

#define Command	2:5
#endif Directives	2:13
#error Command	2:18
#if and #endif Directives	2:13
#if Directives	2:13
#ifndef Command	2:14
#include Command	2:16
#line Command	2:19
#pragma Command	2:20
A	
Accessing Structure Members with the Dot Operator	4:21
Allocating Memory	5:3
Argument Description	8:3
Arithmetic	4:7
Array Initialization	4:1
Arrays	1:16, 3:6
Arrays - Dynamically Sized	5:9
Arrays of Pointers	4:14
Assignment of Pointers and Address	4:8
Assignment and Arithmetic	4:7
B	
Binary	6:9
Bit-Fields	3:5
C	
C Library Functions - Frequently Used	7:2
C Programming Language - Features	1:1
C Programming Language - Program Structure	1:2
C Programming Language - Syntax	1:4
C String I/O Library Functions	7:7
C Stream Routines - Available	6:3
calloc() Routine	5:6
Casting	3:17
Character (char)	1:8
Character Processing	7:10
clearerr() Function	6:26
Command Line Arguments - Parsing	8:4
Command Line Processing - Customizing	8:7
Conditional Commands	2:13
Conditional Compilation	2:12
const Specifier Variable Type	3:21
D	
Data Type Demotion	3:19
Datatypes	1:8
Declarations	4:4
Double-Precision Floating Point (double)	1:13
Dynamic Memory Allocation - Strategy	5:10

E	
Enumerated Sets	3:2
External Variables - Visibility	3:30
F	
fclose()/fcloseall() Functions	6:15
feof() Function	6:24
ferror() Function	6:25
fflush() Function	6:17
File Handles - Predefined	6:8
File Pointers	6:10
File Structure	6:11
Files	6:1
Files and Streams	6:1
Floating Point (float)	1:11
fopen() Function	6:12
fread() Function	6:22
free() Function	5:7
freopen() Function	6:16
fseek() Function	6:18
ftell() Function	6:20
Function Breakdown	1:22
Functions	1:21
fwrite() Function	6:23
G	
getenv() Function	8:8
H	
Hashing Approach to Sparse Arrays	9:9
Header and Other Include Commands	2:15
I	
I/O Functions	6:2
Identifiers	1:5
Include Commands - Other	2:15
Integer (int)	1:9
L	
Library Functions - General Facilities	7:1
Lifetime/Visibility Specifiers	3:24
Linked List	5:12
Linked-List Approach to Sparse Arrays	9:3
Long Integer	1:10

M

Macro Expansion	2:2
Macro Substitution	2:8
Macros vs. Functions	2:7
Macros with Parameters	2:10
main() Function	1:23, 8:1
malloc() Routine	5:4
Manifest Constants	2:9
Memory Allocation Program	5:11
Memory/Buffer	7:11
Memory Usage During Program Execution	5:1
Message Pragma	2:22
Moving through a List	3:16
Multi-Dimensional Arrays	4:17

P

Parameters and Returns	4:23
Pointer Notations	4:5
Pointer and Strings	4:10
Pointer Strings Versus Array Strings	4:13
Pointers	1:15, 4:2, 4:9, 4:19
Pointers to Whatever	4:15
Pointers - Uses	4:3
Pointers to Functions	4:16
Precedence	4:9
Precedence and Pointers	4:9
Preprocessor Commands	2:4
Preprocessor Commands - Other	2:18
Preprocessor - Convention and Syntax	2:3
Preprocessor Directives	2:23
Preprocessor - Syntax	2:3
Program Stages	1:3
Program Statements	1:7
putenv() Function	8:9

R

Referencing Array Elements	4:22
Reserved Words	1:14
rewind() Function	6:21

S

Scope	3:29
Self-Referential Structures	3:14
Short Integer (short)	1:9
Simple Structure	3:9
Sparse Array Processing - Purpose and Function	9:1
Spreadsheets and Sparse Array Processing	9:2
Stacks	5:2
Storage	4:18
Storage Class Specifiers: auto	3:25
Storage Class Specifiers: register	3:28
Storage Class Specifiers: static	3:26
Storage Class Specifiers: extern	3:27

Storage Classes	3:23, 4:1
Storage Classes and Array Initialization	4:1
Stream - Opening	6:6
Stream Pointers - Predefined	6:7
Streams	6:1
String Function	4:11
String Processing	7:10
String Related Functions - Useful	7:12
Strings Initialized as Pointers	4:12
Structure Referencing	4:20
Structure Members - Referencing	3:13
Structure Type - Declaring	1:18, 3:10
Structures	1:17, 3:7, 4:19
Structures - Defining	1:19, 3:11
Structures - Initializing	1:20, 3:12
system() Function	8:12
<u>I</u>	
Text vs Binary	6:9
Text or Binary - Specifying	6:14
Type Conversion with Assignment Operator	3:18
Type Conversions and Casting	3:17
Typedefs	3:1
<u>U</u>	
#undef Command	2:11
Unions and Bit Usage	3:3
<u>V</u>	
Void Data Type	3:20
volatile Specifier Variable Type	3:22