AI Magazine Volume 4 Number 1 (1983) (© AAAI)

## Research in Progress

# The Yale University Cognition And Programming Project

Stephen Slade

*Computer Science Department*
*Yale University*

THE COGNITION AND PROGRAMMING PROJECT (CAPP) in the Computer Science Department at Yale University is an interdisciplinary group exploring a wide range of issues in programming. The current personnel are:

- Dr. Elliot Soloway, Assistant Professor; Dr. Kate Ehrlich, Research Associate
- Lewis Johnson; Jeff Bonar; Valerie Abbott

Work is currently in progress on the following projects:

**What do experts/novices know about programming?**[1] The objective here is to develop a plan-based theory of programming knowledge and to test it empirically with novice and expert programmers. We are also using this theory to study the bugs and misconceptions of novice programmers.

**Cognitively appropriate programming language constructs.**[1] The design of programming languages has largely been determined by *computational* constraints. However, we have identified cases in which the underlying strategy required by a programming language construct is not the one people normally know or use, and we have shown empirically that they have significant difficulty in learning to use such constructs effectively. We have also shown that when the language construct agrees with people's natural problem solving strategies they can learn to use such constructs effectively. The implication is that language designers should be more sensitive to cognitive capabilities which people bring to programming — and that computing educators should be aware of the systematic misconceptions which arise due to *cognitively poor* programming language constructs.

**Cognitively-based complexity measures of programs.** Using our theory of programming plans, we are developing measures of program complexity that are based on the underlying mental effort needed to understand programs. This approach is in contrast to typical measures of program complexity which are sensitive to only surface features of programs.

**Procedurality/Non-procedurality in problem solving.**[2] Algebra word problems are notoriously difficult for students to solve. We have found, however, that students can correctly solve such problems when asked to write computer programs. Moreover, we have recently gathered evidence that the benefits of programming can even transfer, and aid programmers using algebra. The basis of this effect seems to lie in the procedurality inherent in programming, as contrasted with the non-procedurality usually associated with traditional algebra. Our work suggests that if programming languages are made more non-procedural

[1]This project is currently being funded by NSF RISE, under grant number SED-81-12403

[2]This project is currently being funded by NSF IST, under grant number IST-81-14840

then many individuals will not be able to acquire such languages; moreover, it appears beneficial from a problem solving perspective to teach procedural programming languages.

**Intelligent Programming Tutor.**[3] Complementing the empirical projects, we are actively engaged in building an AI-based tutoring system, PROUST, which can assist novice programmers as they are learning to program. The goal of this system is to identify *non-syntactic* bugs in a student's program, and then tutor the student with respect to the misconceptions underlying the bugs. This project serves as an excellent test bed for our evolving theory of programming knowledge, and it also serves to raise the next series of questions which a theory of programming must answer.

**Recent CAPP publications are listed below.**


Soloway, E., Bonar, J., Ehrlich, K. (1982) Cognitive Strategies and Looping Constructs: An Empirical Study, *Communications of the ACM,* in press.

Ehrlich, K. & Soloway, E. (1982) An Empirical Investigation of the Tacit Plan Knowledge in Programming, in *Human Factors in Computer Systems,* J. Thomas and M.L. Schneider (Eds.), Ablex Inc., in press.

Soloway, E., Ehrlich, K., Bonar, J., Greenspan, J. (1982) What Do Novices Know About Programming? In *Directions in Human-Computer Interactions,* B. Shneiderman and A. Badre, Eds., Ablex, Inc., in press.

Soloway, E., Ehrlich, K., (1982) Tacit Programming Knowledge, *Proceedings of Fourth Cognitive Science Conference,* Ann Arbor, Michigan.

Ehrlich, K , Soloway E., and Abbott V. (1982) Styles of Thinking: From Algebra Word Problems to Programming via Procedurality, *Proceedings of Fourth Cognitive Science Conference,* Ann Arbor, Michigan.

Soloway, E., Ehrlich, K., Bonar, J. (1982) The Good, The Bad, and The Ugly in Programming, American Educational Research Association Meeting, N.Y.C.

Soloway, E., Woolf, B., Rubin, E., Bonar, J. (1982) Overview of MENO-II: An Intelligent Programming Tutor, ADCIS Conference on Computer-Based Instructional Systems, Vancouver, B.C.

Bonar, J., Ehrlich, K., Soloway, E., Rubin, E. (1982) Collecting and Analyzing On-Line Protocols from Novice Programmers, *Behavioral Research Methods and Instrumentation,* in press.

Soloway, E., Woolf, B., Rubin, E., Bonar, J. (1982) MENO-II: An Intelligent Programming Tutor, *Journal of Computer-Based Instruction,* to appear.

Soloway, E., Lochhead, J., and Clement, J. (1982) Does Programming Really Enhance Problem Solving Ability? Some Positive Empirical Evidence, in *Computer Literacy,* R. Seidel (Ed.), Academic Press, NYC.

Soloway, E., Ehrlich, K., Bonar, J. (1982) Tapping Into Tacit Programming Knowledge, *Proceedings of the Conference on Human Factors in Computer Systems,* Gaithersburg, Md.

Soloway, E., Woolf, B., Barth, P., and Rubin, E. (1981) MENO-II: Catching Run-Time Errors in Novice's Pascal Programs, Proc. of IJCAI-81, Vancover, B.C,

Soloway, E., Bonar, J., Woolf, B., Barth, P., and Rubin, E. (1981) Cognition and Programming: Why Your Students Write Those Crazy Programs, *Proceedings of the National Education Computing Conference,* Texas.

Clement, J., Lochhead, J., and Soloway, E. (1980) Positive Effects of Computer Programming On Students' Understanding of Variables and Equations, in *Proceedings of the National ACM Conference,* Nashville.

Soloway, E., and Woolf, B. (1980) Problems, Plans, and Programs, in *Proceedings of Eleventh ACM Technical Symposium on Computer Science Education,* Kansas City.



[3]This project is currently being funded by the Army Research Institute, under grant number MDA903-80-0508