CONNECTING HIGH-LEVEL PROGRAMMING CONSTRUCTS TO ASSEMBLY LANGUAGE USING FRANCES

Tutorial Presentation

Kian L. Pokorny, Ph.D.
Division of Computing
McKendree University, 701 College Rd., Lebanon, IL 62254
618-537-6440
klpokorny@McKendree.edu
Tyler Sondag
Dept. of Computer Science
Iowa State University, 226 Atanasoff Hall
Ames, IA 50014
sondag@cs.iastate.edu
Hridesh Rajan
Dept. of Computer Science
Iowa State University, 226 Atanasoff Hall
Ames, IA 50014
hridesh@cs.iastate.edu

Central to computing is machine code generation. Upper level undergraduate students studying computing are quite familiar with high-level languages. Most undergraduate programs in computing begin with a course involving computer programming in a high-level language and as students progress through their studies they gain more experience with high-level languages. Not only is the machine code central to upper level courses in computer science, introductory programming texts address the issue of how high-level languages are translated to execute on the machine [3, 4].

In recent years, as the field of computing has expanded, courses in assembly language have been supplanted with more timely material [1]. In the most recent ACM curricula guidelines assembly language is only mentioned in one of the fourteen sections, Architecture and Organization [2]. One of the ten subsections herein has nine hours recommended and there are nine learning objectives for this subsection. The ninth learning objective is: “Understand how, at the assembly language level, how parameters are passed to subroutines and how local workplace is created and accessed” [2]. Courses in computer architecture, operating systems and compiler design revolve around machine language yet the disconnect that exists from the lack of a course dedicated to the topic of assembly language introduces a gulf in knowledge. This makes it difficult for students to quickly grasp the concepts of these other courses and requires the instructor to spend time with the topic of assembly language.

Frances [5] is a new tool that shows graphically the relationship between statements in a high-level language and the generated assembly language. This tool allows students to quickly make a connection between the written high-level language and the generated assembly instructions. The environment has several unique features that make it useful to students needing to gain an understanding of this relationship.

Computing faculty teaching courses at all levels will be interested in the potential for classroom use. The presentation will provide instruction on the use of the tool, information about supporting material and access to the tool for future use in the classroom.

REFERENCES

[1] Agarwal, K.K., Agarwal, A., *Do we need a separate assembly language programming course?, Journal of Computing Sciences in Colleges*, 19, (4), 246-251, 2004.

[2] *Computer Science Curriculum 2008: An Interim Revision of CS2001, December 2008*. Joint Task Force on Computing Curricula, IEEE Computer Society, Association for Computing Machinery, 2008.

[3] Deitel, P.J., Deitel, H.M., *C++ How to Program 6 ed,* Upper Saddle River, New Jersey: Pearson's Education Inc., 2008.

[4] Gaddis, T., *Starting out with Control Structures Through Objects 6 ed,* Upper Saddle River: Pearson's Education Inc., New Jersey, 2009.

[5]Sondag, T., Pokorny, K.L., Rajan, H., 2010. Frances: A tool for understanding code generation, Proceedings *of the 41st SIGCSE technical symposium on Computer science education, Milwaukee, Wisconsin, USA*, March 10-13, 2010.