USER MANUAL

UCSDPascal

THE CHOICE OF
PROFESSIONALS

PECAN

Identifiers 26
Definition 27
Uniqueness 27
Predeclared Identifiers 28
Comments 28
Token Boundaries 30
Indentation and Legibility 30

CHAPTER 5: DATA TYPES AND OPERATIONS 33
Introduction 33
Constants, Variables, and Expressions 33
Assignment 35
Evaluating Expressions 35
The Use of Data Types 36
Simple Data Types 37
Integer 37
Real 40
Long Integer 44
Boolean 46
Characters 47
Special–Purpose Types 48
Compatibility 49
Conversion Routines 49
Scalars and Subranges 51
Scalars 51
Subranges 52
Simple Types Declared as Scalars or Subranges 53
Scalar and Subrange Routines 53
Structured Types 54
Arrays 55
Strings 60
Sets 63
Records 66
Dynamic Types 70
Files 70
Pointers 74
Space Allocation for Data Types 78
Packed Data 78
Simple Types 79
Scalars and Subranges 81
Structured Types 81

The Sizeof Intrinsic 82

CHAPTER 6: OVERALL PROGRAM SYNTAX 85
The Outline of a Program 85
Program Heading 86
Label Declarations 86
Constant Declarations 87
Type Declarations 87
Variable Declarations 88
Routine Declarations 88
The Main Body 89
Structure and Scope 89
Limitations 91
Forward Declarations 91
Size Limits 92
Segment Routines 92
Units and Separate Compilations 92
The Uses Declaration 93
The Format of a Unit 93
Selective Uses 95
Initialization and Termination Code 98
The Implementation of Units 99

CHAPTER 7: PROCEDURES AND FUNCTIONS 101
Procedure and Function Declarations 101
Value Parameters 103
Variable Parameters 103
Conformant Arrays 104
Interface Conformant Arrays 107
Procedures and Functions as Parameters 108
Calling Procedures and Functions 110
Exit Code 111
The PMACHINE Intrinsic 112
External Routines 115

CHAPTER 8: CONTROL STATEMENTS 117
Compound Statements 117
Conditional Statements 118
The If Statement 118
The Case Statement 120

Repetition 121
The While Statement 122
The Repeat Statement 123
The For Statement 124
Branching 125
The Goto Statement 125
Exit 126
Halt 126

CHAPTER 9: INPUT AND OUTPUT 129
Textfile I/O 129
Read and Readln 129
Write and Writeln 131
Page 132
Handling External Files 133
Opening and Closing Files 133
Untyped Files 135
Power System I/O 137
Keyboard 137
Device I/O 138
The Time Procedure 140
Ioresult 140
Screen I/O 141

CHAPTER 10: MEMORY MANAGEMENT 143
Runtime Environment 143
Segmentation 144
Code Segments 144
Controlling Segment Residence 145
Free Space 146
Heap Management 147

CHAPTER 11: CONCURRENCY 149
Concurrent Execution 149
Processes 149
Initiating a Process 151
Process Synchronization 152
Semaphores 152
Signal and Wait 153
Interrupts 153

Pecan's UCSD Pascal development system ....

The Pascal chosen by knowledgable programmers.

One of the most requested languages, UCSD Pascal offers the ability to write structured programs, making them easy to write, maintain and understand. Pecan Software Systems gives you all these sophisticated features:

- o Native code generation
- o Separate compilation of units
- o Segmentation feature that allows dynamic code swapping: better than overlays.
- o Conditional compilation of parts of your program
- o Include file capabilities
- o Supports string data types and string manipulation routines
- o Extended precision and BCD arithmetic
- o Event-driven multitasking and process synchronization
- o Dynamic allocation of memory
- o Standard Pascal I/O and random access files
- o Conformant arrays
- o Procedural parameters
- o IEEE 64-bit floating point format with full 8087 support
- o Very compatible with Apple Pascal
- o Generates very compact code files
- o Supports up to 16MB of addressable memory for code space

Develop UCSD Pascal programs quickly and easily on most microcomputers, minicomputers (including IBM, Apple, Macintosh, Atari ST, Amiga, DEC, Zenith Kaypro, Stride and Tandy) and operating systems.

Works with Pecan's powerful, integrated development environment.

PECAN