

## Programming Language – Small (PLS)

PLS is a lexically-scoped, block-structured language whose syntax and semantics are similar to those attributed to Pascal. The syntax of PLS is given in the attached Conway diagrams subject to these restrictions.

1. Blanks and end-of-line serve as separators. Other separators include, keywords, "id", and so forth.
2. Real constants are of the form xxx.yyy where the x and y components each contain at least one digit (i.e. .5 is not a legal real constant).
3. String literals can only be used in the "write" and "writeln" statements. String literals are delimited by the double quote (") character and cannot contain embedded double quotes. Strings cannot be broken across line boundaries.
4. Within a PLS program, identifiers are unique to 7 characters. When you generate 3-address code that includes identifiers, however, you MUST preface each identifier with a unique procedure designator, e.g. P01. This makes ALL identifiers appear UNIQUE to the runtime system regardless of scoping.

Note, the PLS identifier ABCDEFGH (P01ABCDEFGH) at program nesting level 1 is not the same as the PLS identifier ABCDEFGH (P02ABCDEFGH) declared at level 2. Consistent code references to each PLS variable MUST be maintained.

5. PLS includes the builtin functions EOF and EOLN.
6. PLS includes the builtin procedures READ, READLN, WRITE, and WRITELN. READ and WRITE must have exactly one expression as a parameter; READLN and WRITELN can have 0 or 1 parameters.

READLN skips to the next line *after* reading data for its argument. Similarly, WRITELN skips to the beginning of a new line *after* printing the value of its argument. For both functions, if no argument is present, then only the skips are performed.

7. The semantics of PLS correspond very closely to the semantics of the Pascal language. When in doubt, consult a text on the Pascal language or try it on GPC (GNU Pascal).
8. PLS does not have any comment facility, coercion is required between integers and reals, all file I/O is through standard INPUT and OUTPUT, and there are no file names or buffer variables in PLS

9. The keywords in PLS are:  
    program, procedure, var, integer, real, char, array, of, begin, end, if, then, else, while, do,  
    read, readln, write, writeln, and, or, eoln, eof, and not.
10. Implicitly, the subscript for all array variables starts at 1.
11. Consecutive mathematical operators are not valid in expressions, e.g.,  $4+-3$ , is invalid.
12. Integer and real constants can be at most 10 characters long.