LIFE-LONG LEARNING – MAKING DISCRETE MATH RELEVANT FOR INFORMATION SYSTEMS PROFESSIONALS

Dr. David F. Wood, Robert Morris University, wood@rmu.edu
Dr. Valerie J. Harvey, Robert Morris University, harvey@rmu.edu
Dr. Frederick G. Kohun, Robert Morris University, kohun@rmu.edu

ABSTRACT

This paper offers practical information and experience on how to design and implement a discrete mathematics course that can be appreciated and used by information systems professionals throughout their work life. Materials and on-line resources and activities are targeted to foster motivation and confidence for students as well as understanding of how the concepts presented serve in learning and will serve them in career settings. The technological and societal relevance of discrete mathematics concepts in the IS curriculum is covered. A matrix correlates each local ABET-accreditable core curriculum with a standard set of discrete mathematics topics to derive relevant topic coverage.

Keywords: Discrete Math, ABET, IS Curriculum

ORIENTATION TO RATIONALE

In our experience developing the curriculum for information systems students at Robert Morris University (RMU), we sought to anticipate the challenges that our IS graduates face now and will face in the future. At the same time, ABET accreditation standards included more mathematics than our previous curriculum had included [1]. As information technologies have grown more complex and information systems more integrated, discrete mathematics provides a formal foundation in understanding the methods in modeling, analysis, specification, design, verification, development, and documentation of information systems. Discrete mathematics offers a formal foundation for concepts and operations covered in core courses of an IS curriculum. Certain discrete mathematics topics provide practical support for problem solving in various programming courses, as well as operating systems, network and data communication, database, and project management.

The interest in incorporating Discrete Mathematics in an IS departmental course came from two major sources. First, we wished a way to give our students a set of general tools and conceptualization methods that are lasting, allowing them to abstract from single specific principles and practices that become rapidly obsolete. Secondly, our students had particular difficulty with Discrete Mathematics as taught in the Mathematics Department due to its total abstraction and removal from any connection with their Information Systems courses, and potential career goals in Information Systems. The math course typically included no computer science or information systems examples, relying on mostly notational problems and solutions. Many Information Systems students had difficulty appreciating the relevance of discrete math to information systems, and were unable to bridge the gap between math theory and IS practice.

Our efforts were to make mathematics and its ability to formulate problems something that students could rely on not only in their undergraduate courses, but also when exposed to job-related problems in the future. By recognizing the underlying patterns and problems, and by remembering their introduction to the subject, discrete mathematics can be a part of their life-long ability to formulate and solve problems.

At RMU, three departments worked together to produce the curriculum recommendations in this paper: Computer and Information Systems, Mathematics, and Engineering (which has a Software Engineering program). An interdisciplinary committee was formed to design the course and materials. This approach has been endorsed in the recommendations of the Committee on the Undergraduate Program in mathematics of the Mathematical Association of America [7].

INTRODUCTION TO MAIN TEACHING METHODOLOGIES

One of the guiding principles of the interdisciplinary committee designing this course included making learning easier and enjoyable and the increase of student confidence. A second principle was that of repeating the same concept in multiple forms, thus reinforcing the abstract principle involved, and facilitating the recognition of patterns in problems. For example, logic and truth tables in propositional logic were compared to predicate logic and Boolean logic. Graph theory was related to circuit design, Geographic Information Systems, and computer network topologies.

OVERVIEW OF DISCRETE MATHEMATICS CURRICULUM TOPICS

Before the committee first began to design the course, the Computer and Information Systems Department created a matrix correlating discrete mathematics topics and IS applications covered in core curriculum courses. The traditional topics of discrete math as taught for mathematics, computer science, and software engineering constituted one dimension of the matrix. The other dimension recognized the core curriculum courses in an ABET-conformant IS program. In each cell of the matrix were listed all of the intersection points pf a given discrete mathematics topic with the topics covered in the relevant course.

Discrete math topics that were not relevant to any of the courses were omitted from consideration in the new course design. The resulting course is able to concentrate on topics with IS applications in a thorough and applied manner, while omitting those whose applicability were more distant.

The resulting list of topics that remained in the new course includes Logic, sets, sequences and strings, number systems and representation of numbers, relations, functions and operators, randomization, permutations, and combinations, relational calculus concepts in database, algorithms, codes, encryption, and compression, graphs and trees, automata and pattern matching and the formal documentation of computer languages. The matrix is presented in Table 1 in reduced form:

Table 1. Correlation of Discrete Mathematics Topics and IS Applications Covered In Core Curriculum Courses

Topic	Programming / Logic	Operating Systems	Network/ Data Communications	Database Management	Project Management
Logic, sets, sequences and strings	PR H Empty Set, Union Intersection, Disjoint Set, Difference, Venn Diagrams, Universe,	NOT AND OR XOR		Cartesian Product SQL Joins etc.	
number systems and representation of numbers		Binary Data Representation and Hex Notation		Domain Range Partial Orders	
relations, functions and operators	VB.Net functions, Java Methods				
randomization, permutations, and combinations			Encryption		Testing and Quality Control
relational calculus concepts in database				n-ary attributes key queries selection projection join, Relational Algebra	
algorithms, codes, encryption, and compression	Data structures assignments		Asymmetric Public Key Encryption		
graphs and trees			Representation of data communications networks		PERT charts
automata and pattern matching	M programming				
formal documentation of computer languages.	BNF specifications				

There were questions about whether formal logic notation should be used in a course intended for information systems majors. A careful analysis revealed that internationally accepted formal notation made the material clearer and easier to learn. Much effort involves understanding how logical structures are communicated in everyday natural language. Ambiguities in natural language, as in specifications for applications, systems, and projects, must be addressed. Formal notation is of great practical help in this process of learning about the meaning of specifications.

PRACTICAL EXAMPLES DRAWN FROM COVERAGE

The interdisciplinary committee created a custom published textbook which contains practical examples used in IS fields. Examples from cell phone technology, database design, construction, and querying were used. Spanning tree practice helps students understand how non-redundant bridging can be set up or multicasting supported in Internet technology. Venn Diagrams are used to document wireless protocols. Graph and tree examples come from networking and other application domains.

SELECTING AND ADAPTING MATERIALS

While there are many popular textbooks for discrete mathematics, those focusing on computer technology are generally directed to a computer science audience [2, 3, 4]. Instead, the committee started with a base text [6] and developed a supplement by having each member of the committee submit exercises and examples relevant to Information Systems [5].

Software resources available for on-line use without charge or for favorable academic licensing were extensively researched. Software for digital logic, predicate logic using the “Tarski’s World” paradigm, and Prolog were used. Students access a web page with links to many discrete math software sites and interactive exercises. The course is highly interactive, with students weekly emailing specific example assignments.

SUMMARY AND REVIEW

The course described here sets a priority on discrete mathematics topics that are the most important and valuable for our students. It gains from clear connections to applications in the core courses of the undergraduate curriculum. It helps students recall, adapt, and reuse the formal knowledge they bring with them to this course and the entire curriculum. It provides students with opportunities to develop and demonstrate practical examples of how the concepts being treated are used in information systems.

Appropriate discrete mathematics instruction enhances life long information systems education by providing a sound formal foundation for and insights into an increasingly complex information technology, helping students recognize and use practical problem-solving tools, and giving information systems professionals insights useful in making management decisions with regard to applications of formal structures in business.

REFERENCES

1. Computing Accreditation Commission, Accreditation Board for Engineering and Technology, Inc. (2003). Criteria for Accrediting Information Systems Programs (Effective

for Evaluations during the 2004-2005 Accreditation Cycle). Baltimore, MD.: Accreditation Board for Engineering and Technology, Inc.
2. Epp, S. S. (2004). *Discrete Mathematics with Applications*, (3rd ed.). Brooks/Cole.
3. Gersting, J.L. (2003). *Mathematical Structures for Computer Science: A Modern Treatment of Discrete Mathematics,* (5th ed.). New York. W. H. Freeman.
4. Grassmann, W. K. &Tremblay, J.P. (1996). *Logic and Discrete Mathematics: A Computer Science Perspective*. New York. Prentice Hall.
5. Harvey, V. J., Harris, B., Holdan, E. G., Maxwell, M. M. & Wood, D. F., eds. (2003). *Discrete Mathematics Applications for Information Systems Professionals*. New York. Pearson.
6. Johnsonbaugh, R.. (2001). Discrete Mathematics (5th ed.). New York. Prentice Hall.
7. Mathematical Association of America. (2004). *CUPM Curriculum Guide 2004: Undergraduate Programs and Courses in the Mathematical Sciences*. Washington D.C. Mathematical Association of America.