Index

#define, 14, 96, 102, 110, 235
#if, 236
#ifdef, 236
#include, 7, 70, 174
__FILE__, 236
__LINE__, 236
* operator, 20, 92, 94, 236
+ operator, 20, 236
++ operator, 25, 121, 230, 236
+= operator, 25, 236
, operator, 230, 236
- operator, 20, 236
-- operator, 25, 230, 236
-> operator, 141, 142, 236
. operator, 133, 142, 236
/ operator, 20, 21, 236
< operator, 30, 236
<< operator, 230, 233, 236
<= operator, 30, 236
= operator, 29, 31, 58, 123, 236
== operator, 29, 30, 123, 236
> operator, 30, 236
>= operator, 30, 236
>> operator, 230, 234, 236
? operator, 230, 236
[] operator, 102, 142, 236
% operator, 20, 21, 210, 236
& operator, 22, 92, 94, 98, 137, 230, 233, 234, 236
&& operator, 30, 236
| operator, 230, 234, 236
|| operator, 30, 236
~ operator, 230, 234
! operator, 30, 236
!= operator, 30, 236
^ operator, 234, 236
~ operator, 236
%c format control, 22, 23, 26, 119
%d format control, 16, 22, 23
%e format control, 23
%f format control, 17, 18, 22, 23
%lf format control, 22, 23
%o format control, 230
%s format control, 23, 24, 119, 120, 137
%u format control, 92
%x format control, 230

a grim era, 132
abs, 71
abstraction, 63, 83, 138, 145, 174, 181
acos, 71
Algol, 4, 212
algorithm, 105, 145, 183, 203–226
    analysis, 106, 203–205
aliasing, 93, 103
amicable numbers, 80
ampersand character, 22, 98
anagrams, 132
approximation, 150, 156–160
argc, 84, 127
argument, 64
argv, 84, 127, 169
array, 18, 101–129, 133, 145, 163, 199
    address, 107, 115, 180
    argument, 110, 135
    assignment, 135
    automatic, 115
    binary output, 195
    bounds checking, 102, 103, 108, 124
    dynamic, 163
    equality, 135
    global, 115
    initialization, 114, 118
    large, 113, 115, 167
    multi-dimensional, 114
    of characters, 118
    of files, 200
    of linked lists, 208
    of pointers, 126
    of strings, 126, 127
    of structures, 142
    parallel, 130, 144
    subscript, 102, 103, 113
    two-dimensional, 111, 118, 123, 126
ascending runs, 131, 228
ASCII, 25, 26, 57, 58, 122, 195, 210, 211
    table, 60
asin, 71
assert, 169
assignment statement, 8, 15, 25, 26, 31, 46, 58, 87, 94
    for arrays, 135
    for strings, 121
    for structures, 135
asymptotic cost, 204
atan, 71
atof, 122, 123
atoi, 122, 123, 129, 132
auto, 230
average-case analysis, 178, 215

backslash character, 19
backup, 10–12
    off-site, 11
base case, 74, 171, 213
Basic, 4
big-Oh notation, 204, 226
binary file, 195
binary numbers, 231

binary search, 105, 158, 206, 226
    ternary, 226
binary search tree, 177, 206, 207, 218
    average depth, 178, 191
    balanced, 191
    deletion, 192
    for sorting, 191
    insertion, 177, 185
    iterative implementation, 192
    polymorphic implementation, 183
    searching, 185
    smallest item, 192
    stick, 178
    traversal, 185
binary tree, 177, 219
    height, 191
    size, 191
bisection method, 158, 161
bit, 17, 91, 230
bit manipulation, 230
boundary case, 79
`break`, 39, 55, 56, 67
`bsearch`, 226
BST, *see* binary search tree
bubble sort, 105, 106, 129, 148, 151, 183, 204, 205
bucket, 210
byte, 91, 115, 165, 195

C
    advantages of, 5
    ANSI standard, 6, 32, 35, 70, 153
    history, 5
    preprocessor, 235, 239
C++, 4
calculation, 63, 145
calculator, 5, 17
`calloc`, 164, 166
`case`, 39
cast, 21, 25, 29, 92, 165, 181
Celsius, 28, 43
central processor unit (CPU), 2
change calculation, 43, 99
`char`, 19, 23, 24, 58, 91, 118, 197, 234
`char**`, 126, 169
Christie, I.W., 178
coding problem, 226
coin toss, 153
combinations, 80
command-line argument, 127, 195
comment, 6, 26, 62, 175, 237
comparison function, 181–183, 185, 190, 191, 217
compiler, 7, 9, 91, 92, 108, 114
    flags, 69, 71, 83
    preprocessor, 235
    warning messages, 35, 36, 89, 90
complex numbers, 143
compound interest, 49, 63, 72, 81
compound statement, 32, 33, 46, 51
computer
    early, 5
    hardware, 2, 10
    memory, 91, 101, 231
    software, 3
    speed, 2
computer science, 1, 105, 150, 203, 225
`const`, 183, 230
constant, 14, 15, 26, 37, 102, 235
    character, 19
    floating point, 19
    integer, 19
    pointer, 107, 115
    type, 18
`continue`, 230
control structure, 8, 101
`core`, 22
core dump, 22
`cos`, 71
`ctype.h`, 72, 123
cube root, 76
curve length, 156, 161

dangling else, 35
data abstraction, 138, 142, 174, 207, 218
data structure, 101, 171
date manipulation, 42, 47, 100, 115
De Morgan's laws, 31
debugger, 237, 239
declaration, 8, 26
    array, 101, 102
    function, 64
    pointer, 92
    structure, 133
`default`, 39
dice roll, 153
dictionary data structure, 206–212
Dijkstra, E.W., 1, 213
distinct words, 123, 132, 144, 163, 167, 183
divide and conquer, 146–151, 212
divide by zero, 21
`do`, 45, 56, 59, 229
`double`, 18, 19, 22, 23, 157, 164, 197, 234
Dutch national flag, 213

editor, 7, 9
`else`, 32, 33
empty statement, 32, 49, 53, 121
end of file, 58, 103
`enum`, 230
`EOF`, 58, 123
equality operator, 29, 34
Erlang, 4
errata page, ix
escape character, 19
Euclidean distance, 143
exclusive or, 234
execution time, 150, 167
exhaustive enumeration, 149
`exit`, 33, 84, 169
`EXIT_FAILURE`, 33, 83
`EXIT_SUCCESS`, 33, 83
`exp`, 71
exponent, 17, 19, 234, 239
exponential growth, 150, 177
expression, 15, 21
    evaluation order, 31
    logical, 29
    pointer, 92
`extern`, 230

`fabs`, 71

Fahrenheit, 28, 43
false, 29, 32, 46
`fclose`, 194, 200
Ferrari, 105, 205
`fflush`, 231
`fgets`, 200
Fibonacci numbers, 60
field width, 23
    negative, 24
FIFO queue, 174
file operations, 193–200
    merging, 199
    random access, 199
`FILE*`, 194, 195
flag, 55, 67, 96, 124
`float`, 17, 19, 23, 101, 157, 164, 234
floating exception, 22
floating point precision, 17, 18, 78, 81, 157, 159, 234, 239
`fopen`, 194, 195, 200
`for`, 45, 53, 101, 102, 171
format control string, 22
Fortran, x, 4, 6, 57, 84
`fprintf`, 193, 194
`fread`, 194, 195, 197
`free`, 164, 166, 170, 185, 200
Free Software Foundation, 7
`freopen`, 194
`fscanf`, 194
`fseek`, 194, 199, 202
`ftell`, 231
full house, 161
function, 63–79, 83–91, 236
    argument variable, 73, 86, 89, 94, 108
    array argument, 107, 110, 111
    call, 65
    choice of arguments, 72
    compilation, 67
    declaration, 64, 84
    evaluation, 65
    library, 70, 83
    `main`, 83
    pointer, 179, 181
    pointer argument, 94, 96
    prototype, 66, 67, 70, 183
    recursive, 74, 91, 147, 149, 171, 185
    return value, 64, 84
    scope, 86, 87, 97
    static, 185
    structure arguments, 141
    without arguments, 84
functional language, 4, 74, 87
`fwrite`, 194, 197, 239

gambling games, 151
`gcc`, 7, 35, 68, 71, 98, 237
`gdb`, 239
generate and test, 145–146, 148, 150
geometric sequence, 167
`getc`, 194, 195
`getchar`, 57, 72
`gets`, 200
`getword`, 124, 168, 185, 187
`goto`, 230
greedy heuristic, 151

guard, 32, 46, 49, 54, 236

handle to structure, 170, 171, 174, 183, 195, 209
hashing, 207–212
    collision resolution, 208
Haskell, 4
header file, 71, 183
heap, 218
    construction, 220
heap sort, 218–222, 228
`helloworld.c`, 6, 97
hexadecimal number, 19, 230, 239
Hoare, C.A.R., 212

identifier, 13, 101, 139
`if`, 32, 39, 54, 101
imperative language, 4
in-order traversal, 185, 190, 207, 218
`inf`, 21, 239
informatics, 1
information, 1, 203
input buffer, 22
insertion sort, 131, 191
`int`, 16, 19, 22, 23, 91, 101, 164, 195
`int_swap`, 95, 99, 106
integer
    arithmetic, 16, 19, 20, 232
    division, 20
    hash value, 208
    negative, 232
    overflow, 16, 48, 210, 230
    subtraction, 232
    unsigned, 92, 164, 232
inversions, 131
`isalnum`, 123
`isalpha`, 72, 123
`isascii`, 72, 123
`isblank`, 123
`isdigit`, 72, 123
`islower`, 72, 123
`isprint`, 123
`ispunct`, 123
`isspace`, 72, 123
`isupper`, 72, 123
iteration, 45, 63, 74, 145

Java, 4

K&R, 5
Kernighan, B., 5
KISS, 10, 32
knapsack problem, 149

lazy evaluation, 32
library, 70, 121
LIFO queue, 174
linear insertion sort, 131
linear search, 105, 124, 125, 170
linked list, 145, 171, 178, 207, 208, 212, 218
    insertion, 172, 177, 190
    searching, 190
Lisp, 4
`list_t`, 171, 190
`log`, 71
logic-based language, 4, 74

logical
and, 29, 32
expression, 29
or, 29, 32
`long`, 230, 234
`long long`, 234
loop, 45–58, 106
body, 46, 56
for reading, 57
guard, 54, 56, 57, 121, 123
infinite, 49
nested, 47, 50, 56, 63, 111
termination, 54, 75
lowercase character, 60, 72
`ls`, 7

`M_E`, 71
`M_PI`, 71
`M_SQRT2`, 71
macro, 236, 239
`main`, 7, 67, 69, 83, 127
return value, 83, 97
`make`, 69, 238
`malloc`, 164, 165, 167, 200, 222
`man`, 71, 121
mantissa, 17, 234, 239
mask operation, 234
`math.h`, 70
mathematics library, 70
median, 216, 227
`memcmp`, 231
`memcpy`, 231
memory, 2, 115, 126
allocation, 163
consumption, 203, 225
leak, 120, 167, 172
management, 167
read-only, 120
word, 91, 101, 164, 231
merge sort, 222–225, 228
merging, 199, 222
ML, 4
modulus, 20
Monte Carlo estimation, 154
moons, 133
multi-way tree, 192
multiplicative operators, 20

`nan`, 21, 239
newline character, 19, 24, 50, 54, 57, 200
Newton Raphson method, 76, 162
non-linear equation, 158
`NULL`, 127, 166
null byte, 118, 121, 124, 127, 169, 200
number representation, 230, 239
numerical integration, 158

$O()$ notation, 204, 226
obfuscated C, 50, 121
object file, 69
object-oriented language, 4
octal number, 19, 230
`od`, 239
operating system, 3, 83, 84, 96
operator
arithmetic, 20
logical, 29
mask, 234
overloading, 233
precedence table, 236
relational, 29
shift, 233
unary, 92
optimal algorithm, 224
overloading, 233

palindromes, 131
parentheses, 20, 31, 142
partitioning, 212, 222, 227
two way, 218, 227
Pascal, 4
perfect numbers, 80
pivot value, 227
PL/1, 4
planets, 133, 142, 177
pointer, 91–96, 107
anonymous, 165, 181
argument, 94, 96, 141
arithmetic, 115
assignment, 116
comparison, 116
constant, 115, 119, 127, 180
difference, 116
in an array, 126
initialization, 92
null, 127
operations, 92
to a binary search tree, 183
to a function, 179
to a linked list, 171
to a pointer, 169
to an array, 107
variable, 92, 119, 164, 170
poker hand, 161
polygon, 143
polymorphism, 176, 182, 183, 185, 226, 227
portability, 5, 164
post-order traversal, 185
postincrement operator, 25, 47, 121, 236
`pow`, 71, 72
pre-reading, 57
precedence, 20, 31, 36, 94, 111, 142
table, 236
predecrement operator, 230
preincrement operator, 25, 230, 236
preprocessor, 235, 239
pretty printer, 238
prime numbers, 55, 62, 67, 145, 209
`printf`, 7, 16, 17, 23, 26, 121, 193
printing numbers, 23
priority queue, 218
problem size, 204
problem solving, 145–160
techniques, *see* approximation, divide and conquer, generate and test, simulation
procedural language, 4, 87
procedure, 84
profiler, 237
program
arguments, 127

development, 9, 78, 83, 175
layout, 15, 50
return value, 84, 97
termination, 84
programming language, 3
Prolog, 4
prompt, 7, 26, 49, 67
pump priming, 57
`putc`, 194
`putchar`, 57

`qsort`, 183, 191, 197, 218, 227
quadratic roots, 41
queue, 174, 190, 207
quick sort, viii, 212–218, 222, 227
pivot value, 212, 214
ternary, 227

`rand`, 151, 153, 216
`RAND_MAX`, 153
`random`, 153
random access file, 199
randomization, 209, 216
reading numbers, 22, 57
`realloc`, 164, 166, 167
recurrence relation, 74, 207, 214, 224, 227
recursion, 74, 81, 91, 130, 145, 147, 149, 185, 206, 212
base case, 75, 171
mutual, 82
`register`, 230
repeat-until, 56
reserved word, 13, 164
`return`, 7, 64, 67, 73, 75, 83, 84
Ritchie, D., 5
root finding, 76, 158, 162
rounding error, 17, 18, 78, 81, 157, 160, 235

scaffolding, 78, 96
scalar variable, 101
`scanf`, 22, 23, 26, 33, 49, 57, 85, 123
Scheme, 4
scope, 86, 87, 97
search tree, *see* binary search tree
searching, 206
algorithm, *see* binary search, linear search
seed, 153
`SEEK_CUR`, 199
`SEEK_END`, 199
`SEEK_SET`, 199
segmentation fault, 93, 96, 103
selection, 32–40, 63, 145
selection sort, 130, 148, 151
semantics, 3
semi-colon, 15, 49, 67
sentinel, 114, 118, 127
separate chaining, 212
separate compilation, 68, 81, 87, 174, 183, 185
sex at noon taxes, 132
shell expansion, 128, 195
shift operators, 233
`short`, 230, 234
shortest path problem, 225
side effect, 31, 58, 87, 89, 121
sign-magnitude representation, 232
simulation, 151–156
`sin`, 71
`size_t`, 164, 182, 190
`sizeof`, 164, 165, 182, 236
slide rule, 5
Smalltalk, 4
software engineering, 1, 79, 238
sorting, 105, 131, 151, 160, 183, 205
algorithm, *see* bubble sort, heap sort, insertion sort, merge sort, quick sort, selection sort
choice of algorithm, 225
exchanges required, 228
source code control, 237
spiral, 45, 54, 75, 106
`sprintf`, 231
`sqrt`, 71, 85
square root, 161
`srand`, 151, 153
`srandom`, 153
`sscanf`, 231
stack, 75, 83, 84, 86, 91, 92, 174, 190, 192, 207
`static`, 90, 185, 230
`stdarg`, 231
`stderr`, 193, 202
`stdin`, 193
`stdio.h`, 72, 194
`stdlib.h`, 33, 83, 122, 151, 183, 218, 226
`stdout`, 193, 202
storage class, 90, 183, 185, 230
`strcasecmp`, 122
`strcat`, 122, 132
`strchr`, 231
`strcmp`, 122, 124
`strcpy`, 121, 122, 124, 135
`strcspn`, 231
stream, 193
string, 7, 14, 118–129, 195
allocation, 169, 187
comparison, 123
hash function, 209
initialization, 119, 126
library, 121
numeric, 123
pointer, 120, 123
`string.h`, 122, 124
`strlen`, 122, 132, 165, 169
`strncat`, 122
`strncmp`, 122
`strncpy`, 122
`strpbrk`, 231
`strrchr`, 231
`strspn`, 231
`strstr`, 231
`strtok`, 231
`struct`, *see* structure
structure, 133–142
argument, 139
array of, 142
assignment, 135
equality, 135
initialization, 134
linked, 170
nested, 137
pointer, 141

printing, 137
reading, 137
recursive, 171
sorting, 160
tag, 134
type naming, 139
stub, 78
subroutine, 84
subset sum problem, 148, 161, 162
`switch`, 37, 45, 56, 229
syntax, 9, 145
`system`, 231
system software, 3

tab character, 19
`tan`, 71
tax rates, 36
`tee`, 202
terminal, 23, 193
testing, 78
text file, 193
three-n problem, 53, 61
timetabling problem, 225
`tolower`, 72
`toupper`, 72
towers of Hanoi, 146, 149, 150, 162
tractor, 105, 205
trapezoidal rule, 158, 161
treadmill, 45, 49, 54
tree, *see* binary tree
tree sort, 191
`tree_t`, 183, 185
triangle numbers, 74, 81
true, 29, 32, 46, 169
two-dimensional
array, 111
coordinates, 143
structure, 177
table, 50, 54, 63
twos-complement representation, 232, 239
type
anonymous, 181
conversion, 21, 25
declaration, 15, 64
for constant, 18
return, 64
user defined, 108
`typedef`, 108, 113, 124, 134, 171

`ungetc`, 231
`union`, 230
Unix, 3, 7, 84, 97, 103, 202, 226, 239
file redirection, 22, 62, 193, 202
`unsigned`, 210, 230, 234
uppercase character, 60, 72

validation, 79
variable, 15
address, 22, 91, 92, 101
argument, 64, 67, 73, 94
automatic, 91
global, 87, 88, 229
in an array, 101
initialization, 16, 92, 114, 134
local, 64, 67, 73, 86, 87
pointer, 91, 92, 94, 115, 170
shadowing, 89
static, 90, 92, 114
structure, 133
`vice`, 7
`void`, 84, 165
`void*`, 165, 181, 183, 185, 190
`volatile`, 230

web site, ix
`while`, 7, 53, 103, 171
Williams, J.W.J., 218
Windows, 3, 7
word count, 61
word frequency counting, 185
worst-case analysis, 107, 214, 218, 222

zero, 29, 118, 169, 239
zodiac, 191