Cited Works of Interactive Fiction

The following bibliography includes only those works cited in the text of this book: it makes no claim to completeness or even balance. An index entry is followed by designer’s name, publisher or organisation (if any) and date of first substantial version. The following denote formats: ZM for Z-Machine, L9 for Level 9’s A-code, AGT for the Adventure Game Toolkit run-time, TADS for TADS run-time and SA for Scott Adams’s format. Games in each of these formats can be played on most modern computers. Scott Adams, “Quill”-written and Cambridge University games can all be mechanically translated to Inform and then recompiled as ZM. The symbol ♦ marks that the game can be downloaded from ftp.gmd.de, though for early games sometimes only in source code format. Sa1 and Sa2 indicate that a playable demonstration can be found on Infocom’s first or second sampler game, each of which is ♦. Most Infocom games are widely available in remarkably inexpensive packages marketed by Activision. The ‘Zork’ trilogy has often been freely downloadable from Activision web sites to promote the “Infocom” brand, as has ‘Zork: The Undiscovered Underground’.


‘Advent’, 2, 47, 48, 62, 75, 86, 95, 99, 102, 105, 113 ex8, 114, 121, 124, 126, 142, 146, 147, 151, 159, 159, 179, 220, 221, 243, 264, 312 ex125, 344, 370, 377, 385, 386, 390, 393, 394, 396, 398, 403, 404, 509 an125. Will Crowther (c. 1975) and Don Woods (1976). Ported to many formats, including L9, TADS and ZM. ♦


‘Alice Through the Looking-Glass’, 2, 86, 121, 130, 142, 210, 221. Gareth Rees, Inform tutorial example, from the opening scenes of the novel by Lewis Carroll. ZM ♦


528


`Cosmic Capers’, 366. See ‘HitchHiker-64’.


`Countdown to Doom’, 353. Peter Killworth, Acornsoft (1984) and subsequently Topologika.


`The Creation’, 144. Suggested but never-written Infocom game by Marc Blank.


`Dallas Quest’, 351. James Garon, Datasoft (1984), from the television soap opera.


'The Fellowship of the Ring', 380. U.S. title of Addison-Wesley re-release of 'Lord of the Rings Game One', *q.v.*


'Galatea', 362. Emily Short (2000). *ZM* ◊


'Hack', 515 *an*137. Traditional maze game written over many years by many hands, and with numerous incarnations, such as 'Moria' and 'Nethack'. In this index, *see also* 'Rogue'.


'Horror of Rylvania', 361. Dave Leary (1993). *TADS* ◊


'I-0', 117. Adam Cadre (1997). *ZM* ◊

'Infidel', 355, 389, 392, 402. Mike Berlyn and Patricia Fogleman, Infocom (1983). *ZM* *Sa1*


'Leather Goddesses of Phobos', 30, 354, 368, 369, 373, 379, 397. Steve Meretzky, Infocom (1986). *ZM* *Sa2*


'Leopold the Minstrel', 244. Jamie Murphy (1996). *ZM*


'List Property', 201. Example Inform code by Graham Nelson. ◊


‘Moonmist’, 114, 403. Stu Galley and Jim Lawrence, Infocom (1986). ZM


‘A Nasal Twinge’, 236 ex102, 497 an102, 497 an103. Example Inform code by Graham Nelson. ♦


'Roman', 384. Peter Killworth, example game in How To Write Adventure Games for the BBC Microcomputer Model B And Acorn Electron (Penguin Books, 1984).

'Ruins', see general index.


'A Scenic View', 111. Example Inform code by Richard Barnett. ◊


'She's Got a Thing For a Spring', 400. Brent VanFossen (1997). ZM ◊


'The Thief', 142, 205, 457 an42. Example Inform code by Gareth Rees. ◊

‘Zork’, 149, 179, 193, 249, 339, 379. Timothy Anderson, Marc Blank, Bruce Daniels, Dave Lebling and others, MIT (1977–9). A public-domain version called ‘Dungeon’ (a name later regretted) was circulated to universities world-wide in 1978 and much imitated. Divided into three parts for a commercial release 1980–2, with final fragments appearing in ‘Enchanter’ and ‘Sorcerer’. ♦
‘Zork I’, 142, 148 ex 42, 149, 188, 233, 326, 385, 389, 390, 393, 404, 405. Subtitle ‘The Great Underground Empire’. Marc Blank and Dave Lebling, Infocom (1980). ZM Sa1 Sa2 (and a further version, the ‘Mini-Zork I’ file, is also ♦)
‘Zork: The Undiscovered Underground’, 1. Mike Berlyn and Marc Blank, Activision under the Infocom label (1997), written to accompany the graphical game ‘Zork: Grand Inquisitor’. ZM
Index of Exercises

The first page reference given is to the exercise and the second to the solution.

1. mushroom picking, 80, 435.
2. Square Chamber, 81, 435.
3. “(first opening door)”, 90, 436.
4. “(first unlocking door)”, 90, 436.
5. before on second noun, 91, 437.
6. orange cloud, 110, 438.
7. diverted drops, 110, 438.
8. confused exits, 113, 438.
10. an “xyzzy” direction, 114, 439.
11. east becomes west, 114, 439.
13. two-way connections, 114, 440.
14. pair of silk gloves, 117, 440.
15. musical box, 119, 441.
16. acquisitive bag, 120, 442.
17. glass and steel boxes, 121, 442.
18. television set, 121, 442.
19. macramé bag, 121, 443.
20. plank bridge, 124, 444.
21. illusory door, 124, 444.
22. cage to open and enter, 128, 445.
23. car driving on roads, 129, 445.
24. pushing ball uphill, 130, 446.
25. Tyndale’s Bible, 132, 446.
26. bearded psychiatrist, 137, 448.
27. guidance computer, 139, 448.
28. removing conversation actions, 139, 448.
29. game of Simon Says, 140, 449.
31. Dyslexic Dan, 140, 450.
32. grammar for Dan only, 141, 451.
33. alarm clock, 141, 451.
34. tricorder, 141, 452.
35. replicator, 141, 452.
36. communications badge, 141, 453.
37. Zen flight computer, 142, 454.
38. Picard and Maharg, 142, 455.
39. Martha the telepath, 142, 455.
40. darkness, abolition of, 145, 456.
41. troll afraid of the dark, 146, 456.
42. thief who wanders, 148, 457.
43. weight monitoring, 148, 457.
44. an egg-timer, 148, 459.
45. sound of scuttling claws, 149, 460.
46. harder scuttling claws, 149, 460.
47. midnight, 150, 460.
48. nightfall and daybreak, 150, 461.
49. mid-air location, 150, 462.
50. long time-scale game, 150, 462.
51. player reacting before, 156, 462.
52. silencing player, 156, 462.
53. the player’s wayhel, 157, 462.
54. where there’s a wayhel..., 157, 463.
55. the hidden burial shaft, 157, 463.
56. Giant with conscience, 158, 465.
57. chessboard of rooms, 164, 465.
59. varying the prompt, 191, 466.
60. remove rules with library messages, 191, 466.
61. add rules with library messages, 191, 466.
62. printing pronouns, 194, 467.
63. chessboard of sixty-four locations, 196, 467.
64. inventory inside box, 197, 469.
65. very verbose mode, 200, 469.
66. double inventory, 203, 470.
67. Scrabble pieces, 204, 470.
68. denominations of coin, 205, 471.
69. I Ching coins, 205, 471.
70. alphabetically sorted objects, 205, 474.
71. tomato in red or green, 210, 475.
72. artiste formerly known as Princess, 210, 475.
73. drinks machine, 210, 476.
74. parsing names like the parser, 210, 478.
75. parsing adjectives, 210, 478.
76. referring to objects by number, 210, 478.
77. wild-card name, 210, 478.
78. multiple-wild-card, 210, 478.
79. the ‘Spellbreaker’ cubes, 213, 479.
80. cherubim plural, 214, 481.
81. genie muddling black and white, 215, 481.
82. printing dummy verb, 221, 482.
83. footnotes, 226, 482.
84. French numbers, 228, 483.
85. token resulting in multiple-object, 228, 483.
86. floating-point numbers, 228, 484.
87. phone numbers, 228, 485.
88. parsing times of day, 228, 485.
89. spaceship control panel, 228, 488.
90. parsing any quoted text, 229, 489.
91. third noun for parser, 229, 489.
92. parsing Inform numbers, 229, 490.
93. and constant names, 229, 490.
94. and character constants, 229, 491.
95. and attribute names, 229, 491.
96. and property names, 229, 492.
99. everything in scope, 234, 493.
100. room divided by glass window, 235, 493.
101. dwarf breathing in dark, 235, 495.
102. player’s nose, 236, 496.
103. sterilising machine, 236, 497.
104. red sticky label, 236, 498.
105. “lock” and “unlock” inferring keys, 239, 499.
106. faintly lit rooms, 240, 499.
107. dative case suffix, 255, 500.
108. pronominal adverb, 255, 501.
110. donne-lui l’oiseau, 268, 502.
111. dessus et dedans, 268, 502.
112. shifting tail of buffer, 268, 503.
113. German da + preposition, 268, 503.
115. Italian contractions, 271, 505.
116. article sets in French, 272, 505.
117. article sets in Italian, 272, 505.
118. Latin contractions, 272, 506.
119. French numbers, 273, 506.
120. printing the time, 273, 507.
121. French pronouns, 277, 507.
122. printf routine, 310, 507.
123. title page, 311, 508.
124. status line invisible, 311, 508.
125. . . . showing treasure, 312, 509.
126. . . . with compass rose, 312, 509.
127. . . . with centred room, 312, 510.
128. wavy margins, 314, 511.
129. press any key, 317, 511.
130. is a key held down?, 317, 511.
131. hurrying the player, 317, 511.
132. making the player’s mind up for her, 317, 511.
133. real-time hourglass, 317, 512.
134. mouse testing, 318, 513.
135. menu of common commands, 318, 514.
136. saving the character, 319, 515.
137. ghosts, 319, 515.
138. exception handler for actions, 320, 515.
See also the index of exercises, the index of works of interactive fiction (except for ‘Ruins’ which has its entry in this index), and the appendices and tables. For detailed definitions of library rules, see the entry here on ‘rules given in full’.

Statements, directives and names of things built-in to the Inform language are in typewriter font: thus for (statement), object (directive), random (built-in function). Typewriter is also used for identifier names defined in the library, such as score (library variable) and NextWord (library routine). For such names, the page reference in bold face is to the detailed description in the Appendix. Verbs are double-quoted: thus “inventory” (library verb). Library files are written thus: "linklpa.h" (library file), ending “.h” even though on some operating systems the filenames omit these endings. Grammatical structures used by the parser are in angle brackets: ⟨noun phrase⟩.

A page reference in the form “123 ex7” means exercise number 7 on page 123. Similarly, “298 an7” means “answer to exercise 7 on page 298”.

:: (superclass operator), 64.
\' (character constants), 10.
\' (dictionary words), 10.
" (double-quotiation mark), 7.
@ (for assembly opcodes), 306.
! (comment character), 6.
! (ICL comment character), 287.
* (marking a routine to watch), 104.
* (multiplication operator), 13.
+ (addition operator), 13.
+ (setting pathnames in ICL), 288.
++ (increment operator), 15.
, (join assignments operator), 26.
- (subtraction operator), 13.
- (unary minus operator), 14.
-- (decrement operator), 15.
--> (array operator), 40.
-> (byte array operator), 42.
-> (in grammar lines), 217.
-> (indicating object parentage), 51.
-> (store result of opcode), 306.
-D (Debugging switch), 95, 96.
-e (economy mode switch), 334.
-i (ignore Switches switch), 287.
-k (debugging info switch), 105.
-M (module switch), 281.
-s (statistics switch), 285.
-S (Strict mode switch), 6, 96, 100.
-U (use modules switch), 281.
-v (story file version switch), 305.
-X (Infix switch), 96.
-x (row of hashes switch), 285.
. (label marker), 36.
. (property operator), 56.
.# (property array size operator), 58.
.& (property array operator), 57.
/ (dividing prepositions), 223.
/ (division operator), 13.
//p (indicates plural word), 10, 213.
“,” (Infix verb), 101.
; (semicolons dividing statements), 7.
; (the mark of Infix verbs), 101.
“;” (Infix verb), 102.
“;examine” (Infix verb), 104.
“;give” (Infix verb), 103.
“;i” (Infix verb), 104.
“;inventory” (Infix verb), 104.
“;move” (Infix verb), 103.
“;remove” (Infix verb), 103.
“;w” (Infix verb), 103.
“;watch” (Infix verb), 103.
“;x” (Infix verb), 104.
< (less than? operator), 19.
<. . > (causing action), 88.
<<. . >> (action and return), 88.
<= (less than or equal? operator), 19.
= (set equals operator), 11.
== (equals? operator), 19.
> (greater than? operator), 19.
>= (greater or equal? operator), 19.
[ (directive), 38.
[ (open routine), 8.
\ (line folding character), 29.
\ (printing a literal backslash), 29.
^ (apostrophe, in dictionary word), 11.
^ (new-line, in string), 7.
^ (printing a literal hat), 29.
_ (permitted in identifiers), 12.
{ (open code block), 21.
| (bitwise OR operator), 16.
|| (condition “or” operator), 20.
} (close code block), 21.
^ (bitwise NOT operator), 16.
^ (printing a literal tilde), 29.
^ (unsetting ICL switches), 288.
^ (used to negate attributes), 60.
^ (within text, quotation mark), 28.
^= (not equals? operator), 19.
^^ (condition “not” operator), 20.
# (directive within routine), 279.
## (notation for action constants), 89.

$ (hexadecimal number), 9.
$? (ICL command), 288.
$$ (binary number), 9.
$huge$ (ICL command), 288.
$large$ (ICL command), 288.
$list$ (ICL command), 288.
$small$ (ICL command), 288.
% (remainder operator), 13.
& (bitwise AND operator), 16.
&& (condition “and” operator), 20.
] (close routine), 8.

1, spurious appearance of in printing, 17, 32.
"52.inf" (example program), 205.

@: (diaeresis accent), 29.
@ (printing a literal at-sign), 29.
@ (string escape character), 29.
@!! (inverted exclamation mark), 29.
@’ (acute accent), 29.
@00 (printing variable 0), 30.
@<< (continental quotation mark), 29.
@>> (continental quotation mark), 29.
@?? (inverted question mark), 29.
@\ (Scandinavian slash accent), 29.
@^ (circumflex accent), 29.
@{a9} (example Unicode symbol), 30.
@^ (tilde accent), 29.
@‘ (grave accent), 29.
@® (ZSCII-code escape sequence), 29.
@AE (ligature), 29.
@ae (ligature), 29.
@c (cedilla accent), 29.
@et (Icelandic eth), 29.
@LL (pounds sterling sign), 29.
@o (Scandinavian ring accent), 29.
@OE (ligature), 29.
@oe (ligature), 29.
@ss (German û), 29.
@th (Icelandic thorn), 29.

Aarseth, E. J., 1.
Abbreviate (directive), 291.
abbreviations, 334.
absent (library attribute), 408.
“abstract” (library verb), 101.
accented characters, 29.
accusative pronouns, 194 ex62, 263, 467 an62.
Achieved (library routine), 163, 419.
Acorn Risc PC 700, 285.
action (library variable), 89.
⟨action phrase⟩, 245.
action_to_be (library variable), 216.
"action.h" (library extension), 333.
ActionPrimitive (library routine), 515 an138.
actions, 87.
checking the sensibleness of, 94.
creation of, 92.
diversion of, 109.
fake, 91.
fake fake, 91.
groups 1, 2, 3 of, 89.
how the parser chooses, 216.
notation for writing as constants, 89.
of the five senses, 109.
running silently, 90.
sequence of processing, 93.
statements to cause, 88.
tracing all actions generated, 99.
“actions” (library verb), 95, 99.
Activision, 357.
actor (library variable), 89, 215.
acute accents, 29.
Adams, Douglas, 4, 346, 357, 376, 407.
Adams, Scott, 147, 200, 348, 351, 376, 399, 407.
adaptive hints, 327.
add_to_scope (library property), 121, 236, 412.
additive (in Property directive), 86.
additive properties, 86.
⟨address⟩ (built-in printing rule), 31.
adjectives, 210 ex75, 478 an75.
possessive and demonstrative, 248.
used as descriptors, 265.
Adventure Game Toolkit, 359.
Adventure International, 351.
Adventure Master, 358.
Adventure Writer, 358.
Æsop, 30.
affix, 252.
after (library property), 78, 412.
AfterLife (entry point routine), 159, 431, 515 an137.
AfterPrompt (entry point routine), 165, 431.
AfterRoutines (library routine), 93, 419.
“again” (library verb), 245.
⟨again-word⟩, 245, 262.
agglutination, 252.
agreement (linguistic), 252, 255 ex109.
AGT (adventure design system), 359.
AIFF file, 324.
air, 389.
Akhnaten (opera), 59.
alarm clock, 141 ex33, 451 an33.
Alcott, Louisa May, 215.
alias (keyword), 269.
Alice Through the Looking-Glass, 355.
“all”, 239.
⟨all-word⟩, 248, 262.
Alligator, son of 7 Macaw, 508 an123.
AllowPushDir (library routine), 129, 419.
alphabet table, 260.
"AltMenu.h" (library extension), 333.
Alumnus Football, 160.
ambiguous object names, 239.
American spellings, 258.
amulets, 188.
Amusing (entry point routine), 161, 431.
AMUSING_PROVIDED (library constant), 161.
General Index

 Anderson, Doug, 374.
 Anderson, Tim, 3, 127, 188, 305.
 Andersson, Torbjörn, 4, 243, 320,
 366.
 ANIMA_PE (library constant), 237.
 "animalib" (alternative library), 193.
 animals, 390.
 animate (library attribute), 134, 408.
 ##Answer (library action), 134.
 Apollo moon landings, 24.
 Arabic, 29, 30.
 @aread (assembly opcode), 316.
 arguments of functions, 16.
 Aristophanes, 339.
 arithmetic expressions, 12.
 arithmetic overflows, 14.
 Arnold, Toni, 243.
 Array (directive), 40.
 "array.h" (function library), 46.
 arrays, 40.
 as property values, 57.
 breaking the bounds, 42.
 maximum number of, 338.
 (article), 248, 265.
 article (library property), 195, 272,
 412.
 articles (library property), 272, 412.
 artiste formerly known as Princess, 210
 ex72, 475 an72.
 Ashmolean Museum, Oxford, 188.
 ##Ask (library action), 134.
 asking questions, 233.
 ASKSCOPE_PE (library constant), 237.
 "AskTellOrder.h" (library extension),
 143, 221.
 assembly language, 306.
 assignments, 15.
 associativity of operators, 14.
 Asturias, Miguel Angel, 167.
 Atkinson, Adam, 347.
 ##Attack (library action), 134.
 Attribute (directive), 59, 282.
 attributes:
 defined, 58.
 defined in library, 408.
 example, 78.
 maximum number of, 338.
 Auden, Wystan Hugh, 374.
 Austin, Pete, Mike and Nick, 351.
 Avon, Kerr, 139.
 babel fish dispenser, 382.
 background colour, 311.
 Baggett, David M., 159, 226 ex83,
 361, 384, 385.
 Baker, Mark, 360.
 banner, 96.
 Banner (library routine), 152, 419.
 Barnett, Richard, 111.
 Barshack, Leonard, 349.
 Bartle, Richard, 346.
 base (of numbers), 9.
 (basic noun phrase), 248.
 Bates, Bob, 357.
 Baum, Michael, 4.
 Bauman, Steve, 360.
 Baumann, Joachim, 243, 312 ex126.
 beach full of stones, 65.
 Bedquilt Cave, 344.
 Beerbohm, Max, 231.
 before (library property), 79, 190,
 413.
 BeforeParsing (entry point routine),
 215, 215 ex81, 431, 481 an81.
 Belew, Adrian, 352.
 Belgian, 258.
 Beowulf, 254.
 Berez, Joel, 304, 305.
 Berlyn, Mike, 1, 4, 355, 401.
 Bicycles! (expletive in Dogg), 266.
 Big Al, 374.
 binary numbers, 9.
 see also bitwise operators.
 Bishop, David, 367.
Bishop, Stephen, 75, 342.
bitwise operators, 16.
Blakemore, Cleveland M., 341.
Blake’s 7, 140, 454 an37.
Blank, Marc, 1, 3, 141 ex36, 305, 321, 340, 349, 378, 380, 401.
Blasius, Volker, 2, 361.
Blobby, Mr, 1.
Blofeld, Ernst Stavro, 137.
Blorb (format for attaching sounds and pictures to a story file), 321.
``blorple'' verb example, 92.
blurb files (function library), 321.
Blyton, Enid, 366.
Bond, James, 137.
Booth, Connie, 326.
Booth, Frank, 42.
bounds of arrays, 42.
box (statement), 33.
Bracey, Kevin, 3, 321, 325.
brackets (arithmetic), 14.
Bradbury, A. J., 363.
Bradbury, Ray, 351.
branch opcodes, 306.
"branch.h" (library extension), 333.
Braun, Martin, 111, 130.
brack (statement), 26.
Brengle, Tim, 346.
Brennan, Gary, 402.
Briggs, Amy, 357, 368, 376.
Broderbund, 356.
broken objectloops, 54.
Brooks, Nigel, 349.
Brown, Neil James, 143, 369.
Bruyère, Jean de la, 131, 158.
@buffer_mode (assembly opcode), 311.
built-in functions, 49.
Bunch, David, 356.
Burneko, Jesse, 236.
⟨but-word⟩, 246, 262.
Buxton, Max, 402.
Byron, George Gordon, Lord, 74.
byte array, 42.
Bywater, Michael, 389.
c_style (library variable), 204.
Cadre, Adam, 34, 46, 117, 333, 336.
call (built-in message), 69.
Callaci, Irene, 143, 221, 392, 406.
Calvino, Italo, 344.
"calyx_adjectives.h" (library extension), 210, 242.
Cambridge University, 3.
cant_go (library property), 112, 413.
Canterbury Tales, 348.
CANTSEE_PE (library constant), 237.
capacity and exhaustion, 386.
capacity (library property), 118, 156, 160, 413.
Carroll, Lewis, 355.
case fall-through, 24.
case sensitivity of dictionary, 207, 337.
cases:
  ablative, 253.
  accusative, 252.
  dative, 252, 253.
  definition, 252.
  genitive, 251.
  in Finnish, 253.
  nominative, 252, 277 ex121.
  vocative, 252.
Cassidy, Jeff, 392.
@catch (assembly opcode), 36, 319.
cedilla accents, 29.
centred status line, 312 ex127, 510 an127.
chains and ropes, 392.
ChangePlayer (library routine), 156, 419.
"changes" (library verb), 99.
changing library messages, 190.
changing short names, 196.
changing the player, 155.
(char) (built-in printing rule), 31.
character of text, 10.
characters legal in unquoted Inform source code, 294.
Charlotte's game, 140 ex29, 449 an29.
Chaucer, Geoffrey, 348.
child (built-in function), 50.
children (built-in function), 50.
ChooseObjects (entry point routine), 239, 431.
circumflex accents, 29.
Clarke, Amanda, 174.
Clarke, Arthur C., 351.
Class (built-in metaclass), 49, 65.
Class (directive), 60.
class (part of Object statement), 61.
classes:
and additive properties, 86.
defined, 48.
example of, 84.
inheritance rules, 86.
Clavell, James, 366.
clearing the screen, 311.
Cleese, John, 326.
Clinton, William J., 30.
clothing (library attribute), 116, 408.
"clothing.h" (library extension), 117.
Clover, Andrew, 143, 151, 230.
clues, 392.
"cmap.h" (library extension), 114.
Coconut of Quendor, 371.
code blocks, 21.
Coe, Michael D., 174.
Collatz, Lothan, 27.
colours, 311.
⟨command⟩, 245.
command line syntax, 286.
comments, 6.
common properties, 56, 70.
CommonAncestor (library routine), 419.
Compass (library object), 261.
compass rose, 312 ex126, 509 an126.
CompassDirection (library class), 113, 261.
compilation, 5.
compile (ICL command), 288.
compiler error, 290.
component parts of objects, 236.
CompuServe, 360.
Conan Doyle, Sir Arthur, 367, 368, 390.
"conan, put every sword into the box" (parsing example), 250.
concealed (library attribute), 108, 408.
conditional compilation, 278.
conditions, 18.
conjunction (in parsing), 246.
connection of possessive adjectives to pronouns, 265.
⟨connective⟩, 246.
constant, 10.
Constant (directive), 39.
consult_from (library variable), 131.
##Consult (library action), 131.
consult_words (library variable), 131.
container (library attribute), 118, 408.
Contention of Ajax and Ulysses, The, 87.
continents, 43.
continue (statement), 26.
contraction forms, 271.
"converse.h" (library extension), 333.
Coolidge, Calvin, 397.
copy (built-in message), 66.
copyright message (of a game), 75.
copyright on Inform, ii.
copyright symbol, 261.
Cornelson, David, 4, 236, 362.
Covington, Alan, 346.
create (built-in message), 66.
creating data structures, 38.
creating objects during play, 65.
creature (grammar token), 222.
Cree, Graeme, 366, 368.
Crimes Against Mimesis, 365.
Croft, Lara, 157 ex55.
Cross, Duncan, 4.
Crowther, Patricia, 343.
Crowther, Will, 3, 4, 75, 343, 393.
Cunniff, Ross, 346.
Cunningham, Graham, 350.
Cyr-Jones, Liz, 357, 379, 402.
Cyrillic, 29.
d_to (library property), 412.
daemon (library property), 147, 413.
daemons:
  maximum number active, 148.
  running order, 462 an49.
  starting and stopping, 147.
  tracing active daemons, 99.
Dague, Ricardo, 114.
Dahlgren, Glen, 357.
Daley, Nicholas, 114, 230.
darkness, 144.
  abolition of, 456 an40.
  affecting scope, 231.
  changing scope within, 235 ex101.
  moving through, 145.
  nightfall, 150 ex48, 461 an48.
  Prince of, 457 an42.
  special object, 145.
  when it occurs, 145.
DarkToDark (entry point routine), 145, 432.
Darrah, Bill, 356.
data structures, 38.
dative pronouns, 264.
dead ends class, 48.
deaf1ag (library variable), 79, 158.
deafness, a period of, 156.
death, 158.
  and resurrection, 159.
DEATH_MENTION_UNDO (library constant), 161.
DeathMessage (entry point routine), 158, 432.
DEBUG (built-in constant), 97.
debug mode, 96.
debugging, 96.
  Infix, 96.
  information file, 105.

relying to objects by number, 210 ex76, 478 an76.
  strict mode, 96.
  suite of verbs, 96.
decimal numbers, 9.
decimal places, 228 ex86.
decipherment puzzles, 392.
declared but not used, 8.
decrement, 15.
Default (directive), 279.
default (in switch statement), 23.
declared constants, 39.
  ⟨definite article⟩, 248.
deleting objects during play, 65.
  ⟨demanding number⟩, 248.
deMause, Neil, 406.
Denman, William, 352.
describe (library property), 126, 198, 413.
description (library property), 77, 156, 413.
  ⟨descriptor⟩, 248, 265.
Deserted Village, The, 334.
destroy (built-in message), 66.
diaereses, 29.
dialects, 258.
  see also languages.
Diaz, Jose Luiz, 243.
Dickinson, Emily, 466 an58.
dictionary:
  characters allowed in, 207.
  maximum size of, 337.
  resolution and case sensitivity, 337.
  tokenisation using, 317.
  untypeable words in, 208.
dictionary of Maya glyphs, 132.
DictionaryLookup (library routine), 255, 420.
directions, 113.
  direction objects, 113, 439 an9.
  direction properties, 113, 412.
  Maya, 114 ex9.
  port and starboard, 439 an9.
directives, 38.
"dirs.h" (library extension), 114.
"dirsmap.h" (library extension), 114.
disambiguation, 239 ex105, 499 an105.
Discarded Image, The, 106, 147.
Disch, Thomas M., 352, 378.
disjunction (in parsing), 246.
division, 13.
division by zero, 13.
do (loop statement), 24.
Doctor Who, 125.
Dogg, 266.
Doherty, Paul David, 4, 105, 243, 347, 363.
"domenu.h" (library extension), 333.
Don Juan, 74.
Donne, John, 82.
door_dir (library property), 122, 414.
door (library attribute), 122, 408.
door_to (library property), 122, 414.
doors, 122.
trapping movement through, 124.
two-way, 123.
"doors.h" (library extension), 124.
"doors2.h" (library extension), 124.
double spacing, 28.
Dowling, Patrick, 346.
@draw_picture (assembly opcode), 315.
DrawStatusLine, 509 an125.
Dungeons and Dragons, 343, 369.
"dunno.h" (library extension), 242.
Dutch, 269.
Dutton, Gevan, 320.
Dying Earth, 370.
e_to (library property), 412.
each_turn (library property), 149, 414.
EACHTURN_REASON (library constant), 235.
Early Modern English, 244.
earth, 389.
Earthsea, 370.
Easter eggs, 105.
edible (library attribute), 77, 409.
Edlbauer, Florian, 4.
Egyptian magical amulets, 188.
ELEMENTARY_TT (library constant), 228.
"elevator.inf" (example game), 111, 130.
Elliot, Thomas Stearnes, 166, 206.
Ellison, Elizabeth, 360.
else (in if statement), 21.
Elvish, 30.
em-space, 519.
embedded routines, 57.
##Empty (library action), 90.
encapsulation, setting up, 58.
enclitic, 252.
@encode_text (assembly opcode), 317.
end game, 379.
Endif (directive), 279.
"English.h" (library file), 243.
enterable (library attribute), 127, 409.
entries of an array, 40.
entry point routines:
catalogued, 431.
defined, 152.
@erase_picture (assembly opcode), 315.
@erase_window (assembly opcode), 311, 314.
Erdös, Paul, 27.
error messages, 8, 290.
fatal errors, 290.
memory allocation, 289, 291.
error numbers used by parser, 237.
escape sequences, 29.
et_flag (library variable), 432.
etype (library variable), 139, 237.
"examine" versus "read", 133.
Excellent Prismatic Spray, The, 370.
EXCEPT_PE (library constant), 237.
execution (of program), 7.
exotic ways the game can end, 158.
expressions, 12.
Extend (directive), 98, 218.
extensions of the library, 190.

Fake_action (directive), 91.
fake actions, 91.
fake fake actions, 91, 142.
false (built-in constant), 10.
familiar form (linguistic), 257.
fatal error, 290.
Fawlty Towers, 326.
Fawn Hoof, 342.
female (library attribute), 195, 269, 409.
filename translation, 286.
Finnish, 243, 252, 258.
fire, 388.
first (part of Extend directive), 219.
Firth, Roger, 4.
Fischer, Kathleen, 337, 362.
flag variables, 20.
"flags.h" (function library), 46, 336.
Flathead dynasty, 364.
Flaubert, Gustave, 5.
Fleming, Ian, 137.
floating-point numbers, 228 ex86, 484 an86.
fly in amber, 209.
Fogleman, Patricia, 355.
"follower.h" (library extension), 143.
font (statement), 34.
Footnote, The, 482 an83.
footnotes, 226 ex83, 482 an83.
for (loop statement), 26, 300.
Forchione, Kevin, 230.
foreground colour, 311.
Forman, C. E., 381, 392, 395.
formatted text, 311.
found_in (library property), 108, 414.
Freebern, Ryan, 4.
fried green tomato, 206.
Fry, Stephen, 71.
ftp.gmd.de (archive), 2.
function arguments:
defined, 16.
maximum number of, 338.
functions, 16.
functions, built-in, 49.
“g” (library verb), 245.
Gabriel, Peter, 441 an15.
Galley, Stu, 341, 356, 366, 376.
GamePostRoutine (entry point), 94, 432.
GamePreRoutine (entry point), 93, 432.
Gauß, Carl, 30.
Gazzard, Sue, 379.
Genatossio, Carl, 355.
gender in parsing, 195.
gender-number-animation (GNA), 247, 269.
general (library attribute), 409.
general parsing routines, 226.
generating random numbers, 34.
Generic Adventure Games System, 359.
Genesis, 441 an15.
German, 243, 247, 252, 253, 257, 264, 267, 268 ex113, 269, 270, 503 an113.
’sz’ ligature, 29.
@get_cursor (assembly opcode), 314.
@get_wind_prop (assembly opcode), 313.
GetGNAOfObject (library routine), 270, 420.
Gilbert, W. S., 201.
Giner-Sorolla, Roger, 365, 382, 387, 391.
##Give (library action), 134.
give (statement), 59, 301.
Global (directive), 40.
global variables, 39.
  maximum number of, 338.
glue, 15.
glulx, 334.
glyphs, 131.
Goldsmith, Oliver, 334.
"gonear" (library verb), 101.
Gordon, Ken, 351.
Gorges, Arthur, 364.
Gotham City, 125.
"goto" (library verb), 101.
GPR_FAIL (library constant), 227.
GPR_MULTIPLE (library constant), 227.
GPR_NUMBER (library constant), 227.
GPR_PREPOSITION (library constant), 227.
GPR_REPARSE (library constant), 227.
Grafton, Anthony, 482 an83.
grain (library constant), 220.
grain (library function), 220.
grain (library property), 220.
grain (library routine), 220.
Grammar (library constant), 220.
Grammar (library property), 220.
Grammar (library variable), 220.
Grammar (library function), 220.
Grammar (library routine), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
Grammar (library property), 220.
identifiers, 12.
if (statement), 19.
Ifdef (directive), 279.
Iffalse (directive), 279.
Ifndef (directive), 279.
Ifnot (directive), 279.
Iftrue (directive), 279.
Ikaheimonen, J. P., 243.
"imem.h" (library extension), 73, 337.
(imperative verb), 245.
Implementors, 350.
implicit taking, 224.
Import (directive), 282.
in (object in this? operator), 53.
in_to (library property), 412.
Include (directive), 278.
increment, 15.
indefinite article, 195.
(indefinite article), 248.
indentation (in source code), 22.
IndirectlyContains (library routine), 420.
indistinguishable object names, 212.
individual properties, 70.
INFIX (built-in constant), 97.
Infix (debugger), 96.
infix (in linguistics), 252.
inflection, 252.
"Info.h" (library extension), 236.
Infocom:
  advertising, 352.
  bugs in games, 404.
  business history, 356.
  dedication, 4.
  and Douglas Adams, 353, 357.
  founded, 349.
  and genre fiction, 354.
in German, 243.
Invisiclues, 326.
play-testers, 402, 406.
revival of cult of, 361.
unpublished emails, 340.
uses Inform, 1.
  winding-down, 350, 357.
  ZIL design language, 304.
Inform Command Language, see ICL.
Informese, 244.
InformLibrary (library object), 76.
"infotake.h" (library extension), 193.
inglish, 352.
inheritance, 60.
initial (library property), 77, 155, 415.
initial possessions, 152.
Initialise (entry point routine), 75, 152, 432.
inp1 (library variable), 89.
inp2 (library variable), 89.
@input_stream (assembly opcode), 309.
input streams, 309.
InScope (entry point routine), 235, 432, 456 an39.
inside_description (library property), 127, 415.
instances, 60.
internal text format, 317.
interpreter, 5.
interrupt routines:
  text line countdown, 314.
  timed during keyboard input, 316.
  when sound effect ends, 316.
invent (library property), 197, 415.
inventories, 197.
inventory_stage (library variable), 197, 204.
inversion (statement), 284.
Invisiclues, 161, 326.
Irving, Francis, 36.
is see-through, 146.
ISO Latin-1, 29.
IsSeeThrough (library routine), 146, 420.
"istring.h" (function library), 46.
It, 257.
Italian, 269, 271 ex115, 272 ex117, 505 an115, 505 an117.
ITGONE_PE (library constant), 237.
itobj (old library variable), 238.

J4, 393.
Jaeger, Jay, 349.
Jekyll and Hyde, 281.
John Prince, 350.
Johnson, Dale, 352.
Jokisch, Stefan, 3.
jump (statement), 36.
JUNKAFTER_PE (library constant), 237.

Kalus, Max, 124.
keep_silent (library variable), 90.
Kellum, Patrick, 320.
Kensington Park, 304.
Kentucky Cave Wars, 343.
Kernighan, Brian, 22.
key and door puzzles, 387.
Kidder, Tracy, 345.
Killworth, Peter, 362, 364, 384, 385.
Kinder, David, 2.
King Lear, 457 an42.
Kinyon, Michael, 379, 406.
##Kiss (library action), 134.
Klimas, Chris, 320, 333, 362.
Klingon, 30.
Klobe, Kirk, 3.
Known Space, 353.
Kronheimer, Peter, 142.

labels (in programs), 36, 306.
Lagarias, Jeffrey, 27.
Laird, John, 353.
language definition file, 243.
LanguageAnimateGender (constant), 271.
LanguageArticles (array), 272.
LanguageContraction (routine), 271.

LanguageContractionForms (constant), 271.
LanguageDescriptors (table), 265.
LanguageDirection (routine), 273.
LanguageGNAsToArticles (array), 272.
LanguageInanimateGender (constant), 271.
LanguageNumber (routine), 273.
LanguageNumbers (table), 266.
LanguagePronouns (table), 263.
LanguageTimeOfDay (routine), 273.
LanguageToInformese (routine), 266.
LanguageVerb (routine), 274.
LanguageVersion (constant), 259.
see also Compass and CompassDirection.
languages:
American English, 258.
Arabic, 30.
Belgian French (dialect of French), 258.
Dogg, 266.
Dutch, 269.
Early Modern English, 244.
Elvish, 30.
Finnish, 243, 252, 258.
German, 243, 247, 252, 253, 257, 264, 267, 268 ex113, 269, 270, 503 an113.
Hebrew, 244, 247.
Informese, 244.
Italian, 269, 271 ex115, 272 ex117, 505 an115, 505 an117.
Klingon, 30.
Latin, 246, 252, 253, 267, 269, 272 ex118, 506 an118.
Mayan, 114 ex9.
Nalian, 392.
Norwegian, 267, 269.
Norwegian, rural dialects of, 269.
of the Pacific rim, 30.
Old English, 254, 254.
Russian, 266.
Spanish, 243, 248, 252, 264, 267, 269.
Swiss French (dialect of French), 258.
Tagalog, 248.

Lanz, Volker, 143.
Laskey, Mike, 360.
Last Lousy Point, 403.
last (part of Extend directive), 219.
Latin, 246, 252, 253, 267, 269
ex118, 506 an118.
Leary, Dave, 361.
Legend Entertainment, 357.
LeGuin, Ursula K., 370.
Les plaisirs de la porte, 122.
##LetGo (library fake action), 119.
Letwin, Gordon, 349.
Level 9, 114 ex10, 351.
Lewis, C. S., 106, 147, 369, 376.
Lewis, Tony, 105.
Leyendas de Guatemala, 167.
library, 5.
library messages, 190.
catalogued, 426.
library routines, 419.
LibraryMessages (library object), 190.
life (library property), 134, 415.
ligatures, 29.
light, 144.
daylight, 150 ex48, 461 an48.
light (library attribute), 78, 145, 409.
light source puzzles, 386.

line break (white space), 6.
Link (directive), 281.
linking, 281.
"link1pa.h" (library file), 282.
"link1lv.h" (library file), 283.
"links.h" (library extension), 126.
list style, 201.
list_together (library property), 203, 415.
"Lister.h" (library extension), 205, 277.
listing_together (library variable), 204.
Little Women, 215.
Liz Cyr-Jones, 350.
1m_n (library variable), 190.
1m_o (library variable), 191.
local variables, 11.
Locale (library routine), 200, 420.
location (library variable), 75.
lockable (library attribute), 409.
locked (library attribute), 119, 409.
Long Count of Mayan time, 233.
long jump, 319.
Long, David, 349.
"longint.h" (function library), 36.
Look, 198.
“look inside”, 120.
lookmode (library variable), 469 an65.
LookRoutine (entry point routine), 199, 433.
LoopOverScope (library routine), 232, 421.
LOOPOVERSCOPE_REASON (library constant), 235.
loops, in object tree, 52.
Lovecraft, H. P., 354.
LTI_Insert (library routine), 268, 421.
luck, 393.
McCaffrey, Anne, 366.
McCarthy, C. A., 376.
McLean, Don, 342.
Mac OS, 286.
Macbeth, 152.
machinery puzzles, 387.
MacNeice, Louis, 435.
magic, 369.
Magnetic Scrolls, 351.
Maharg, Noslen, 142 ex38.
Main (compulsory routine), 7.
@make_menu (assembly opcode), 318.
"makemaze.inf" (example game), 114.
making grammar, 92.
Mai (library attribute), 195, 269, 409.
Malmberg, David R., 359, 360.
Mammoth Cave, 75, 342.
"manual.h" (library extension), 333.
map connections, 112.
Master Game, 379.
master-list of commands, 97.
Mataga, William, 355, 356.
MAX_CARRIED (library constant), 160.
MAX_SCORE (library constant), 162.
MAX_TIMERS (library constant), 148.
Maya directions, 439 an9.
Mayan dictionary, 132.
<br>mercy, 385.
(me-word), 249, 262.
"megalook" verb, 232 ex98, 493 an98.
Meier, Stefan, 363.
Melbourne House, 249, 351, 380.
Melchionna, Luca, 344.
members, 60.
memory, 288.
compiler settings, 288.
maximum size of game, 334.
needed by compiler, 288.
small, large or huge, 288.
memory settings, 286.
Menick, Jim, 363.
Menu (library class), 327.
menu of text options, 326.
"Menus.h" (optional library file), 326.
Mercant of Venice, The, 367.
Meretzky, Steve, 357, 370, 379, 395, 397.
Mrical, Joe, 193.
Merry Wives of Windsor, The, 367.
Message (directive), 280.
messages, 62.
macaw example, 84.
maximum parameters, 338.
to routines, strings and classes, 69.
tracing all messages sent, 99.
"messages" (library verb), 62, 99.
Mestel, Jonathan, 364, 396.
‘meta’ actions, 90.
meta (part of Verb directive), 217.
metaclass (built-in function), 49.
Metaclass (imaginary directive), 72.
metaclass, definition of, 49.
Micklus, Lance, 358.
Micro Adventurer, 350.
mid-air location, 150 ex49.
middle game, 377.
midnight, 150 ex47.
Mikado, The, 201.
Milton, Marcus, 358.
MIT Dynamic Modelling Group, 350.
Mitchell, Philip, 380.
MMULTI_PE (library constant), 237.
MOD file, 324.
modular extensions, 190.
modules, 281.
"money.h" (library extension), 193.
Montaigne, Michel de, 134.
Montfort, Nick, 4, 362.
Monty Python’s Flying Circus, 118.
mood mazes, 391.
mood rainbow, 364.
Moriarty, Brian, 349, 354, 371, 376, 394, 402.
Moskowitz, Denis, 117.
mouse clicks, 520.
@mouse_window (assembly opcode), 318.
move (statement), 51.
move_window (assembly opcode), 313.
"MoveClass.h" (library extension), 143.
moved (library attribute), 409.
MoveFloatingObjects (library routine), 109, 421.
"movie.h" (library extension), 193.
moving all objects in one place to another, 54.
MULTI_PE (library constant), 237.
multiexcept (grammar token), 222.
multiheld (grammar token), 222.
Murie, Andrew C., 236.
Murphy, Jamie, 244.
n_to (library property), 412.
nagual, 157 ex53.
(name), 248.
name (library property), 106, 206, 248, 416.
(name) (built-in printing rule), 31.
names per object (limit), 338.
Naming of Cats, The, 206.
Nardinocchi, Ilario, 243.
narrator, 368.
Nation, Terry, 140.
ne_to (library property), 412.
Nelson, Admiral Lord Horatio, 445 an23.
Nelson, Graham, ii, 4, 359, 361.
Nelson, Toby, 4.
neuter (library attribute), 195, 269, 409.
neutral form (German), 257.
new_line (statement), 33.
Newell, Bob, 4, 243.
NewRoom (entry point routine), 155, 433.
Newton, Isaac, 67.
NextWord (library routine), 207, 421.
NextWordStopped (library routine), 227, 421.
Nilsson, Thomas, 359, 364, 372.
Niven, Larry, 353.
NO_PLACES (library constant), 164.
(no-word), 262.
nominative pronouns, 277 ex121, 467 an62.
Norwegian, 267, 269.
rural dialects of, 269.
nose, 236 ex102, 497 an102.
NOTHELD_PE (library constant), 237.
nothing (built-in constant), 49.
NOTHING_PE (library constant), 237.
notify_mode (library variable), 163.
"notify" verb, 163.
notin (object not in this? operator), 53.
##NotUnderstood (library fake action), 138.
(noun), 248.
noun (library variable), 89, 225.
(noun phrase), 246.
noun phrases:
  animation, 247.
  definite and indefinite, 247.
  gender, 247.
  GNA, 247.
  number, 247.
  quantity, 247.
NounDomain (library routine), 421.
"NPCEngine" (library extension), 143.
(number), 246.
number base, 9.
number (library property), 416.
NUMBER_PE (library constant), 237.
NUMBER_TASKS (library constant), 162.
(number) (grammar token), 222.
numbers, 9.
w_to (library property), 412.
object 31, 210 ex76.
Object (built-in metaclass), 49.
Object (directive), 50.
OBJECT_SCORE (library constant), 162.
object tree, 49.
    movements in, 52.
    printing out, 98.
    really a forest, 52.
    setting up initial state, 50.
ObjectIsUntouchable (library routine), 95, 232, 422.
objectloop broken error, 54.
objectloop (statement), 53.
objects, 47, 76.
    child, sibling and parent of, 50.
    creation and deletion in play, 65.
    definite articles of, 195.
    duplicate and plural, 212.
    encapsulation of, 58.
    giving attributes to, 59.
    giving properties to, 55.
    grouping of in lists, 203.
    indefinite articles of, 195.
    inheritance from classes, 60.
    inventory entries of, 197.
    listed in room descriptions, 199.
    maximum number of, 338.
    maximum number of names, 338.
    moving around the tree, 52.
    philosophy of, 71.
    printing lists of, 201.
    referred to by number, 210 ex76, 478 an76.
    removing from tree, 52.
    sending messages to, 62.
    short names of, 194.
    tracing movements of, 99.
    with indistinguishable names, 212.
    see also object tree.
"objects" verb, 164.
"objlstr.h" (library extension), 105.
obsolete usages, 303.
(of-word), 262.
ofclass (object of class? operator), 48.
offers light, 146.
OffersLight (library routine), 146, 422.
Old English, 254, 254.
Olsson, Magnus, 320.
on (library attribute), 125, 409.
"on", "at" or "in", 227.
O’Neill, Jeff, 243, 357.
only (part of Extend directive), 219.
(oops-word), 245, 262.
opcodes, 306.
open (library attribute), 118, 409.
openable (library attribute), 119, 410.
oparadigm, 13, 306.
operator precedence, 14.
operators, 11, 12.
    full table of, 517.
Option (library class), 327.
optional puzzles, 394.
or (keyword used in conditions), 19.
##Order (library action), 135.
orders (library property), 138, 155, 416.
(other-word), 248, 262.
OuLiPo (Ouvroir de Littérature Potentielle), 344.
out_to (library property), 412.
"out" verb, 129.
"OutOfRange.h" (library extension), 193.
@output_stream (assembly opcode), 309.
output streams, 309.
overflows, arithmetic, 14.
overture, 371.
Owen, Mark, 3.
Pärt, Arvo, 33.
Paavola, Olli J., 347.
packing case, 118.
page break (white space), 6.
pairs of verbs to separate, 219.
palindromic Finnish word for soap dealer, 243.
Palladas of Alexandria, 278.
Parallelloipided, Boris J., 280.
parent (built-in function), 50.
parentheses, 14.
Parker, Dorothy, 165, 257.
Parker, Marnie, 105, 114, 193, 362.
parse_name (library property), 209, 213, 417.
parsed_number (library variable), 227.
ParseNoun (entry point routine), 209, 433.
ParseNumber (entry point routine), 226, 433.
parser:
   error numbers, 237.
   general introduction, 207.
level 0 (word stream), 207.
level 1 (grammar lines), 215.
level 2 (individual tokens), 222.
level 3 (scope of noun phrases), 231.
level 4 (noun ambiguities), 240.
levels, tracing activity of, 100.
parser_action (library variable), 213.
parser grammar extensions:
   adjectives, 249.
adverbs, 249.
demonstrative pronouns, 251.
genitive case, 251.
interrogative pronouns, 251.
possessive pronouns, 251.
pronominal adverbs, 268 ex111, 502 an111.
parser_one (library variable), 204, 213.
parser_two (library variable), 204, 213.
"Parser.h" (library file), 75.
ParserError (entry point routine), 237, 433.
"parserm.h" (library file), 282.
ParseToken (library routine), 227, 422, 446 an25.
PARSING_REASON (library constant), 235.
path variables, 286.
Paul, James, 356.
Pelak, Robert, 4.
Penney, Jason, 320, 322.
people, 390.
Perkins, Geoffrey, 357.
perlBlorb, 321.
persona of player, 155.
Petersen, Dale, 344.
philosophy of objects, 71.
Phoenix, 3, 347.
"phototalk.inf" (example program), 333.
Picard, Captain Jean-Luc, 142 ex38.
@picture_data (assembly opcode), 315.
@picture_table (assembly opcode), 315.
Pinsky, Robert, 352, 355.
Pintar, Judith, 360.
Pittsburgh, Joel Berez’s mother’s coffee table in, 304.
pixels, 312.
PlaceInScope (library routine), 234, 422.
“places” verb, 164.
plagiarism, 160.
plants, 390.
Plato, 147.
Platt, David, 349.
Player’s Bill of Rights, 363.
player’s origin, 152.
Players (software company), 349.
PlayerTo (library routine), 154, 422.
plural (library property), 212, 417.
plural markings on words, 214.
plural objects, 212.

##PluralFound (library fake action), 214.
pluralname (library attribute), 195, 269, 410.

PNG-format image, 324.
Pohl, Frederik, 366.
Ponge, Francis, 122.

Popol Vuh, 508 an 123.
⟨possessive adjective⟩, 248, 265.
Pratchett, Terry, 366.

precedence of class inheritance, 86.
precedence of operators, 14.
prefix, 252.
⟨preposition⟩, 246.
prepositions, 223.
Prince, John, 357, 404.
print (built-in message), 69.
@print_form (assembly opcode), 310.
print_ret (statement), 31.
print (statement), 31.
print_to_array (built-in message), 70.

printing a string array, 43.
printing hexadecimal numbers, 32.
printing output, 28.
printing rules, 31.
printing-variables, 30.
PrintRank (entry point routine), 163, 433.
PrintTaskName (entry point routine), 162, 433.
"printtime.h" (library extension), 151.
PrintVerb (entry point routine), 434, 482 an 82.

Prisoner, The, 128.
private (part of Object statement), 58.
Procyon, 109.

Professional Adventure Writer, 358.
programming errors, 13.
prologue, 376.
prompt, 165, 191 ex 59, 466 an 59.

##Prompt (library fake action), 191.
pronomin adverb, 255 ex 108, 268 ex 113.
⟨pronoun⟩, 249, 263.
PronounNotice (library routine), 238, 423.
pronouns, 238, 467 an 62.
accusative, 263.
dative, 264.
demonstrative, 251.
interrogative, 251.
nominative, 277 ex 121.
possessive, 251.
“pronouns” (library verb), 264.
PronounValue (library routine), 238, 423.

proper (library attribute), 195, 410.
proper noun, 195.

properties:
additive, 86.
common vs individual, 70.
defined in library, 412.
definition of, 56.
encapsulation of, 58.
example, 78.
holding arrays, 57.
holding routines, 57, 83.
maximum number of common, 338.

Property (directive), 282.
protagonist, 368.
Proust, Marcel, 90 ex 3.

provides (object provides property? operator), 56, 58.
pseudo-randomness, 35.
punctuation in dictionary words, 208.
“purloin” (library verb), 100, 234 ex 99, 493 an 99.

PushDir_wireless, 129.
@put_wind_prop (assembly opcode), 313.
puzzles:
air and gases, 389.
animals, 390.
capacity and exhaustion, 386.
clues to, 392.
conversation, 391.
decipherment, 392.
earth and digging, 389.
fire, 388.
keys and doors, 387.
light source, 386.
luck and randomness, 393.
machinery, 387.
mazes, 385.
monsters, 390.
optional puzzles, 394.
people, 390.
plants, 390.
rewards for solving, 394.
riddles, 392.
ropes and chains, 392.
timed, 386.
utility objects, 387.
vehicles, 388.
water, 388.

Queneau, Raymond, 344.
questions, asking yes or no, 164.
questions, parsing the player’s, 233.
Quill, The, 358.
Quirie, Duncan, 358.
quit (statement), 36.
quotation marks (European), 29.
quotation marks beautiful, 165.
quotation text, 489 an90.

radix, 9.
Raleigh, Sir Walter, 364.
random (built-in function), 35, 295.
“random” (library verb), 97.
random numbers, 34.
Raszewski, L. Ross, 37, 46, 73, 95,
124, 193, 230, 236, 320, 333, 337.
Ravenous Bugblatter Beast of Traal, 366.

Ravipinto, Daniel, 380.
reached, statement which cannot be, 31.
react_after (library property), 94,
137, 417.
REACT_AFTER_REASON (library constant), 235.
react_before (library property), 93,
109, 137, 417, 436 an3.
REACT_BEFORE_REASON (library constant), 235.
@read_char (assembly opcode), 316.
@read_mouse (assembly opcode), 318.
read (statement), 44.
“read” versus “examine”, 133.
reading books, 131.
real_location (library variable), 145.
reasons for scope searching, 235.
receive_action (library variable),
120.
##Receive (library fake action), 119.
“recording” (library verb), 97.
recreate (built-in message), 66.
recursion, 18, 21.
limit to extent of, 338.
Redrup, Bob, 385.
Reed, Ken, 358.
Rees, Gareth, 2, 4, 78, 142, 143, 205,
Release (directive), 283.
remainder, 13.
remaining (built-in message), 66.
##Remove (library action), 89.
remove (statement), 52.
Rendezvous with Rama, 353.
Rennie, John, 389.
Replace (directive), 191, 280.
replace (part of Extend directive),
219.
replacing grammar, 219.
“replay” (library verb), 97.
resolution of dictionary, 207, 259, 337.
resolving ambiguity, 239.
@restore (assembly opcode), 319.
restore (statement), 36.
resurrection, 159.
return (statement), 18.
return values, 16.
reverse (part of Verb directive), 218.
rewards for solving puzzles, 394.
rfalse (statement), 18.
Rian, Júrund, 4.
Riccardi, Giovanni, 243.
Rice, Grantland, 160.
Richard Dyer, 350.
Richards, Matthew, 3.
Richie, Stuart, 352.
riddles, 392.
rings, 29.
Ritchie, Dennis, 22.
Roberts, Mike, 1, 359, 361, 367, 386, 388, 395.
Robertson, Evin, 46.
Rodgers, Steve, 358.
Romeo and Juliet, 28.
routines, 16.
room descriptions, 198.
ROOM_SCORE (library constant), 162.
“room”, etymology of, 76.
routines, 16.
marking to be watched, without Infix
(trace routines switch), 104.
maximum depth of recursion, 338.
simple example of, 79.
see also interrupt routines.
Royal Society For Putting Things On Top
Of Other Things, The, 118.
rtrue (statement), 18.
‘Ruins’, 74, 369.
DeathMessage entry point, 158.
Initialise entry point, 153.
PhotographSub action routine, 92.
PrintRank entry point, 163.
Treasure class, 84.
Antechamber, 171.
iron cage, 463 an55.
skeletons, 463 an55.
Burial Shaft, 463 an55.
ancient honeycomb, 85.
ending, 174.
Forest, 74.
month-old newspaper, 166.
mushroom, 77, 435 an1.
packing case, 118.
red-tailed macaw, 83.
speckled mushroom, 167.
stone-cut steps, 80, 83.
wet-plate camera, 92.
iron cage, 445 an22.
lattice diagram, 381.
low mist, 108.
Lower End of Canyon, 172.
horrifying chasm, 119, 158.
map of game, 168.
player’s initial possessions, 153.
Mayan dictionary, 131.
sketch map of Quintana Roo, 166.
sodium lamp, 125.
Pumice-Stone Ledge, 173.
incised bone, 173.
score of scuttling claws, 149 ex45.
Square Chamber, 81 ex2, 106.
carved inscriptions, 107.
shaft of sunlight, 107.
Stooped Corridor, 123.
pygmy statuette, 85.
stone door, 122.
title page, 508 an123.
transformed to wayhel, 157 ex53.
Upper End of Canyon, 172.
pumice-stone ball, 129.
warthog, 462 an53.
Wormcast, 113 ex8.
  dropping objects in, 110 ex7.
  glistening white eggsac, 169.
  passing through, 157 ex54.
Xibalbâ, 171.
  stela, 171.
rules given in full:
  are objects distinguishable?, 213.
  change of location, 154.
  describing objects in room descriptions, 199.
  end of turn sequence, 150.
  parsing text against a single object, 208.
  printing inventory lines, 197.
  printing room descriptions, 198.
  printing short names, 196.
  processing actions, 93.
  resolving ambiguous nouns, 240.
  what's normally in scope, 231.
  when there is light, 146.
  see also world model.
run-time crashes, 13.
running out of memory, 291.
Russian, 266.
Rutherford, Michael, 441 an15.
Ruz, Alberto, 167.

s_to (library property), 412.
SACK_OBJECT (library constant), 161.
saippuakauppias, 243.
@save (assembly opcode), 319.
save (statement), 36.
saving data, 319.
saving the character, 319 ex136, 515 an136.
scenery (library attribute), 107, 410.
SCENERY_PE (library constant), 237.
Schmidl, Gunther, 114, 347, 363.
Schmidt, Miron, 210, 363.
Schmuckal, Peter, 349.

Schulz, Manuel, 363.
scope:
  addition to, 121, 236.
  full rules of, 231.
  looping over, 232.
  testing, 232.
  “scope” (library verb), 98.
scope_reason (library variable), 235, 455 an38.
scope_stage (library variable), 234.
SCOPE_TT (library constant), 228.
ScopeWithin (library routine), 234, 423.
score (library variable), 161.
score notification, 163.
  “score” verb, 217.
scored (library attribute), 155, 162, 410.
scoring system, 402.
scoring systems, 161.
Scrabble pieces, 204 ex67, 470 an67.
@scroll_window (assembly opcode), 314.
scrolling screen, 165.
se_to (library property), 412.
Serial (directive), 283.
@set_colour (assembly opcode), 311, 313.
@set_cursor (assembly opcode), 311, 314.
@set_font (assembly opcode), 314.
@set_margins (assembly opcode), 314.
@set_window (assembly opcode), 310.
SetPronoun (library routine), 238, 423.
SetTime (library routine), 150, 423.
setup files using ICL, 287.
Shakespeare, William, 28, 152, 457
an42, 481 an80.
Shakespearean games, 367.
Shaw, George Bernard, 231.
Shelley, Percy Bysshe, 147.
Sheridan, Richard Brinsley, 188.
Shirley, James, 87.
short_name_indef (library property), 270, 418.
short_name (library property), 196, 417.
Short, Emily, 362.
Short, Nigel, 449 an28.
##Show (library action), 134.
“showobj” (library verb), 98.
“showverb”, 217.
“showverb” (library verb), 98.
sibling (built-in function), 50.
silent actions, 90.
Sinclair, Anita, 351.
Sinclair, Georgina, 389.
slash accents, 29.
small prime numbers, 42.
Smalltalk-80, 64, 73.
"smartcantgo.h" (library extension), 114.
smell (sense of), 109.
Smith, Brian D., 114.
Smith, Jayson, 126.
Smith, Lucian, 392, 395.
Smith, Rod, 355.
“snavig” spell, 156.
Snow, 435.
@sound_effect (assembly opcode), 315.
sound of scuttling claws, 149 ex45.
"sound.h" (function library), 320.
SoundTracker file, 324.
source code, 5.
source-level debugger, want of any, 105.
sp (built-in variable), 307.
space (white space), 6.
spaces (statement), 33.
Spacewar!, 345, 359.
Spanish, 243, 248, 252, 264, 267, 269.
punctuation, 29.
special_number (library variable), 134.
SpeedIF, 362.
"spin.inf" (example program), 114.
@split_window (assembly opcode), 310.
Spooner, Dennis, 125.
St John Passion (by Pärt, Arvo, opening line from libretto), 33.
St Peter, 215.
stack, 307.
stack frame, 319.
stack usage (limit), 338.
standard_interpreter (library variable), 308.
Standard interpreters, 308.
Star Trek, 141 ex34, 141 ex35, 141 ex36, 380.
Stark, Adam, 333.
StartDaemon (library routine), 147, 423.
StartTimer (library routine), 148, 423.
statement cannot be reached, 31, 302.
statements, 7.
static (library attribute), 107, 410.
status line:
  changing appearance of, 311.
status line (defined), 164.
Statusline (directive), 150.
stem, 252.
Stevens, Duncan, 4.
StopDaemon (library routine), 147, 424.
Stoppard, Tom, 266.
StopTimer (library routine), 148, 424.
store opcodes, 306.
story files, 5.
Story (library constant), 160.
streams of input/output, 309.
Strict mode, 6, 96.

STRICT_MODE (built-in constant), 97.

string arrays, use of, 43.

String (built-in metaclass), 49, 69.

String (form of array), 43.

String (statement), 30, 440 an 12.

(strings) (built-in printing rule), 32.

strings of characters, 42.

structures of data, 38.

Stub (directive), 279.

STUCK_PE (library constant), 237.

style of list, 201.

style (statement), 34, 313.

"style.h" (function library), 320.

sub-objects, 236.

subclasses, informal idea of, 61.

suffix, 252.

sunrise and sunset, 150 ex 48, 461 an 48.

superclass, 64.

supporter (library attribute), 118, 119, 410.

supporting scenery, 200.

sw_to (library property), 412.


switch (statement), 23.

switchable (library attribute), 125, 410.

switches (command line), 285.

Switches (directive), 287.

SwitchOption (library class), 332.

Symposium, The, 147.

Synapse Software, 355.

synonyms, 215.

system file, 104.

System_file (directive), 190, 280, 303.

tab (white space), 6.

tab-skip, 519.

table:

of conditions, 518.

of operators, 517.

table (form of array), 43.

TADS, 1, 359.

Tagalog, 248.

“take all”, 239.

##Take (library action), 89.

“take” verb, 217.

talkable (library attribute), 137, 410.

TALKING_REASON (library constant), 235, 455 an 38.

Tartt, Donna, 87.

task_scores (library array), 162.

TASKS_PROVIDED (library constant), 162.

taste (sense of), 109.

team of four adventurers, 156.

telepathy, 142 ex 39, 455 an 39.

teleportation, 154.

##Tell (library action), 134.

Ten Green Bottles, 21.

Teraform, 3.

terminating characters, 317.

terminating (part of Zcharacter directive), 317.

Tessman, Kent, 359.

TestScope (library routine), 232, 424.

TESTSCOPE_REASON (library constant), 235.

Text Adventure Development System, 1, 359.

text buffers, 42.

text cursor, 311.

text formatting, 311.

"text_functions.h" (function library), 320.

text of a command, 208.

Thackray, Jonathan, 3, 346, 348.

thedar (library object), 145.

<then-word>, 245, 262.

##TheSame (library fake action), 213.

thief in ‘Zork’, 148 ex 42, 457 an 42.

Things To Come, 400.

Thompson, Sir Eric, 166.

Thornton, Adam, 365.
Three Men in a Boat, 353.
@throw (assembly opcode), 36, 319.
"throw rock at dalek", 90.
##ThrowAt (library action), 90, 134.
##ThrownAt (library fake action), 90.
Thurston, Dylan, 41.
Ticmanis, Linards, 243.
tidying-up operations, 147.
Tiffany, 30.
tilde accents, 29.
time_left (library property), 148, 418.
the_time (library variable), 150.
time of day, 149, 228 ex88, 273, 485 an88.
time_out (library property), 148, 418.
time sequence, 150.
timed puzzles, 386.
TimePasses (entry point routine), 434.
"timepiece.h" (library extension), 151.
timers, 148.
maximum number active, 148.
starting and stopping, 148.
tracing active timers, 99.
"timers" (library verb), 99.
"timers" verb, 99.
"timesys.h" (library extension), 230.
"timewait.h" (library extension), 151, 230.
title page, 311 ex123, 508 an123.
Titus Andronicus, 367.
to (part of move statement), 51.
toadstool poisoning, 79.
@tokenise (assembly opcode), 317.
tokenising, 316.
tokens, 216, 222, 421.
tomato, 206.
too many errors, 290.
T00FEW_PE (library constant), 237.
T00LIT_PE (library constant), 237.
topic (grammar token), 226.
touch (sense of), 109.
touchability, 95, 232.
Tourtelot III, Gair, 174.
"trace" (library verb), 100.
Tractatus Logico-Philosophicus, 47, 194.
Traité des systèmes, 385.
transcript, 310.
transparent (library attribute), 120, 137, 410.
travertine, 47, 345.
treasures on status line, 312 ex125.
“tree” (library verb), 98.
Trent, 30.
Trewartha, Alan, 143.
Trubshaw, Roy, 346.
true (built-in constant), 10.
TryNumber (library routine), 208, 424.
Turner, Anson, 193, 205.
two-way door, 123.
Tyndale, William, 132 ex25, 446 an25.
u_to (library property), 412.
umlaut accents, 29.
uminary minus, 14.
“undo” (library verb), 98.
⟨undo-word⟩, 262.
Unicode, 30, 520.
Universal Exports, 43.
Unix, 97.
UnknownVerb (entry point routine), 220, 337, 434.
UnsignedCompare (library routine), 424.
until (part of do statement), 24.
untouchability, 95, 233.
untypeable words, 208.
upper-level window, 310.
UPTO_PE (library constant), 237.
USE_MODULES (library constant), 282.
“use” verb example, 225.
"utility.h" (function library), 320.
"V6Lib.h" (function library), 320.
vague_obj (library variable), 238.
VAGUE_PE (library constant), 237.
vague_word (library variable), 238.
Valediction, Of Weeping, 82.
Vance, Jack, 370.
variable strings, 440 an12.
variables, 11.
vehicles, 128.
Verb (directive), 217.
VERB_PE (library constant), 237.
⟨verb phrase⟩, 245.
verb_word (library variable), 140, 215.
verb_wordnum (library variable), 140.
"VerbLib.h" (library file), 75.
"verbLibm.h" (library file), 282.
verbs (Inform and English), 215.
Verdu, Mike, 357.
Versions, see Z-machine.
Vierra, Joe, 355.
Virgil (P. Vergilius Maro), 267.
visited (library attribute), 83, 155, 410.
VN_* (built-in constant), 281.
vocabulary size (limit), 337.

w_to (library property), 412.
Wagner, David, 114.
"waittime.h" (library extension), 230.
@@WakeOther (library action), 134.
Waldeck, Jean Frederic, 132.
warning messages, 8.
catalogued, 300.
Waste Land, The, 166.
water, 388.
Web page, 2.
weights, 148 ex43.
Weinstein, Jacob Solomon, 361.
Welch, Mark, 359, 360.
‘welcome’ message, 75.
Wells, H. G., 400.
“what is a grue”, 233.
“What next?”, 191 ex59, 466 an59.

"whatis.h" (library extension), 236.
when_closed (library property), 123, 418.
when_off (library property), 418.
when_on (library property), 418.
when_open (library property), 123, 418.
while (loop statement), 24.
white space, 6.
"whowhat.h" (library extension), 236.
Wichura, Christopher, 4.
Williamson, Gil and Beryl, 401.
Wilson, Gerry Kevin, 361, 367, 380, 401.
@window_size (assembly opcode), 313.
@window_style (assembly opcode), 313.
Winograd, Terry, 3.
with_key (library property), 119, 418.
with (part of Object statement), 55.
Wittgenstein, Ludwig, 47, 194.
wn (library variable), 207.
Wood, John, 4, 374.
Woods, Don, 4, 76, 345, 349.
word array, 42.
word breaking, 207, 311.
WordAddress (library routine), 208, 424.
WordInProperty (library routine), 255, 424.
WordLength (library routine), 208, 425.
workflag (library attribute), 202, 410.
world model:
1. Substance, 176.
2. Containment, 178.
5. Time, 183.
historical development, 3.
worn (library attribute), 116, 410.
WriteListFrom (library routine), 201, 425.
wrong guesses, 403.
Wuthering Heights, 167.
Wyber, Brendon, 361.

Xarfaggi's Physical Malepsy, 370.
XYZZY, nothing happens.
XYZZYnews, 1.

Y2, 393.
Yans, Cynthia, 360.
Yeandle, Graeme, 358.
⟨yes-word⟩, 262.
"yesno.h" (library extension), 95.
YesOrNo, 164.
YesOrNo (library routine), 425.
"you don't need to refer to", 106.

Z-machine:
   assembly language, 306.
   encoding text for dictionary, 317.
   introduction to, 304.
   limitations of, 334.
   memory architecture of, 307.
   stack, 307.
   versions of, 305, 334.
   Z-characters, 259.
   Z-machine standard, 5.
   testing adherence to, 308.
   Z-Machine Standard Code for Information Interchange, see ZSCII.
   Zazie dans le mÂetro, 344.
   Zcharacter (directive), 260, 317.
   ZIL, 304.
   "znsi.h", 46.
   Zork Implementation Language, 304.
   ZSCII, 10, 260.
   ZSCII value, 519.
   Zwissler, A. M. F., 492 an96.
This edition was begun on a beige Acorn Risc PC700 in 1999 and finished on a lime green Apple iMac DV in 2001, in the attic of a North Oxford town house (just across the road from the one in which Tolkien wrote *The Lord of the Rings*). The text was set in Monotype Plantin, a font based on designs by Christophe Plantin of Antwerp (1514–89), augmented by Computer Modern Roman typewriter by Donald Knuth (1938–). The printer’s ornaments are with one exception detailed from works of Albrecht Dürer (1471–1528): Chapter I, *Melancholia I* (engraving, 1514); Chapter II, *The Wire-Drawing Mill* (watercolour and gouache on paper, 1489); Chapter III, *St Jerome in his Study* (engraving, 1514); Chapter IV, *Portrait of a Young Venetian Woman* (oils on wood panel, 1505); Chapter V, *Traumgesicht* (ink on paper, 1525); Chapter VI, *The Large Turf* (watercolour and gouache on paper, 1503); Chapter VII, *The Knight, Death and the Devil* (engraving, 1513); Chapter VIII, *Wing of a Roller* (watercolour and gouache on vellum, 1512). The ornament to §23 is believed to be tzutz, the Mayan verb for completion. Type was set using CMacT\TeX\, Tom Kiffe’s port of Knuth’s program (1983), employing macros adapted from those used to typeset *The \TeX\book* (though sadly not the \plugh macro in that work’s Appendix D). Indices and bibliography were prepared automatically by scripts written in MacPerl 5, Matthias Neeracher’s port of Larry Wall’s formatting language. Final PDF was distilled using dvipdfm by Mark A. Wicks, and the book was printed from PDF by DeHARTs Printing of Silicon Valley.

facilis descensus Averno
noctes atque dies patet atri ianua Ditis
sed revocare gradum superasque evadere ad auras
hoc opus, hic labor est... latet arbore opaca
aureus et foliis et lento vimine ramus...