Index

access data member, 230
member function, 261
access restrictions private, 269
public, 269
access specifier private, 269
public, 269
Ackermann function, 203
addition assignment operator, 45
addition operator, 41
address of a memory cell, 13
of a variable, 23
address operator, 129
adjustment of array argument, 175
algorithm cache oblivious, 15
algorithmic denial, 64
ANSI
random number generator, 274
antivivace, 64
application program, 6
argument-dependent lookup, 235
Arithmetic Evaluation Rule 1, 38
Arithmetic Evaluation Rule 2, 59
Arithmetic Evaluation Rule 3, 39
arithmetic expression, 25
arithmetic operators, 41
arithmetic type, 38
arity, 26
array, 124
as function argument, 175
dimension, 143
drawbacks, 154
element, 123
fixed length, 154
incomplete type, 125
index, 125
initialization, 125
initialization from string literal, 140
initialiser list, 125
multidimensional, 143
simulation by onedimensional array, 144
not self-describing, 127
of pointer, 145
out-of-bound index, 124
pointer to, 144
random access, 125
security issues, 154
subclass, 125
suffix operator, 125
underlying type, 124
zero-terminated, 140
array-to-pointer conversion, 131
ASCII code, 139
assertion, 167
assignment
memberwise, 232
of a struct value, 231
of reference, 241
pointer, 129
assignment operator, 27
associative operation, 38
associativity right, 39
right, 39
associativity of operator, 38
attacker, 154
automatic storage duration, 80
base of a floating point number system, 105
BASIC
programming language, 174
behavior implementation defined, 20
undefined, 20
unspecified, 20
binary expansion
of natural number, 48
of real number, 106
binary operator, 24
binary representation
of int value, 49
of unsigned int value, 49
binary search, 220
binary to decimal conversion, 49
binomial coefficient, 219
bit, 13
bitwise operator, 65
block, 76
BODMAS, 38
body of do statement, 83
of for statement, 73
of function, 168
of while statement, 81
bool, 61
Booleans, 59
Boolean evaluation, 62
Boolean expression, 63
Boolean function, 59
completeness, 60
break statement, 84
bubble sort, 207
bug, 77
built in type, 23
burst, 54
C++ standard, 20
cache, 15
cache oblivious algorithm, 15
call arguments, 188
call by reference, 242
call by value, 242
call stack, 168
cancellation
in floating point computations, 114
cast (functional notation), 103
cast expression, 103
central processing unit (CPU), 13
char literal, 139
promotion to (unsigned) int, 139
type, 139
character, 139,
control, 140
line break, 140
choosing numbers
game, 272
Church, Alonzo (1903-1995), 91
Church's Turing Thesis, 91
data
access specifier, 269
constructor, 263
definition, 268
implementation, 269
member declaration, 269
member function, 260
member function call, 262
member operator, 266
method, 262
285

<table>
<thead>
<tr>
<th>Concept</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>control character</td>
<td>140</td>
</tr>
<tr>
<td>control flow</td>
<td>71</td>
</tr>
<tr>
<td>iteration</td>
<td>72</td>
</tr>
<tr>
<td>jump</td>
<td>84</td>
</tr>
<tr>
<td>linear</td>
<td>71</td>
</tr>
<tr>
<td>selection</td>
<td>71</td>
</tr>
<tr>
<td>statement</td>
<td>72</td>
</tr>
<tr>
<td>variable</td>
<td>74</td>
</tr>
<tr>
<td>conversion</td>
<td></td>
</tr>
<tr>
<td>array to pointer</td>
<td>131</td>
</tr>
<tr>
<td>explicit</td>
<td>103</td>
</tr>
<tr>
<td>floating point</td>
<td>102</td>
</tr>
<tr>
<td>implicit</td>
<td>47</td>
</tr>
<tr>
<td>integral</td>
<td>47</td>
</tr>
<tr>
<td>promotion</td>
<td>63</td>
</tr>
<tr>
<td>standard</td>
<td>263</td>
</tr>
<tr>
<td>user defined</td>
<td>265</td>
</tr>
<tr>
<td>CPU</td>
<td>13</td>
</tr>
<tr>
<td>Cramer's rule</td>
<td>158</td>
</tr>
<tr>
<td>data encapsulation</td>
<td>259</td>
</tr>
<tr>
<td>data member access for</td>
<td>236</td>
</tr>
<tr>
<td>of struct</td>
<td>229</td>
</tr>
<tr>
<td>De Morgan's laws</td>
<td>63</td>
</tr>
<tr>
<td>debugging output</td>
<td>77</td>
</tr>
<tr>
<td>decimal-to-binary conversion</td>
<td>48</td>
</tr>
<tr>
<td>declaration</td>
<td>30</td>
</tr>
<tr>
<td>local</td>
<td>77</td>
</tr>
<tr>
<td>of a class member</td>
<td>269</td>
</tr>
<tr>
<td>of a function</td>
<td>172</td>
</tr>
<tr>
<td>of a variable</td>
<td>23</td>
</tr>
<tr>
<td>of friend</td>
<td>276</td>
</tr>
<tr>
<td>struct</td>
<td>230</td>
</tr>
<tr>
<td>declaration statement</td>
<td>30</td>
</tr>
<tr>
<td>declarative region</td>
<td>78</td>
</tr>
<tr>
<td>default argument of a function</td>
<td>185</td>
</tr>
<tr>
<td>default construction</td>
<td>264</td>
</tr>
<tr>
<td>default initialization</td>
<td>232</td>
</tr>
<tr>
<td>default initialization</td>
<td></td>
</tr>
<tr>
<td>by default constructor</td>
<td>264</td>
</tr>
<tr>
<td>definition of a class</td>
<td>268</td>
</tr>
<tr>
<td>of a function</td>
<td>168</td>
</tr>
<tr>
<td>of a variable</td>
<td>24</td>
</tr>
<tr>
<td>struct</td>
<td>228</td>
</tr>
<tr>
<td>delete expression</td>
<td>138</td>
</tr>
<tr>
<td>denormalized number</td>
<td>117</td>
</tr>
<tr>
<td>difference operator</td>
<td>129</td>
</tr>
<tr>
<td>dimension</td>
<td></td>
</tr>
<tr>
<td>multidimensional array</td>
<td>143</td>
</tr>
<tr>
<td>directive</td>
<td></td>
</tr>
<tr>
<td>include</td>
<td>21</td>
</tr>
<tr>
<td>using</td>
<td>32</td>
</tr>
<tr>
<td>discarding control</td>
<td>250</td>
</tr>
<tr>
<td>discriminant</td>
<td></td>
</tr>
<tr>
<td>of a quadratic equation</td>
<td>114</td>
</tr>
<tr>
<td>divide and conquer</td>
<td>207</td>
</tr>
<tr>
<td>division assignment</td>
<td>45</td>
</tr>
<tr>
<td>operator</td>
<td></td>
</tr>
<tr>
<td>do statement</td>
<td>83</td>
</tr>
<tr>
<td>body</td>
<td>83</td>
</tr>
<tr>
<td>domain</td>
<td>166</td>
</tr>
<tr>
<td>of a function</td>
<td>110</td>
</tr>
<tr>
<td>random</td>
<td>44</td>
</tr>
<tr>
<td>random number generator</td>
<td>272</td>
</tr>
<tr>
<td>dynamic memory allocation</td>
<td>136</td>
</tr>
<tr>
<td>dynamic programming</td>
<td>146</td>
</tr>
<tr>
<td>dynamic storage duration</td>
<td>136</td>
</tr>
<tr>
<td>Dynamic Storage Guideline</td>
<td>139</td>
</tr>
<tr>
<td>editor</td>
<td>11</td>
</tr>
<tr>
<td>effect</td>
<td></td>
</tr>
<tr>
<td>of a function</td>
<td>22</td>
</tr>
<tr>
<td>of a statement</td>
<td>29</td>
</tr>
<tr>
<td>of an expression</td>
<td>25</td>
</tr>
<tr>
<td>of a function, 25</td>
<td></td>
</tr>
<tr>
<td>effect (semantical term)</td>
<td>23</td>
</tr>
<tr>
<td>element of array</td>
<td>123</td>
</tr>
<tr>
<td>encapsulation of data</td>
<td>259</td>
</tr>
<tr>
<td>equality point</td>
<td>129</td>
</tr>
<tr>
<td>Eratosthenes' sieve</td>
<td>123</td>
</tr>
<tr>
<td>Euclidean algorithm</td>
<td>199</td>
</tr>
<tr>
<td>evaluation of an expression</td>
<td>25</td>
</tr>
<tr>
<td>order of operands</td>
<td>27</td>
</tr>
<tr>
<td>short circuit</td>
<td>64</td>
</tr>
<tr>
<td>evaluation sequence</td>
<td>40</td>
</tr>
<tr>
<td>Excel 2007 bug</td>
<td>107</td>
</tr>
<tr>
<td>executable</td>
<td>12</td>
</tr>
<tr>
<td>execution</td>
<td>29</td>
</tr>
<tr>
<td>explicit conversion</td>
<td>103</td>
</tr>
<tr>
<td>exponent</td>
<td></td>
</tr>
<tr>
<td>of a floating point number</td>
<td>105</td>
</tr>
<tr>
<td>expression</td>
<td>25</td>
</tr>
<tr>
<td>arithmetic</td>
<td>25</td>
</tr>
<tr>
<td>Boolean</td>
<td>63</td>
</tr>
<tr>
<td>cast</td>
<td>103</td>
</tr>
<tr>
<td>composite</td>
<td>25</td>
</tr>
<tr>
<td>constant</td>
<td>124</td>
</tr>
<tr>
<td>delete</td>
<td>138</td>
</tr>
<tr>
<td>effect</td>
<td>25</td>
</tr>
<tr>
<td>evaluation of</td>
<td></td>
</tr>
<tr>
<td>evaluation sequence</td>
<td>40</td>
</tr>
<tr>
<td>function call</td>
<td>168</td>
</tr>
<tr>
<td>literal</td>
<td>23</td>
</tr>
<tr>
<td>value</td>
<td>26</td>
</tr>
<tr>
<td>mixed</td>
<td>47</td>
</tr>
<tr>
<td>new</td>
<td>137</td>
</tr>
<tr>
<td>of type void</td>
<td>169</td>
</tr>
<tr>
<td>order of effects</td>
<td>54</td>
</tr>
<tr>
<td>primary</td>
<td>25</td>
</tr>
<tr>
<td>value</td>
<td>26</td>
</tr>
<tr>
<td>type of</td>
<td>25</td>
</tr>
<tr>
<td>value of</td>
<td>25</td>
</tr>
<tr>
<td>variable</td>
<td>23</td>
</tr>
<tr>
<td>expression statement</td>
<td>30</td>
</tr>
<tr>
<td>expression tree</td>
<td>40</td>
</tr>
<tr>
<td>fair dine</td>
<td>273</td>
</tr>
<tr>
<td>false</td>
<td>61</td>
</tr>
<tr>
<td>Fibonacci numbers</td>
<td>200</td>
</tr>
<tr>
<td>finite</td>
<td></td>
</tr>
<tr>
<td>finite floating point number system</td>
<td>105</td>
</tr>
</tbody>
</table>
alphabet, 211
fractal, 214
graphical interpretation, 213
initial word, 211
production, 211
Landenberger, Arndt (1926–1985), 217
line break character, 140
linear congruential generator, 269
linear congruential method, 269
linear control flow, 71
linker, 179
Linux, 14
literal, 23
bool, 61
char, 139
doable, 101
float, 101
hexadecimal, 51, 161
int, 38
long double, 116
octal, 51
string, 139
unsigned int, 47
loaded dice, 273
local declaration, 77
local scope, 78
logical parentheses, 51
logical operators, 82
logical parentheses
leading operand, 51
secondary operand, 51
long double, 116
long int, 53
lookup
argument-dependent, 235
loop, 72
infinite, 74
progress towards termination, 74
invoke, 26
inline to value conversion, 26
Mac OS, 14
machine epsilon, 109
machine language, 11, 14
macro, 167
main function, 22
main memory, 13
Mandelbrot set, 119
manipulation
of a floating point number, 105
mathematical induction, 100
member access
in a struct, 230
member access operator, 230
member function, 260
access for, 261
call, 262
implicit call arguments, 261
member function
and overload, 262
member operator
of class, 266
member specification
of a struct, 229
member wise assignment, 232
member-wise initialization, 231
memory cell, 13
address, 13
memory stack, 139
merge-sort, 207
complexity, 209
Merge sort primitive, 7
method
of class, 262
minimum-sort, 204
complexity, 205
mixed expression, 47
modulation, 177
modulo assignment operator, 45
modulo operator, 43
multi-dimensional array, 143
dimension, 143
simulation by multi-dimensional array, 144
multiplication assignment operator, 45
multiplication operator, 27, 41
mutating function, 177
name
clash, 22
hiding, 79
of a class, 269
of a function, 168
of a type, 23
of a variable, 23
of formal argument, 168
qualified, 22
unqualified, 22
namespace, 22
namespace scope, 78
nested type, 267
new expression, 137
normalized floating point number, 105
null pointer, 130
null pointer value, 130
null statement, 29
numeric limits
of floating point types, 116
of integral types, 45
object, 24
unnamed, 24
object code, 178
open source software, 180
operand, 26
evaluation order, 27
operating system (OS), 14
Linux, 14
Mac OS, 14
Unix, 14
Windows, 14
operator
addition, 41
addition assignment, 45
address, 129
arithmetic, 41
arithmetic assignment, 45
array, 26
assignment, 27
associativity, 38
binary, 26
binding, 38
bitwise, 65
conditional, 92
dereference, 129
division assignment, 45
division operator, 271
function call, 271
functional notation, 233
functionality, 26
index notation, 234
input, 28
integer division, 43
left-associative, 29
logical, 62
modulo, 43
modulo assignment, 45
multiplication, 27, 41
multiplication assignment, 45
operator, 26
output, 28
overloading, 233
post-decrement, 44
post-increment, 44
pre-decrement, 44
pre-increment, 44
precedence, 38
relational, 62
return value, 28
subscript, 126, 134
subtraction, 41
subtraction assignment, 45
ternary, 92
ternary operator, 28
unary, 26
unary minus, 44
unary plus, 44
operator token, 27
overloaded, 39
order of effects, 54
OS, 14
offset, 126
array index, 126
output operator, 28
output stream, 28
overflow
of value range, 46
overloading
  argument-dependent lookup, 235
  best match, 235
  of functions, 233
  of operators, 233
overloading resolution, 234
parallel computer, 16
past the end pointer, 132
PEMDAS, 38
permutation, 158
perpetual calendar, 192
pipe, 69
platform, 14
point of declaration, 78, 91
pointer, 128
  adding an integer, 132
  arithmetic, 132
  assignment, 129
  comparison, 133
  equality, 129
  initialization, 129
  null, 130
  null value, 130
  past the end, 132
  subscript operator, 134
  subtraction, 134
  to array, 144
  type, 128
pointer type
  underlying type, 128
porting, 12
post-decrement operator, 44
post-increment operator, 44
postcondition
  of a function, 166
precondition
  of a floating point number system, 105
  rational numbers, 226
preliminaries
  infinite, 198
  primitive, 202
  tail end, 201
premise call, 197
recursive function, 157
reversing, 200
termination, 200
refactoring, 88
relevance, 241
  assignment, 241
  initialization, 241
  Reference Manual, 243
  reference type, 241
relational operators, 62
relative error
  in floating point computations, 109
reserved name, 24
return by reference, 242
return by value, 242
return statement, 30, 93
return type, 168
return value, 26
Reverse Polish Notation, 68
right associativity, 39
runtime error, 167
value, 26
Sarrus’ rule, 158
scope
  global, 78
  local, 78
namespace, 78
semantics
  of a declaration, 78
  of a function declaration, 172
  of a struct, 232
  of an expression, 30
  of a function, 157
  of a struct, 232
  of a statement, 72
  of a variable, 93
  of a variable declaration, 131
  of a variable expression, 127
  semantical value range
  of a struct, 232
setwise, 147
separate compilation, 178
sequence point, 54
Sheffer stroke, 64
Sheffer, Henry M. (1838–1964), 64
short circuit evaluation, 64
short int, 53
shortest path problem, 146
side effect, 25
Sieve of Eratosthenes, 123
sign
  of a floating point number, 105
signature of a function, 157
signed char, 63
significant
  of a floating point number, 105
Single Modification Rule, 65
source code, 12
availability, 140
spaghetti code, 174
special character, 139
standard conversion, 265
standard error, 28
standard input, 27
standard library, 22
  mathematical functions, 186
  std::cerr, 28
  std::cin, 28
  std::cout, 28
  std::endl, 185
  std::ifstream, 183
  std::ofstream, 183
standard output, 28
statement, 29
  break, 84
  compound, 76
  continue, 85
  control, 72
  declaration, 30
  do, 83
  execution, 29
  expression, 30
  for, 73
  goto, 83
  if, 71