

INDEX

= (assignment operator)	130, 816
= 0 (as function definition)	827
== (equality test operator)	146
! (logical NOT operator)	159
!= (inequality test operator)	146
#define	140, 158
#include	100, 112, 124
% (arithmetic operator)	131
& ("address of" operator)	486, 600, 644
& (bitwise AND operator)	507
&& (logical AND operator)	159
(* ...). (operator combination)	see -> (pointer operator)
* (arithmetic operator)	131
*/ (comment delimiter)	125
*=	163
*this	614
+ (arithmetic operator)	131
++ (increment operator)	162, 163
+= ("add to memory" operator)	162
, (comma operator)	168
- (arithmetic operator)	131
-- (decrement operator)	163
-= ("subtract from memory" operator)	163
-> (pointer operator)	613
. (member access operator)	467, 557
/ (arithmetic operator)	131
/* (comment delimiter)	125
// (comment delimiter)	125
/=	163
2-3 tree	870...873
:: (scope qualifier operator)	557
<< (output operator)	109, 129, 211, 305, 807, 808
<= (less than test)	146
>= (greater than test)	146
>> (input operator)	110, 128, 210, 306, 807, 808
? ... : (operator combination)	see Conditional expression
[] operator	287, 313
\ character	129
\0	see null character
\n	307
\r	307
^ (bitwise XOR operator)	507
(bitwise OR operator)	507
(logical OR operator)	159

~ (bitwise NOT operator)	507
π -Cannon (program example)	268 ... 273
abs() (maths function)	250, 251
A-to-D	26
Abbreviations in C++	110, 131, 162, 189, 508
Abstract class	795, 824, 826, 830, 853
Abstract data type	102
Abstraction (functions)	238
ADA	97
Adapter pattern	1073
Address register	5
Addresses	see Memory addresses
Addressing modes	8
Algol	86
AlgolW	94
ALU	4, 5
Arguments to functions	81, 137, 303, 318, 645
Arithmetic operators	131
ARPA network	64
Array	76, 600
Array bounds and checks	284
Array definitions	286, 310
Array initialization	289, 311
Arrays and pointers	606, 614, 650...653
Arrays as arguments to functions	299...304, 318
Arrays in assembly language	283, 600
Arrays, multidimensional	310...318
ASCII character code	11
Assembler	34
Assembly language	34, 71
assert.h	261
Assertions	261
Assignment statement	127, 130
Automatic storage	91, 253, 287
AVL tree	842...868
bad() (filestream function)	211
Balance (property of treenode)	841
Base class	827
BASIC	83
BCPL	98
Bell	129
Big O()	see Cost (complexity) of calculations
Binary files	208, 486
Binary search	420, 422
Binary tree	659, 699...721, 798, 799, 875
Bit fields	529...531
Bitmaps	528, 552...575, 783
Bits	9, 505...510
Block (disk memory)	17, 22
Blocks (program construct)	155
bool (data type)	158
Boolean	95, 156...158

Borland	105, 278, 361, 1135
Bounded array	928...932
Branch instruction	7
break statement	170
BTree	873...
Building a project	117
Built in data types	126
Bus	3, 15
Bytes	9
C	98
C functions called from C++	250
C++	101, 164
C++ input and output	107... 111
Cache memories	10
Calculator (example program)	179... 185
Call graph	452, 456
case (part of switch statement)	174
case fall through	175
catch	940
CD-ROM disks	16
cerr	109
CFC (program example)	147...149
char* (C's generic pointer)	606
char* (pointer to character(s))	606, 607, 615
Character data	95, 126
cin	108
class	547
Class (shared) data	789...792
Class declaration	549...553
Class definition	553
Class browser	1072
clear() (filestream function)	261
Clock	4, 25, 363
close() (filestream function)	211
COBOL	86, 87
Code coverage tool	445...448, 866...868
Collection classes	659...722
Column	see Arrays multidimensional
Command handlers	1078, 1144, 1153, 1156
Comments	73, 124, 125
Comparison operators	146
Compilation errors	116
Compiler directives	278
Compilers	73...75
Compound statement	146, 155
Concrete class	824
Conditional compilation	278, 360, 363
Conditional expression	189
const	139, 303, 619
const member functions	559
Constants	139, 619
Constructor functions	558, 559

continue statement	170
Control characters	129
Control structures	95
Converting data	142
Cost (complexity) of calculations	314, 423, 428, 432, 669, 672, 700, 841
Count Sort	426
cout	109
CPU	3, 5...9,
ctype.h	310
Cursor addressable displays	355
D-to-A	26
daddr_t	875
Data register	5, 29
Data structures	76, 78, 85, 86, 286, 310
Data types	75, 95, 126
Debugger	117, 118, 134
Declarations	114
default (part of switch statement)	178
Default arguments for functions	247
Definition v. declaration	114
delete operator	603, 617
delete [] operator	605, 617
Design (documentation)	455...457, 729, 732, 784, 825, 1008
Design (examples)	111, 132, 135, 180, 193, 199, 225, 269, 274, 290, 328, 344, 374, 92,407, 451...457, 488, 517, 529
Design of classes	563, 581, 661, 673, 682, 711, 765, 778, 1020, 1023, 1025, 1032
Designs using classes	627, 726, 759, 956, 1007, 1018, 1065
Destructor functions	808, 969
Device drivers	62, 98
Direct memory access	20
Disk-head	18, 19
Disks	16...22, 52
DO statement	see For statement
Doom (game)	1015
do ... while	see Repeat loop
double (data type)	126
Dynamic array	659, 680...689, 727
Dynamic binding	835
Dynamic data	85, 96, 101
Editor	106
Eiffel	102
Empty body for loop	169
Encryption example	328...343
End of file condition	215, 216
endl	110
enum	158, 459...465
eof() (filestream function)	212, 215
ET++	1135

Ethernet	66
Exchange rates (example program)	132
Exceptions	941...949
exit()	191, 938
Expressions	130
fabs() (maths function)	151
fail() (filestream function)	212
Fetch-decode-execute cycle	5, 30
File directory	22
File input and output	207 ... 222
File of records	482...503, 526
Files	21, 57...58, 85, 86
Files	see also Modules
Filescope	288
filestream objects	208, 597
Flags register	6, 29, 30
float (data type)	126
Floating point numbers	13
Flow chart	457
For statement	78, 95, 145, 165 ... 170
Format	82, 110, 143, 197, 217 ... 222
FORTRAN	50, 51, 73, 79... 83, 91
Forward declaration	245
Frameworks	1069, 1072, 1133...1136
Free storage	85
friend relationships	792...793, 794, 799, 974
fstream library	208 ... 215
Function call mechanism	253
Function declarations	243
Function definition	239
Function examples	259 ... 263, 290...
Function prototypes	136, 245, 271, 293, 495
Functions	237 ... 258
Functions' results	242
Generic code	660, 915
get pointer for a file	485
getline() function	307
Global data	43, 287, 294
good() (filestream function)	211
Hash table	513...517, 530, 618, 621, 761
Hashing	509...513
Header dependencies	745
Header files	100, 112, 244, 359, 480, 496, 978
Heap	101, 599, 600...605
Heap-based data	see Dynamic data
Heapsort	673
Hexadecimal	506, 611
Hierarchy	824, 966, 1016, 1024, 1029
Histogram (example program)	290...298
IDE	105...107
If statements	74, 77, 78, 95, 185 ... 189
ignore (cin.ignore() function)	184

Implementation (examples)	112, 134, 136, 195, 227, 271
implicit argument (this)	555
Including files	100
Inheritance	820...838, 839, 1068
Initializing variables	141
Initializing variables	see also constructors
inline functions	255, 480, 561
Input handling loop	182, 183
instance of class	547, 556
Instruction Register	6, 29
Instruction repertoire	7
Instructions	6, 7, 32
Integer data	76, 95, 126
Integer data, representation	12
Integer division	132
Interpreted languages	83
iomanip library	197
iostream	107, 108
IR	See Instruction Register
Iteration	39, 71, 145...170
Iterators	792...803
Job control language	51, 62, 66, 113
Libraries, subroutine (function)	42, 80, 124, 125, 136
Lifetimes of variables	599
limit.h	204
Linking a program	79, 80, 326
Linking-loader	80
Lisp	84
Lists	655...657, 659, 690...699
Loader	33, 80
Local data	43, 126, 253, 287
Local networks	65
Logical expressions	157
long int	126
Loop control variable	167
Loops	See iteration
Lvalue	130
Macros	1139
MacApp	1135
Macintosh OS	68
Magic numbers	139
main()	113, 124
Mangling of names	249
Maths library	136, 137
Matrix	see Array
Member functions	548, 553
Memory addresses	10, 486, 600, 648, 649
Memory leak	603, 817
Memory manager	600
Mnemonics	7
Modules	745, 976, 1042, 1126
Modulo	131

Multiple inheritance	835...838
Multiprogramming	53...55
Murphy's 1st law of programming	260
Naming rules and conventions	114, 138, 240, 460, 553
Nested functions	92
Nested loops	296, 297, 314
Networks	654
new operator	602, 604
new [] operator	604
Newton's square root (program example)	149...152
NULL	609
null character	304
null pointer	609
Numbers (multiprecision)	575...597
Objects	102, 547
Op-code	8, 32
open() (filestream function)	210, 213, 214
Operand	8
Operating System	6, 11, 21, 47...
Operator functions	802...808
Operator precedence	131, 159, 160
OS	see Operating System
Overflow	13, 132
Overloading of function names	250 ... 252, 559
Packet switching networks	65
Pages (computer memory)	11, 60
Parentheses in expressions	160
Pascal	94
Pascal case statment	96
Passing arguments	82, 137, 300, 645, 646
PC	See Program Counter
Peripheral device controller	3, 19, 24
Peripherals	16 ... 26
Persistent data	1148
pH (example program)	135
Pointer arithmetic	652
Pointers	486, 555, 600, 605...617, 645
Polymorphism	835, 1016
Priority queue	659, 667...679, 957
private implementation section of class	550, 551
Processes	61, 62
Profiler	448...450, 597
Program Counter	6, 29
Project	105, 117
Promoting (a node in a tree)	708, 849, 898
Property vectors	526
protected interface of class	
Prototypes (function)	see Function prototypes
Pseudo-code	112, 181, 456
public interface of class	550, 551
Pure virtual function	827, 968
put pointer for a file	485

Queue	659, 660...667, 932...935
Quicksort	432...440, 922, 923
RAM	9
rand() (stdlib function)	264, 976
Random access to record file	483
Random numbers	266
Real (floating point) data	76, 95, 126
Records	86, 87, 96
Recursion	87...
Recursive function	87, 257 ... 259, 421, 432, 703, 704, 709, 848, 849, 883
Redirection of i/o	208
Registers	5
Remainder	131
Repeat loop	145, 164
Reserved words	138
Resources (data structures)	1136
Return (in subroutine)	42, 42
return statement	239
Reusable components	102, 549, 1065
ROM	9
Round-off errors	14
Row	see Arrays multidimensional
Scenarios (program design)	729, 960, 1007
Scope	156
Scripts (for job control)	113
Searching an array	389, 420
Sector	see Block
Seek (operation on disk)	18
Segments	601, 602
Selection	40
sentinel data	192, 215
Sequence (instructions/statements)	29, 39, 71, 123...144
Sequential access to record file	483
Serial I/O devices	23
setfill() (iomanip function)	218
setw() (iomanip function)	218
Shell (job control)	67
short	126
Side effects with functions	237, 242
Simula	102
Simulation	102, 623, 951
Singleton pattern	1024
sizeof() operator function	321
sleep()	362
Smalltalk	102
Sorting	426, 429, 673
Splitting a tree node	871, 887...890
Spooling	61
Stable sorting mechanisms	427
Stack frame	91, 92, 253
Stacks	87, 91, 92, 118, 602, 798

Standard libraries	125
Statements	73, 127
static (class member)	789...792
static (type qualifier for variable)	288
Static data space (segment)	101, 288, 601
Streamability	see Persistent data
string.h	
Strings	304...310, 618
strlen() function	
struct	465...470, 475, 478, 526
Subroutine call instruction	41
Subroutines	40, 72
switch statement	173 ... 179
switch statement	see also Pascal case statement
Symantec	105, 278, 361, 1135
Symbol table	37
Symbolic computation	84
Syntax checking	116
Syntax rules	36, 72
System's code	47
Tab	129
Tapes	22
Template instantiation	918...921
Template classes	927...935
Template functions	916...927
Testing and test data	112, 197, 231, 571, 597, 664, 677, 687, 698, 719, 723, 774, 860...868, 912
Text files	208
this	556
throw	940
Timesharing	55...57
Top-down functional decomposition	451
Track (disk memory)	16
Tree "traversal"	719
Trees	see Binary tree, AVL tree, BTree, 2-3 tree
try	940
Twos complement numbers	12, 13
Type casts	607, 608
Type safe linkage	250
typedef	158, 318
uninitialized pointers	609, 610
union	470...473
Unix	67
Unsigned numbers	204
Validating function arguments	261
Variable definitions	114, 126, 156, 605
Vector	see Array
virtual functions	827
Virtual memory	58...61
virtual tables	834, 835
void	113, 242

void* (generic pointer)	606
void**	684
While statement	95,145...155
Windowrep class	1023
Window classes	1023
Words (computer memory)	9
Xerox research center	68