

# **A Survey of Computational Paradigms in Undergraduate Mechanical Engineering Education in the ASEE SE Region**

*B. K. Hodge, Ph.D., P.E.<sup>1</sup>*

## **Abstract**

The thirty-nine ASEE SE Section schools with mechanical engineering programs were surveyed to determine the usage of programming languages (such as FORTRAN) versus the use of arithmetic systems (such as MathCad or Matlab). Web sites and catalogs were used to determine the following: (1) programming courses required, (2) use of programming in curricula, (3) use of arithmetic systems in curricula, and (4) junior-level analysis courses required. The survey showed that about two thirds required at least one course in programming, that about one half made extensive use of programming languages, that about one half used arithmetic systems in significant fashions, and that about one half required junior-level engineering analysis courses.

## **Background**

At some point in any engineering endeavor calculations must be made and “numbers” generated. The manner of doing calculations in the engineering workplace and in engineering education has continuously evolved, especially since World War II. Prior to that time engineering calculations were accomplished in a completely manual fashion using mechanical calculators, slide rules, log tables, and nomograms. In the eighteenth century, addition, subtraction, multiplication, division, and log tables were laboriously used. The situation was greatly improved by the slide rule [the first useful one was introduced about 1850 (1)] and the mechanical calculator, but generating numbers was, until after World War II, still a labor-intensive undertaking. Feynman’s (2) anecdotal account of neutron diffusion calculations at Los Alamos in the 1940’s is a good example of the drudgery and tediousness of extended pre-computer calculations. The digital computer fundamentally altered the use of “manual” calculations and replaced it with machine-based computations. Initial efforts were hard-wired (literally) with patch boards, but by the early 1950’s higher-level programming languages evolved. For engineering computations, FORTRAN became the dominant programming language. However, as these advances were taking place, both the engineering workplace and engineering education struggled to effectively utilize the promise of the “computer” and to define the relationship between the computations and engineering. Indeed, one could argue that these struggles are ongoing.

The situation is much clearer today than a decade ago and much progress has been made in effective utilization and in the computer/engineering symbiosis. Hodge (3) identified four computational modes in the long-term evolution of Energy Systems Design (ESD), a specific mechanical engineering course at Mississippi State University. The course started out as hand-held calculator based (1981-1984) but quickly evolved into a course requiring extensive FORTRAN or BASIC programming to solve problems. During the late 1980’s and early 1990’s, the assignments moved from programming based to software application/modification based. However, by the mid-1990’s, arithmetic systems such as MathCad, Matlab, Mathematica, and EES (Engineering Equation

---

<sup>1</sup> Mechanical Engineering Department, Mississippi State University, Mississippi State, MS.

Solver) had been developed to levels that were very congruent with problem formulation and offered significant computational and visualization enhancements. Hodge's experiences with integrating MathCad as the arithmetic engine in the ESD course are described in Hodge and Taylor (4). The use of MathCad permitted much more realistic problems to be assigned and allowed the students to focus on problem formulation and solution (engineering tasks), rather than on programming issues. To some extent these four different modes of accomplishing arithmetic parallel the evolution of computational paradigms in engineering and engineering education.

Certainly, two dominant influences in the engineering workplace and in engineering education the latter part of the twentieth century were the ever-increasing capability of digital computers and the ever-increasing utility of software. Indeed, engineering education has struggled to integrate and accommodate the evolving hardware and software into curriculums. The pre-EC2000 ABET accreditation criteria for a number of years specifically addressed the integration of computers into engineering curricula. Until the advent of arithmetic systems, such integration meant devoting time and effort to structured programming in higher-level languages (FORTRAN for example). Thus, the use of arithmetic systems such as MathCad represents a more fruitful path that seems to portend the future. Indeed, as Baker (5) points out, "...the days of amateur programming are over." That is to say, except for highly skilled engineering specialists with post-BS degree education, engineers are not likely to do much programming in their careers. Note that this does not imply that engineers won't use computers, only that applications, not programming, will be the engineering tasks.

Conversations with other mechanical engineering educators have confirmed that the debate between programming and arithmetic systems in engineering education is a topic of considerable interests and that a number of mechanical engineering programs are making, or are considering making, significant changes in their curricula in this arena. A survey of ASEE SE Section mechanical engineering programs concerning the shifting computational paradigms is thus warranted.

### **Survey**

The survey is based on information provided on mechanical engineering departments' home pages on the World Wide Web (WWW) and/or in college and university catalogs or bulletins. Virtually all ME programs have home pages that present such diverse information as curricula, course descriptions, and educational goals. The ASEE SE Region contains 39 mechanical engineering programs. Appendix A presents a list of the ASEE SE Section schools with mechanical engineering programs. Web sites and/or catalogs (predominately web sites) were consulted to obtain information about all 39 mechanical engineering programs. Issues of interested include:

1. Programming courses (FORTRAN, C++, ....) required.
  - a. Taught by computer science (CS)
  - b. Taught by an engineering department
2. Indications of programming language usage.
3. Indications of arithmetic software usage.
4. The presence of a junior-level "analysis" course
5. Departmental goals involving computer usage.

Information pertaining to the above is usually included in web sites or catalogs. However, specific details of the interplay between programming and arithmetic systems in a given curriculum may or may not be clearly addressed in either a web site or a catalog. Many times the department's goals are useful in determining the roles played by programming or arithmetic systems. In several instances the available information did not permit the determination of programming versus arithmetic system utilization.

The survey was more time consuming than originally envisioned because of difficulties in finding information on the web sites. The following observations related to web sites were made:

1. Web sites differ dramatically in structure.

- The curriculum may be hard to find.  
Course descriptions may be hard to find.
2. Web sites may not be current.  
Some departmental web sites were unchanged since 1998.
  3. Web sites with online catalogs were the most useful.
  4. Some web sites contain only very general information.

### **Survey Results**

Tabulations, rather than lists of schools, will be the mode of presentation. The results of the survey are as follows:

1. Curricula with higher-level programming courses:  
27/39 require a higher-level programming course  
13 are taught by computer science departments  
14 are taught by engineering departments
2. Higher-level programming used in several courses:  
18/39 make extensive use of programming in major courses  
7/39 are unclear about the use of programming in major courses
3. Arithmetic systems used in several courses:  
16/39 make extensive use of arithmetic systems in major courses  
7/39 are unclear about arithmetic systems usage
4. Curricula with junior-level applied analysis courses:  
17/39 have computer-based analysis courses  
a mix of arithmetic systems/programming used  
several curricula have differential equations as a junior course (not counted as an applied analysis course)

### **Conclusions**

The results of the survey of 39 curricula indicate significant usage of both programming and arithmetic systems by various schools. Information from virtually all mechanical engineering programs indicated computer usage, but details were scant for some web sites and catalogs. However, arithmetic systems have been tenable only about six years, so the penetration of arithmetic systems into curricula is really significant. To this author, the most likely scenario is that arithmetic systems such as MathCad, Matlab, and EES will improve over time and continue to displace programming in mechanical engineering undergraduate programs. Many of these arithmetic systems are becoming highly integrated with word processor, plotting, symbolic manipulation, and computational capabilities. This level of integration will likely lead to more acceptance in engineering education as well as the engineering work place.

As pointed out by several reviewers (and as already concluded by the author), the use of catalogs and web sites to ascertain the usage of programming vis-à-vis arithmetic systems is inexact at best. A more accurate approach would be to send out a survey form to ASEE SE ME programs. Plans are underway to initiate such a survey.

### **References**

Kaplan, E., (1998) "Calculating Machines," [WWW.webcom.com/calc/](http://WWW.webcom.com/calc/).

Feynman, R. P. and Leighton, R., (1991), *Surely You're Joking, Mr. Feynman!*, QPBC, Camp Hill, PA.

Hodge, B. K., (1998), "The Evolution of a Required Energy Systems Design Course," ASME Paper 98-WA/DE-9, presented at the 1998 IMECE, Anaheim, CA, November.

Hodge, B. K., and Taylor, R. P., (1998), "The Impact of MathCad in an Energy Systems Design Course," Proceedings of the 1998 ASEE Annual Conference and Exhibition, Session 2266, Seattle, WA, June.

Baker, A. J., Pionke, C. D., Taylor, M. J., and Luttrell, B., (1996), "FEM Brought to the Academic Engineering Desktop," Proceedings of the 1996 ASEE Southeastern Section Meeting, Gatlinburg, TN, April.

### **B. K. Hodge**

Dr. Hodge received his aerospace engineering BS and MS degrees from Mississippi State University (MSU) and his mechanical engineering MS and PhD from the University of Alabama. He has worked for Thiokol Corporation and Sverdrup (AEDC). Currently he is a Giles Distinguished Professor, a Hearin-Hess Professor of Engineering, and Professor of Mechanical Engineering at MSU where he teaches graduate and undergraduate courses in the thermal sciences and conducts related research. In 1993 he was named as one of four Master Teachers at MSU. Since joining the faculty Dr. Hodge has developed six new courses and written two textbooks. He serves as the undergraduate coordinator for the mechanical engineering program at MSU and as director for the MSU Industrial Assessment Center. He is the 1999-2000 President of the ASEE SE Section.

## **APPENDIX A**

### List of ASEE SE Section Schools with Mechanical Engineering Programs

Auburn  
Christian Brothers  
Clemson  
Duke  
Florida Atlantic  
Florida Institute of Technology  
Florida International  
FAMU/FSU  
Georgia Tech  
Louisville  
Memphis  
Mercer  
Miami  
Mississippi State  
New Orleans  
North Carolina A&T  
North Carolina State  
Old Dominion  
Tennessee State  
Tennessee Tech  
Tuskegee  
Tulane  
University of Alabama  
University of Alabama (Birmingham)  
University of Alabama (Huntsville)  
University of Central Florida  
University of Florida  
University of Kentucky  
University of Mississippi  
University of North Carolina (Charlotte)  
University of South Alabama  
University of South Carolina  
University of South Florida  
University of Virginia  
University of Tennessee  
University of Tennessee (Chattanooga)  
Vanderbilt  
Virginia Tech  
VMI