**Exercise 4-1.** Write the function strindex(s,t) which returns the position of the rightmost occurrence of t in s, or -1 if there is none.

**Exercise 4-2.** Extend atof to handle scientific notation of the form

123.45e-6

where a floating-point number may be followed by e or E and an optionally signed exponent.

**Exercise 4-3.** Given the basic framework, it's straightforward to extend the calculator. Add the modulus (%) operator and provisions for negative numbers.

**Exercise 4-4.** Add the commands to print the top elements of the stack without popping, to duplicate it, and to swap the top two elements. Add a command to clear the stack.

**Exercise 4-5.** Add access to library functions like sin, exp, and pow. See <math.h> in [Appendix B, Section 4](file:///C:\Users\Valentin\AppData\Local\Temp\Rar$EXa1580.12686\tcpl\appb.html#sb.4).

**Exercise 4-6.** Add commands for handling variables. (It's easy to provide twenty-six variables with single-letter names.) Add a variable for the most recently printed value.

**Exercise 4-7.** Write a routine ungets(s) that will push back an entire string onto the input. Should ungets know about buf and bufp, or should it just use ungetch?

**Exercise 4-8.** Suppose that there will never be more than one character of pushback. Modify getch and ungetch accordingly.

**Exercise 4-9.** Our getch and ungetch do not handle a pushed-back EOF correctly. Decide what their properties ought to be if an EOF is pushed back, then implement your design.

**Exercise 4-10.** An alternate organization uses getline to read an entire input line; this makes getch and ungetch unnecessary. Revise the calculator to use this approach.

**Exercise 4-11.** Modify getop so that it doesn't need to use ungetch. Hint: use an internal static variable.

**Exercise 4-12.** Adapt the ideas of printd to write a recursive version of itoa; that is, convert an integer into a string by calling a recursive routine.

**Exercise 4-13.** Write a recursive version of the function reverse(s), which reverses the string s in place.

**Exercise 4-14.** Define a macro swap(t,x,y) that interchanges two arguments of type t. (Block structure will help.)