

Re: New symbolic/numeric/dynamic/intuitive programming language

Source: <http://sci.tech-archive.net/Archive/sci.math/2008-02/msg03547.html>

- *From:* rem642b@xxxxxxxxxx (Robert Maas, see <http://tinyurl.com/uh3t>)
 - *Date:* Thu, 21 Feb 2008 12:41:45 -0800
-

From: "Mark Nudelman" <ma...@xxxxxxxxxxxxxxxxxxxxxxxxxxxx>
This seems like an instance of the COBOL school of language design, the philosophy being that the closer a program looks to ordinary English, the better it is.

COBOL actually had at least one good idea: PICTURE clauses. In COBOL's formative years, 80-column Hollerith cards, with all data laid out in fixed columns, were the standard for input and output. Even printed reports used fixed format. When a data layout was specified once, it could be used both to parse incoming data and to generate outgoing data in the same format. The modern equivalent for variable-length fields with nested structure would be BNF. (The more popular regular expressions, aren't powerful enough, IMO.) I haven't seen any programming language that directly used BNF for its standard way to specify input/output parsing/formatting, which I see as a deficiency in **all** programming languages to date.

So, here's a task for the "English-like" folks: Try to find an English-like syntax for expressing BNF, then incorporate it into a modern programming language. Here's a strawman starting idea: A WFF can be a 'p' or 'q' or 'r' or 's', or an 'N' followed by a WFF, or (a 'C' or "A" or 'K' or 'E') followed by two WFFs. (That idea is not original with me. Shake-a-WFF was a fun lively game!)

.