names, restrictions, 115
Wizard, 133–139
(AmiBroker keyword)
   flow control structures, 294–295
Analysis phase, 125
Analysis window, 129, 131
Annual return, performance metric, 91
APO. See Absolute price oscillator
Appels, Gerald, 53, 56b
ApplyStop function, 132
Arguments, usage, 161f
Arithmetic expressions, 116
ASCII
   database (futures prices), 121
   data importer, 114
ATR. See Average true range
Auto-generated formula, 135, 137
Average (VBA function), 160–161
Average Directional Index (ADX), 208
Average Directional Movement Index (ADX), 26–32
14-day ADX, calculation, 28–29
length, 28
   trend detector function, 29f
Averages, 24
Average true range (ATR), 11b, 272
   ATR Sell MOC, 35b, 46b, 49b, 52b, 56b
   BuyToCover MOC, 40b, 46b, 49b, 52b, 56b
   5X ATR profit objective, usage, 165–166
   lookback, parameter (change), 282
   multiplier, parameter (change), 281
   3X ATR stop, usage, 164, 165–166
   2 ATR, usage, 69
   usage, 70–71
Averaging methods, performances, 59, 62
   avgLen, default value, 124
   avgMP, 119
   AvgTrueRange, 208
B
Backtest
   equity curve, profit (display), 283f
   parameters, setting, 126f
   performance report, 279f
Backtest button, 129
Backtester settings, example, 262f
Backtesting
   Excel, usage, 145
   software structure, 149–154
   flowchart, 150f
Bandy, Howard, 114
Bars held, performance metric, 91
BASIC. See Beginner’s All-purpose Symbolic Instruction Code
BBandBot, 124
BRandTop, 124
Beginner’s All-purpose Symbolic Instruction Code (BASIC), 145, 154
Berkely, George, 243
Berlinski, David, 1
bin increment (binInc) variable, 267
binTuple, 267
Bollinger algorithm
   benchmark, 73
   optimization, 104t
Bollinger Bands, 62–68
   algorithm
      best parameter sets, 74t
      version 1 (performance), 65t
      version 2 (performance), 68t
      breakout, 93t
calculation
  2 standard deviations, usage, 63
  60 days standard deviation, usage, 63
data, Keltner Channel data (comparison), 71f
example, 90b
function, 196
Function Module, 319
optimization, 72f
routine, defining, 148f
60-day, two-SD Bollinger Band, example, 63f
subroutine, 160–161
system, example, 9b–10b
trend algorithm, 64
trend follower function, 63b–64b
BollingerBand subroutine, 147
Bollinger Benchmark, Keltner Challengers
(contrast), 74t
Bollinger, John, 62
Bollinger performance, three-dimensional contour chart, 98f
Boolean expression, 116
Boolean flags, usage, 15b
BOTTOM failure, 39
bottoming out, term (removal), 6
Building Reliable Trading Systems (Fitschen), 264
buy (AmiBroker keyword), 293
Buy-and-hold mentality, 26
BUY order, 121
buyprice (AmiBroker keyword), 293
buy (keyword), recognition, 4

C
CCI. See Commodity Channel Index
CFDT. See Cumulative frequency distribution table change (keyword), recognition, 4
Choppy Market Indicator (CMI), 243–244
Chromosomes
crossing, VBA Excel (usage), 238–240
dartboard
creation, VBA Excel (usage), 236
example, 236
dart-throwing blindfolded monkey (simulation),
VBA Excel (usage), 236–237
finding moms and pops, VBA Excel (usage), 238–240
fitness of chromosomes (testing), VBA Excel (usage), 232–233
genes allocation process, VBA Excel (usage), 240
genesis passage, 240f
initial population setup, VBA Excel (usage), 232
mutation, 240–243
VBA Excel, usage, 242–243
population, 231
probability of being hit by dart calculation, VBA Excel (usage), 235
replacement function, VBA Excel (usage), 237–238
reproduction, 238–240
selection, 233–238
slice size, 234
  fitness conversion, VBA Excel (usage), 235
class (object), 176
ClassicShell, 169
class rsiClass(object), 317–318
class stochClass(object), 316–317
Cluster analysis. See Walk-forward optimization
CMI. See Choppy Market Indicator
Code, arguments (presence), 161f
Code Wizard. See AmiBroker Function Language
Coincident indicator performance, 54t
Combination lock (workings), finite statement machine modeling, 17f
Comment text, usage, 157f
Commitment of Traders Report (COT), 111
commName, 183
Commodity Channel Index (CCI), 48–53
algorithm
coincident indicator performance, 54t
example, 221–222
performance, 51t
trades, generation, 50f
OB/OS system description, 49b–50b
profit objective, usage, 327
protective stop, usage, 327
system, usage, 53f
trend following system, 52b
p-code, 52b
20-day CCI, calculation, 48
Computer language, syntax (understanding), 2
Conditional execution, 294–295
Connors/Alvarez, time-based exit (usage), 327–328
Consecutive bar exact sequence paradigm, 12, 21
Consecutive bar sequence, 21–23
COT. See Commitment of Traders Report
cover (AmiBroker keyword), 293
Cover arrays, 121–122
coverprice (AmiBroker keyword), 293
Crossover rate, 247
Crude oil, 60-day, two-SD Bollinger Band (example), 63f
Cumulative daily equity/drawdown, plotting (example), 153f
Cumulative frequency distribution table (CFDT), 265

curBar (sAverage function parameter), 172

Curve fitting, 227
prevention, 247–249

Customize Ribbon, 154

D

Daily bars (pivot high points), strength differences (examples), 13

Dartboard
creation, VBA Excel (usage), 236
example, 236f

Dart-throwing blindfolded monkey (simulation),
VBA Excel (usage), 236–237

Data arrays (ESB), 301

Data arrays (PSB), 309

DataMaster worksheet, usage (example). See Excel System Backtester

DC. See Donchian Channels

Definiteness (algorithm criteria), 1

Dependencies, nesting, 221f

Developer tab option, example, 155f

Directional movement (DM), 26–29
algorithm, 31f
performance, 33t
concept, introduction, 26
examples, 27f
system description, 30b–31b

Directional Movement Index (DX), 29

DM. See Directional movement

DMA. See Double simple moving average
dnSum (variable), 177

Donchian algorithms, results, 103t, 105t

Donchian Channels (DC), 90b
breakout, 93t

Donchian lengths, Donchian Channel breakout (usage), 93t

Double simple moving average (DMA) algorithm, 95

Double simple moving average (DMA) crossover, 89b
algorithm, 92t
downward, term (removal), 6

Draw down. See Trade draw down

DX. See Directional Movement Index

E

EasyLanguage, 203, 204, 323
abbreviations, 212t
algorithm codes, 325–334
chart, strategy application, 213f
codes, usage, 333–334

comments, 209
Connors/Alvarez, time-based exit (usage), 327–328
editor, 206f
line number display, 219f
editor, usage, 208f
functions, 324–325
inputs, change (process), 215f
keywords, 212t, 324–325
mean reversion strategy (change), input variables (usage), 215f
profit objective, usage, 326–327
project
creation, 220f
strategy, addition, 220f
protective stop, usage, 326–327
remarks, 209
reversion analysis strategy, 214f
samples, 221–224
SKIP words, 209
strategy
adjustment process, 215f
algorithm, dependencies/functions (nesting), 221f
syntax, 209–221
Turtle, LTL filter/pyramiding/position sizing (usage), 328–332

Editor. See AmiBroker Function Language

line number display (EasyLanguage), 219f

Edit (keyword), recognition, 4

Effectiveness (algorithm criteria), 2

ELD files
importing, 323–324
reimporting, 273

EMA. See Exponential moving average

End Of Day (EOD)
data feeds, 114
historic database, 151

Entry logic, 7b

Entry rules, template (example), 9b–10b

EOD. See End Of Day
equityClass, 188, 310–311

Equity curve
chart, maximum drawdown (display), 280f
creation, 247
example, 107f
mean reversion algorithm, 108f
multiple randomized equity curves, chart, 263f
pattern system, 109f
profit, display, 283f
stops, addition, 288f

EquityStream worksheet, macro (launch), 153f
INDEX

ES. See Exhaustive search
ESB. See Excel System Backtester
Eurocurrency downturn, CCI system detection, 53f
Excel
  data, 148–149
  worksheet, example, 151f
  ribbon, customization, 155f
  usage, 145. See also Visual Basic for Applications.
  Visual Basic Editor, example, 156f
ExcelSystemBackTester, 152
Excel System Backtester (ESB), 145, 301
  components, 149, 151
  data arrays, 301
  DataMaster worksheet, usage (example), 152
  EquityStream worksheet, macro (launch), 153f
  functions, 302–308
  keywords, 302
  programming environment, 154–163
  Results worksheet, trades/performance metrics
    listing, 153f
  source code, 147
  subroutines, 302–308
  VBA code, 155–163
ExcelSystemTester, functions/subroutines, 148
exclude (AmiBroker keyword), 293
Exhaustive search (ES)
  algorithm, parameters (results), 245t
Exhaustive search (ES) optimization engines, 228
Exit
  procedure, 162
  rules, template (example), 10b–11b
Exit logic, 7b–8b
exitPos function, 180
Exploration
  results, 296f
  usage, 295–298
Exponential moving average (EMA), 58–62
Expressions
  AmiBroker Function Language (AFL), 116–117
  arithmetic expressions, 116
  logical expressions, 116
  results, 117–118
Finding moms and pops, VBA Excel (usage), 238–240
Finiteness (algorithm criteria), 1
Finite state machine (FSM), 16, 24
  code, example, 224–226
  diagram, 17–18
  modeling, 17f, 18f
Fitness conversion, VBA Excel (usage), 235
Fitness of chromosomes (testing), VBA Excel (usage), 232–233
Fitschen, Keith, 264
5X ATR profit objective, usage, 165–166
Fixed Fractional, 272
  formula, 283
  money management overlay, 281–284
  parameters, change, 281–282
Flowchart (FC), 21, 24
  example, 22f
  usage, example, 23f
Flow control structures, 294–295
  for loop, 139, 184, 186
  for-loop, 174
FORTRAN, 168
  14-day ADX, calculation, 28–29
  14-day RSI, calculation, 32, 34
  14-day stochastic (FAST), calculation, 42
  FSM. See Finite state machine
Functions
  definition, colon (usage), 172
  Functions, nesting, 221f
  Future leak, advantage, 148
  Futures mode option, 127
  Futures prices, ASCII database, 121
G
Garbage in garbage out (GIGO), 152
Generation, 231
  concepts, 230
Genes allocation process, VBA Excel (usage), 240
Genetic algorithms (GAs)
  concepts, 230
  crossover rate, 247
  curve fitting, prevention, 247–249
  generations, 246
  identification, 228–230
  mutation rate, 246
  population size, 246
  usage, 243–247
Genetic optimization (GO), 95, 227
getData(), 182
GetData (ESB component), 149
Graphical user interface (GUI), 168
**INDEX**

<table>
<thead>
<tr>
<th>Page</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>185</td>
<td>head (variable)</td>
</tr>
<tr>
<td>180–181</td>
<td>Helper functions (PSB)</td>
</tr>
<tr>
<td>228, 230</td>
<td>Hermawanto, Denny</td>
</tr>
<tr>
<td>318</td>
<td>Highest Function/Module</td>
</tr>
<tr>
<td>55f</td>
<td>Histogram pivot points, usage</td>
</tr>
<tr>
<td>253f</td>
<td>Hindsight, usage</td>
</tr>
<tr>
<td>5f</td>
<td>Histogram system. See Moving Average Convergence Divergence</td>
</tr>
<tr>
<td>230</td>
<td>Holland, John</td>
</tr>
<tr>
<td>69</td>
<td>How to Make Money in Commodities (Keltner)</td>
</tr>
<tr>
<td>310</td>
<td>IDE. See Integrated development environment identifiers (PSB)</td>
</tr>
<tr>
<td>181–182</td>
<td>Lists, initiation (PSB)</td>
</tr>
<tr>
<td>4</td>
<td>load (keyword), recognition</td>
</tr>
<tr>
<td>116</td>
<td>Logical expressions</td>
</tr>
<tr>
<td>4–5</td>
<td>Long division procedure, example</td>
</tr>
<tr>
<td>14–15</td>
<td>Iteration growth rate, parameters (example)</td>
</tr>
<tr>
<td>114</td>
<td>Integrated development environment (IDE)</td>
</tr>
<tr>
<td>250–251</td>
<td>In-sample (IS) time period</td>
</tr>
<tr>
<td>1–</td>
<td>input (algorithm criteria)</td>
</tr>
<tr>
<td>253</td>
<td>InSampleBarsPerRun</td>
</tr>
<tr>
<td>75t</td>
<td>Keltner Channel, algorithm, best parameter sets</td>
</tr>
<tr>
<td>74t</td>
<td>Keltner Challengers, Bollinger Benchmark (contrast)</td>
</tr>
<tr>
<td>262</td>
<td>Last trade was a loser filter (LTLF)</td>
</tr>
<tr>
<td>218</td>
<td>lookback (sAverage function parameter)</td>
</tr>
<tr>
<td>294</td>
<td>Loops</td>
</tr>
<tr>
<td>16</td>
<td>LONG position, establishment</td>
</tr>
<tr>
<td>171–179</td>
<td>Indicator.py</td>
</tr>
<tr>
<td>169–170</td>
<td>Instruction statement, variability</td>
</tr>
<tr>
<td>110b</td>
<td>LastBarOfMonth (LBM) pattern</td>
</tr>
<tr>
<td>79</td>
<td>Last trade was a loser filter (LTLF)</td>
</tr>
<tr>
<td>220</td>
<td>MeanReversion (name)</td>
</tr>
<tr>
<td>212t</td>
<td>Keywords (EasyLanguage)</td>
</tr>
<tr>
<td>69</td>
<td>Keltner, Chester</td>
</tr>
<tr>
<td>48</td>
<td>Lambert, Donald</td>
</tr>
<tr>
<td>42</td>
<td>Lane, George C.</td>
</tr>
<tr>
<td>110b</td>
<td>Keltner Channel, algorithm, best parameter sets</td>
</tr>
<tr>
<td>108f</td>
<td>Mean reversion (MR)</td>
</tr>
<tr>
<td>86</td>
<td>Kaufman, Perry</td>
</tr>
<tr>
<td>215f</td>
<td>Mean reversion (MR)</td>
</tr>
<tr>
<td>223</td>
<td>maxContractPoints</td>
</tr>
<tr>
<td>223</td>
<td>maxContractProfit</td>
</tr>
<tr>
<td>280f, 282f</td>
<td>Maximum drawdown, display</td>
</tr>
<tr>
<td>282</td>
<td>Maximum quantity, parameter (change)</td>
</tr>
<tr>
<td>109b</td>
<td>Mean reversion (MR)</td>
</tr>
<tr>
<td>228</td>
<td>Maximum quantity, parameter (change)</td>
</tr>
<tr>
<td>171</td>
<td>Markets To Test, opening</td>
</tr>
<tr>
<td>188</td>
<td>marketSystemClass</td>
</tr>
<tr>
<td>185</td>
<td>Market on Close (moc)</td>
</tr>
<tr>
<td>162</td>
<td>Market (mkt) order</td>
</tr>
<tr>
<td>171</td>
<td>Markets To Test, opening</td>
</tr>
<tr>
<td>153f</td>
<td>Macro, ESB launching</td>
</tr>
<tr>
<td>123</td>
<td>MarginDeposit</td>
</tr>
<tr>
<td>294</td>
<td>marketDataClass, objects</td>
</tr>
<tr>
<td>185</td>
<td>LTLF. See Last trade was a loser filter</td>
</tr>
<tr>
<td>218</td>
<td>lookback (sAverage function parameter)</td>
</tr>
<tr>
<td>294</td>
<td>Loops</td>
</tr>
<tr>
<td>16</td>
<td>LONG position, establishment</td>
</tr>
<tr>
<td>114</td>
<td>Integrated development environment (IDE)</td>
</tr>
<tr>
<td>16</td>
<td>LONG position, establishment</td>
</tr>
<tr>
<td>162</td>
<td>Market (mkt) order</td>
</tr>
<tr>
<td>162</td>
<td>Market (mkt) order</td>
</tr>
<tr>
<td>171</td>
<td>Markets To Test, opening</td>
</tr>
<tr>
<td>188</td>
<td>marketSystemClass</td>
</tr>
<tr>
<td>220</td>
<td>MeanReversion (name)</td>
</tr>
<tr>
<td>212t</td>
<td>Keywords (EasyLanguage)</td>
</tr>
</tbody>
</table>
Mini Russell day trading system, narrow range day (usage), 110f
mmStop defaults, 132
MOC. See Time-based market order
moc. See Market on Close
Money management, 271
   Fixed Fractional money management overlay, 281–284
   stop, exit procedure, 162
Money Management window, 281
   parameters, change, 283f
Monte Carlo analysis, 227, 258–264, 322
Monte Carlo settings, AmiBroker Monte Carlo settings dialog, 262f
Monte Carlo simulation
   AmiBroker Monte Carlo simulation, 261–264
   distribution statistics, 264t
   implementation, Python (usage), 259–261, 265–269
Monte Carlo source code, 320–322
Monte Carlo straw broom chart, 263f
Moving Average Convergence Divergence (MACD), 53–58
   calculation, 12-day/26-day moving averages (usage), 55
   histogram system, 56b–57b
Moving averages
   crossover algorithm, 88
   double simple moving average (DMA) crossover, 89b
      algorithm, 92t
   exponential moving averages, 58–62
      simple moving averages, 58–62
      single simple moving average (SMA) crossover, 89b
      algorithm, 91t
   triple simple moving average (TMA) crossover, 89b–90b
      algorithm, 92t
   20-day moving averages, examples, 59f
   200-day moving average, 5
   weighted moving averages, 58–62
MR. See Mean reversion
Multi-algorithm strategy (MAS), 108–111
Multiple randomized equity curves, chart, 263f
Mutation, 240–243
   concepts, 230
   occurrence, 240–241
   rate, 246
   VBA Excel, usage, 242–243
myAvg, 115
myCondition, variable (initiation), 210
myDayCount, 158
MyFirstPortfolio, 277f, 288
MyFirstStrategyGroup list, 277f
   selection, result, 277f
myProfitTarg, 158
MySecondAlgoWizard, 137
mySum, 115
myTickValue, 163
myTrueRange, 148
myValue, 130–131
N
Narrow range day, usage, 110f
Net profit, performance metric, 90
New Concepts In Technical Trading Systems (Wilder), 26
New Trading Systems and Methods (Kaufman), 86
Number of trades, performance metric, 91
numcolumns (AmiBroker keyword), 294
O
OB/OS system description. See Commodity Channel Index
offset (sAverage function parameter), 172
OOS. See Out-of-sample
Operators
   AmiBroker Function Language (AFL), 116–117
   precedence, 117–118
Optimization
   genetic optimization, 227
   parameter sets, 98t
   3-D optimization chart, 130f
Optimize button, 129
Orders, types, 161–162
Oscillators, 25, 26
   crossover, entry signal usage. See Stochastics.
Outlining, 206
Out-of-sample (OOS)
   in-sample (IS), ratio, 102
   number of bars, 253
   percentage, 256
   report, 257t
   testing, 251–252
   example, 252t
   time period, Donchian algorithm results, 103t, 105t
   timespan, 104–105
OutOfSampleBarsPerRun, 253
output (algorithm criteria), 1
P
Paintbar, 206
Paradigms, example, 12–23
Parameters
  Bollinger performance, three-dimensional contour chart, 98f
  change, 136f
  comparison, 94t
  sets, 96t–97t
  optimization usage, 98t
  robust parameter sets, presence, 247f
  setting, 126f
  stress, 248
Partner matching probability, 238
Pattern system, equity curve, 109f
  p-code, 36b, 46b
Percentage price oscillator (PPO), 57
Percent Risk, parameter (change), 281
Performance metrics, 90–91
  example, 153f
Peters, Tim, 170
Pivot bar, strength (basis), 13
Pivot high points, strength differences (examples), 13f
Pivot point entry algorithm, example, 19b
Pivot-point long-entry algorithm, FSM modeling, 18f
PM. See Portfolio Maestro
pointvalue (AmiBroker keyword), 294
Population, 231
  concepts, 230
  initial population setup, VBA Excel (usage), 232
  size, 246
Portfolio
  AFL results, 128f
  algorithm, usage (decision), 101–108
  analysis, 271, 284–290
  backtesting, preparation, 279
  composition, 100–108
  correlation matrix, portion, 100f
  equity curve, stops (addition), 288f
  parameters, 104t
  profit targets, examples, 286f, 287f
  sample, Turtle system 2 (example), 85t
  Settings, 285f
  setup, PSB (usage), 182–193
  stop loss
    appearance, 287f
    setting, 286f
    stop option, 287f
  trades, stops
  exclusion, 289t
  inclusion, 289t
portfolioClass, 187–188, 313–315
Portfolio Maestro (PM), 203, 271, 272–290
  backtest performance report, 279f
  equity curve chart, maximum drawdown (display), 280f
  fixed fractional money management overlay, 281–284
  launch window, 273f
  Manage Strategy Group button, indication, 278f
  Money Management window, parameters (change), 283f
  MyFirstPortfolio, 277f
  MyFirstStrategyGroup, selection, 277f
portfolio
  analysis, 284–290
  backtesting, preparation, 279f
  equity curve, stops (addition), 288f
  profit targets, examples, 286f, 287f
  stop loss, appearance, 287f
  stop option, 287f
Portfolio Settings, 285f
  position size, profit/drawdown (correlation), 284f
Returns and Equity tab, maximum drawdown (display), 282f
strategy group, 273–274
  list, 277f
  settings, adjustment, 278f
  strategy, addition, 274f
Strategy Group settings button, 278f
  symbol, addition, 275f
  Total P/L option, 281f
Trade Analysis window, trading statistics display, 280f
Turtle 55 strategy, appearance, 274f
UGTATS list, selection, 275f
UGTATS Market List, 274–281
UGTATS markets, symbols, 276f
Portfolio Profit Target, impact, 287
Portfolio sample, Turtle system 1
  last trade was a loser filter (LTLF)
    absence, 81t, 82t
    inclusion, examples, 83t, 85t
positionscore (AmiBroker keyword), 294
PositionSize, 123
positionsize (AmiBroker keyword), 294
Position size, profit/drawdown (correlation), 284f
Position sizing, 99–100
  futures mode (AmiBroker), 298–299
  usage, 328–332
PPO. See Percentage price oscillator
Predictive indicator, 25
Price bar interface, 118–120
reserved words, 118
Price-based indicators, 25, 58
prices (sAverage function parameter), 172
Probability of being hit by dart (calculation), VBA Excel (usage), 235
Problem solving, 230–231
Profit factor, performance metric, 91
Profit/loss, performance metric, 91
Profit objective, usage, 326–327
Programming environment, 154–163
Protective stop, usage, 326–327
PSB. See Python System Backtester
PSBBollinger, opening, 170
Pseudocode, 5–6
algorithm pseudocode, 6
conversion process, 12
Pyramiding, usage, 80, 328–332
Python
code, example, 6
dynamic typing, 168–169
installation, 169
interpreted language, 168
shell, presence, 168
usage, 167, 259–261, 265–269
reasons, 167–169
user’s group, presence, 169
Python Monte Carlo simulation, 261
Python Programming (Zelle), 168
Python System Backtester (PSB), 168, 309
Bollinger Bands Function Module, 319
classes, 310–315
class rsiClass(object):, 317–318
class stochClass(object):, 316–317
configuration, 182–193
data arrays, 309
data import, 182–193
equityClass, 310
helper functions, 180–181
Highest Function/Module, 318
identifiers, 310
IDLE, usage, 169–170
import section, 179–193
indicator classes/functions, 315–319
indicator.py, 171–179
installation, 169–170
keywords, 310, 320–322
lists, 309
initiation, 181–182
Lowest Function/Module, 319
marketDataClass, 311
Monte Carlo
analysis, 322
source code, 320–322
portfolioClass, 313–315
portfolio setup, 182–193
rsiClass, 174–178
sAverage, 172–174
function, parameters, 172
Simple Average Function/Module, 319
Start Trade Drawdown
section, 320–321
source code, 320–322
structure, 171–193
SystemTester.py, 178–179
tradeInfoClass, 311–313
usage, 193–202
variables, initiation, 181–182
Q
Quantitative Trading Systems (Bandy), 114
R
rampUp, 157
Ramp-up data, insufficiency, 158f
Random number generator (RNG), 231, 239
tool, 258–259
usage, 260
Range
calculation, 111b
duration, 231
Profit factor, performance metric, 91
Profit/loss, performance metric, 91
Profit objective, usage, 326–327
Programming environment, 154–163
Protective stop, usage, 326–327
PSB. See Python System Backtester
PSBBollinger, opening, 170
Pseudocode, 5–6
algorithm pseudocode, 6
conversion process, 12
Pyramiding, usage, 80, 328–332
Python
code, example, 6
dynamic typing, 168–169
installation, 169
interpreted language, 168
shell, presence, 168
usage, 167, 259–261, 265–269
reasons, 167–169
user’s group, presence, 169
Python Monte Carlo simulation, 261
Python Programming (Zelle), 168
Python System Backtester (PSB), 168, 309
Bollinger Bands Function Module, 319
classes, 310–315
class rsiClass(object):, 317–318
class stochClass(object):, 316–317
configuration, 182–193
data arrays, 309
data import, 182–193
equityClass, 310
helper functions, 180–181
Highest Function/Module, 318
identifiers, 310
IDLE, usage, 169–170
import section, 179–193
indicator classes/functions, 315–319
indicator.py, 171–179
installation, 169–170
keywords, 310, 320–322
lists, 309
initiation, 181–182
Lowest Function/Module, 319
INDEX

Reserved words, 118
Results worksheet. See Excel System Backtester
Reversal, algorithm exit procedure, 162
Reversion analysis strategy (EasyLanguage), 214f
Risk aversion, 64, 66
Risk of ruin (RoR)
calculation, 88
competitors, examples, 89b–90b
impact, 87–88
RNG. See Random number generator
Robust parameter sets, presence, 247f
RoR. See Risk of ruin
roundlotsize (AmiBroker keyword), 294
Round quantity, parameter (change), 282
RSI. See Relative Strength Index
rsiClass, 174–178
Ruin, risk
calculation, 88
impact, 87–88
Run menu, 199
run (keyword), recognition, 4
S
sAverage, 172–174
function, parameters, 172
save (keyword), recognition, 4
Selection (chromosomes), 233–238
Selection of the fittest, concepts, 230
sell (AmiBroker keyword), 293
Sell arrays, 120–121
SELL order, 121
sellprice (AmiBroker keyword), 293
Send To Analysis, 125
setSysMarkInfo, 190
Sharpe ratio, performance metric, 91
short (AmiBroker keyword), 293
Short entries, 9b–10b
shortprice (AmiBroker keyword), 293
short (keyword), recognition, 4
Simple Average Function/Module, 319
Simple moving average crossover
algorithm, 88
Simple moving average crossover system
AFL usage, 120
3X ATR stop, usage, 164
Simple moving averages, 58–62
Single simple moving average (SMA)
crossover, 89b
algorithm, example, 91t
SKIP words, 209
Slow %D, 42
Slow %K, 42
Slow stochastic system description, 46b
SMA. See Single simple moving average
Source code, initiation, 157f
Standard deviation (SD) Bollinger Band, 62
START state, 16–18, 20–21
Start Trade Drawdown, 264–269
analysis, 90–91
calculation, 265
source code, 320–322
Stochastic (STO), 25, 42–47
algorithm, performance, 47t
14-day stochastic (FAST), calculation, 42
oscillator crossover, entry signal usage, 45f
profit objective, usage, 326–327
protective stop, usage, 326–327
slow stochastic system
description, 46b
p-code, 46b
trading algorithm, flowchart (FC) diagram, 44f
usage, examples, 41f, 60t–61t
VBA function, 161
Stop loss, setting, 286f
Stop (stp) order, 161
Strategy Group
concepts, 273
creation, 281
Straw broom chart, 263f
Subroutines. See Visual Basic
returns, 161
values, 161
Symbol List, creation, 275
Syntax
AmiBroker Function Language (AFL), 129–133
EasyLanguage, 209–221
errors, list (TradeStation), 219f
Sys Drawdown, performance metric, 90–91
systemMarket, 191
systemMarketClass, 189
System TA2, WFO default settings (usage),
252–254
SystemTester module, 155
SystemTester.py, 178–179
T
TA. See Trading algorithm
TDE. See TradeStation Development Environment
10-day average, calculation, 111b
TestSystem (ESB component), 149, 151
3D Optimization chart, 129
3D Optimization chart, 130f
3X ATR stop, usage, 164
ticksize (AmiBroker keyword), 294
Time-based market order (MOC), exit procedure, 162
TMA. See Triple simple moving average
TOP failure swing, occurrence, 39
TotalBarsPerRun, 253
Total P/L option, 281f
Total transaction costs, performance metric, 91
TP. See Typical price
Trade Analysis
   tab, 279
   window, trading statistics display, 280f
Trade-by-trade report, 128f
Trade draw down, initiation, 264–269
tradeInfoClass, 311–313
Trades
   example (generation), RSI (usage), 35f
geneneration (example), stochastics (usage), 41f,
   60f–61f
   signals (initiation), histogram pivot points
   (usage), 55f
   skipping, 149
TradeStation, 113, 323
   algorithm codes, 325–334
   Connors/Alvarez, time-based exit (usage),
   327–328
dates/times, storage, 212
functions, 324–325
keywords, 324–325
long position exit instruction, 218
Portfolio Maestro (PM) launch window, 273f
profit objective, usage, 326–327
protective stop, usage, 326–327
syntax errors, list, 219f
Turtle, LTL filter/pyramiding/position sizing
   (usage), 328–332
usage, 227–230
TradeStation Development Environment (TDE),
   203, 204
   algorithm, 208f
code verification/recalculation, confirming
   message, 218f
   EasyLanguage editor, 206f
   initial screen, 205f
TradeStation IDE, 204–209
   home screen, 204f
   trading apps, 205f
   tradeTuple, 261, 266
Trading
   idea, pseudocode conversion process, 12
   statistics, display, 280f
   system development, genetic algorithms (usage),
   243–247
Trading algorithm 1 (TA1), EasyLanguage Code
   (usage), 333
Trading algorithm 2 (TA2)
   EasyLanguage code, usage, 333
   WFA report, 254t
Trading algorithm 3 (TA3)
   EasyLanguage code, usage, 333–334
   WFO report card TA3, 257t
Trading algorithms, 77
   describing/programming, 12
   example, 7b–8b
   flowchart (FC)
      example, 78f
      usage, example, 23
   programming, Boolean flags (usage), 15b
   RSI trading algorithm, flowchart (FC) diagram,
      37f
   stochastic trading algorithm, flowchart (FC)
      diagram, 44f
Trading Blox testing platform, 82, 84
Trend
   detector, function, 29f
   follower, 63b
   following system. See Commodity Channel Index.
   trading, 86–100
Trend follower function
   Bollinger Bands, 63b–64b
   Keltner Channel, 69b–70b
 Trigger, incorporation, 44
Triple simple moving average (TMA) crossover,
   89b–90b
   algorithm, 92t
True range (TR), 28
Turtle 55 strategy, appearance, 274f
Turtle, LTL filter/pyramiding/position sizing
   (usage), 328–332
usage, 227–230
Turtle system
   examples, 79b
   LTLF
      absence, examples, 81t, 82t
      inclusion, examples, 83t, 85t
   trading algorithm flowchart, example, 78f
12-day moving averages, usage, 55
20-day CCI, calculation, 48
20-day moving averages, examples, 59f
26-day moving averages, usage, 55
Typical price (TP), 48, 69
V
Variable bar liberal sequence paradigm, 12, 21
Variable bar sequence, 13–21
depiction, long entry signal description, 14f
Variables, 115–116
initiation (PSB), 181–182
names, restrictions, 115
Vars statement, 210
VBE. See Visual Basic Editor
Visual Basic Editor (VBE), 154
example, 156f
Visual Basic for Applications (VBA)
average, function, 160–161
code. See Excel System Backtester.
code, comment text (usage), 157f
comments, ending, 157f
error message, example, 158f
Excel, usage, 232–240
functions/subroutines, 147–148
RSI function, 161
source code, initiation, 157f
stochastic function, 161
usage, 113
Volatility
cycle, function, 84
usage, 64

W
Walk forward, 227
Walk-forward analysis (WFA), 250
forward roll, graphic representation, 251f
report, 254	
Walk-forward efficiency (WFE), 254
Walk-forward optimization (WFO), 106, 228
cluster analysis, 256–258
cluster results matrix, 256f
default settings, usage, 252–254
entries/exits code, 255
examples, 249–258
in-sample testing, 250–251
TA2, example, 251t
out-of-sample (OOS)
report, 257t
testing, 251–252
element, 252t
report card TA3, 257t
Walk-forward optimizer
usage, 107f
value, 249–258
Walk-forward performance, equity curve
(example), 107f
Walk-forward testing
backtest, usage, 106f
out-of-sample (OOS) time period, 105t
steps, 105
Weighted moving averages, 58–62
WFA. See Walk-forward analysis
WFE. See Walk-forward efficiency
WFO. See Walk-forward optimization
Wilder, J. Welles, 26, 30, 32, 35

Z
Zelle, John, 168