

**INFORMS Computing Society**  
**Education Committee**  
**Initial Report Draft**  
**November 2007**

**Jill Hardin, Chair**

**Kevin Furman**

**Allen Holder**

**David Rader**

**Cesar Rego**

# 1. Introduction

The ICS Education Committee was charged in March 2007 with outlining appropriate curricula for undergraduate students wishing to pursue graduate or industry work at the OR/CS interface. The committee members are

- Jill Hardin, Chair, Virginia Commonwealth University, jrhardin@vcu.edu
- Kevin Furman, ExxonMobil, kevin.c.furman@exxonmobil.com
- Allen holder, Trinity University, aholder@trinity.edu
- David Rader, Rose-Hulman Institute of Technology, David.Rader@rose-hulman.edu
- Cesar Rego, University of Mississippi, crego@mit.edu

The members of the Committee represent diversity in department affiliation as well as in institution size and mission. The Committee was also privileged to have the counsel of two advisors: Harvey Greenberg (University of Colorado at Denver) and Ariela Sofer (George Mason University).

What is presented here is merely a starting point for discussion. It is the Committee's intention to actively solicit feedback from the ICS membership at large, and it is our earnest hope that the membership will actively engage in dialogue around the issues presented here.

## 2. Process, Objectives, and Goals

### 2.1 Objectives

As previously stated, the Committee's charge was to outline appropriate curricula for undergraduate students planning to pursue work at the OR/CS interface. Given the diversity of programs offering courses in operations research, and given the diversity of curricula across institutions in general, the Committee determined that the best approach is to outline a list of skills, rather than a list of courses, that would prepare students for work at the OR/CS interface. The Committee further determined that skills should be separated into three tiers:

- Tier One: skills deemed essential for future success at the OR/CS interface;
- Tier Two: skills deemed important for future success, but not essential;
- Tier Three: skills deemed helpful in work at the OR/CS interface, but not necessary for success.

After extensive discussion, the list was further refined into a list of skills and a set of "awarenesses". For example, we believe it is important for students to have a sense of the difference between solutions obtained analytically and those obtained computationally, but we do not expect students to be well-versed in numerical techniques.

It is important to note that the Committee's aim is *not* to make curriculum recommendations for undergraduate departments and programs. While we hope that these recommendations will be taken into consideration when advising students and designing curricula, we respect the autonomy of individual departments, programs, and institutions. Our primary aim is to help undergraduate students navigate the many elective course options available to them at their respective institutions as they seek to be well-prepared for future graduate and professional work at the OR/CS interface. We have sought to take into consideration the diversity among students who choose to pursue such a path, considering specifically students from programs in business, computer science, industrial engineering, mathematics, and operations research. As such, we have sought to keep the list of Tier One skills and awarenesses as small as possible while still encouraging preparedness.

## 2.2 Process Description

To develop the Tier One set of skills and awarenesses, the Committee members individually developed lists of skills deemed to be essential. Contents of these lists, especially where they differed, were discussed via several conference calls and intermediate email discussions. It should be noted that Tier One as presented in this document was still under discussion until the time of writing; this is a testament to the challenging nature of the task at hand. Tier Two and Tier Three are also included in this document as a combined list. However it should be noted that these have been subject to little careful discussion; they will be subject to further scrutiny—including separating Tier Two from Tier Three—as the Committee moves forward in 2008.

The Committee also felt it was important to put these discussions in context through a variety of investigations. Investigations were undertaken of industrial job postings, graduate program admissions requirements, and curriculum recommendations developed by related professional organizations (e.g. ACM, SIAM, etc.) Individual summaries of these investigations are included later in this report. These studies, however, do support the Tier One skills and awarenesses as presented.

We wish to reiterate that the recommendations provided here are simply a starting point for discussion. While the Committee has made every attempt to ensure their relevance and usefulness, we are very interested in feedback and discussion from the ICS membership at large. We encourage ICS members who wish to comment to contact any of the committee members at any time. Additionally, members will have two formal opportunities to provide feedback at the INFORMS 2007 Annual Meeting in Seattle:

- **ICS Education Committee Panel Discussion**, Sunday, November 4, 4:30-6:00 p.m. A Committee member will present an overview of our process and recommendations. The majority of the session time will be reserved for discussion.
- **ICS Business Meeting**, Monday, November 5, 6:15-7:15 p.m. A brief overview of the Committee's work will be presented, and Committee members will be available to receive feedback.

In addition to these feedback opportunities in Seattle, the Committee plans to conduct a survey in late 2007 or early 2008 to actively solicit feedback from ICS members. We urge members to take advantage of these opportunities to provide input and to shape the future of education in OR/CS.

### **3. Tier One**

#### **3.1 Skills and Awarenesses**

##### **Skills**

*These skills are highly recommended before pursuit of industry job or graduate study at the OR/CS interface.*

##### ***Fundamental Mathematical Competencies***

- Calculus
  - Differentiation
  - Integration
  - Sequences and series
  - Multivariate
- Linear Algebra
  - Systems of linear equations – finding solutions and understanding the structure of solutions (e.g., Gaussian elimination, linear independence of vectors, nullspace, etc.)
- Discrete Mathematics
  - Sets and set operations
  - Logical expressions
  - Basic graph theory
  - Functions and relations
- Probability & Statistics
  - Basics of probability
  - Random variables, expected value, variance
  - Sampling distributions, Central Limit Theorem
  - Hypothesis testing of means and sample proportions
  - Confidence intervals for means and sample proportions
  - Introduction to simple linear regression

##### ***Operations Research Modeling***

- Ability to create a mathematical model describing a particular situation or application
- Ability to separate model from data
- Know methods and software available to solve models
- Ability to interpret, analyze, and communicate model solutions

### ***Technical Writing and Presentation***

- Ability to organize and support analysis and convincing arguments (e.g. mathematical proofs, statistical analysis)

### ***Basic Computing Competencies***

- Familiarity with Contemporary Operating Systems (e.g. Windows, Unix, etc.)
- Experience using general scientific software (e.g. Matlab, Maple, SPSS, SAS, etc.)
- Experience using spreadsheet and other Office-type packages

### ***First-year Programming Competencies***

- Fluency in a compiled computer programming language (e.g. C, C++, Java)
- Familiarity with scripting languages (e.g. Python, VBA, optimization modeling languages)
- Familiarity with common data structures (e.g. linked lists, stacks, queues, heaps, trees)
- Ability to analyze run-time of basic algorithms

## **Awarenesses**

*These are concepts that a student is highly recommended to understand and of whose significance they should be aware before pursuit of industry job or graduate study at the OR/CS interface.*

- Numerical Analysis
  - Difference between computed and analytical solutions
  - Awareness of error analysis
    - Round-off error
    - Errors in approximation techniques
- Operations Research Modeling
  - Familiarity with various model types and their applications
    - Mathematical programming models
    - Simulation models
    - Analytical probability models
  - Ability to choose appropriate model type for a given application (e.g. understand the difference between capabilities of simulation and mathematical programming).
  - Determine whether exact or heuristic methods are most appropriate in solving a model

## 3.2 Justification

Operations Research and Computer Science use the tools of mathematics to solve and analyze problems, and students who lack a lucid appreciation for the discipline will be limited in their ability to understand and advance the OR/CS interface. With this in mind, Tier One begins with a list of suggested mathematical skills, each of which is a supportive pillar of OR/CS. The list is not intended to be exhaustive, indeed, if all serviceable areas of mathematics were listed, then the entirety of mathematics would appear. Instead, the mathematical skills of Tier One are considered to be a core from which a student can progress to a more advanced study.

The Calculus is one of the most impressive accomplishments of humankind, and the primary topics of differentiation, integration, and convergence are paramount to OR/CS. The totality of examples is too immense to state succinctly, but without differentiation one could not model a differential equation, could not capture the essence of rate, and could not state the standard optimality conditions; without integration, one could not define a continuous probability, could not analytically solve a differential equation, and could not calculate simple statistics; and without convergence one could not analyze algorithms. Beyond these and other applications within OR/CS, the study of calculus is typically a student's first step toward mathematical rigor and maturity, both of which are required to understand and apply the fields of OR and CS.

Topics in linear algebra and discrete mathematics are similarly crucial to the development of OR/CS. In particular, the ability to solve and analyze a system of equations or inequalities, together with the geometric insight provided by this analysis, motivates many topics in OR. In addition to the basic logic that underlies much of discrete mathematics, the sub-disciplines of graph theory and combinatorics provide methods to define relationships and to model and exploit structure, all of which are important tools in OR and CS.

A taxonomy of OR would likely split the discipline into deterministic and stochastic realms, and from this perspective, an introduction to probability and statistics represents one of the two major threads in OR. Moreover, the ability to present and analyze data statistically provides a student with intellectual dexterity as she or he confronts the inherent randomness of many applications. As such, a thorough study of introductory probability and statistics is necessary of any student working within OR/CS.

Although at the undergraduate level, it should not be expected that all OR/CS students would be proficient OR modelers, at least some familiarity with various model types is absolutely necessary to interact in a meaningful way at the OR/CS interface. The basic understanding of the independence of OR models from their data is perhaps one of the most fundamental concepts in OR. Given a familiarity of various model types, an essential skill for any OR practitioner is the ability to match an appropriate model type after obtaining an understanding of some academic or business problem or application.

Obviously, the ability to at least use some basic software or methods to solve a given model is of practical importance at the OR/CS interface in order to put the given modeling knowledge to some use. It would not be expected that the student necessarily understand the methods, but would be able to implement a given model and call a generic solver or calculation system to obtain solutions. Lastly, the ability to understand and analyze the results of an OR problem

solution and communicating it to others is necessary to interact with others and explain any research or conclusions to be drawn.

A student at the OR/CS interface needs to have a basic computing skill set such that he/she can enter a position and be able to smoothly transition into the computing environment of his graduate advisor or new job – which could be one of multiple operating systems including Windows, Linux, UNIX, OSX, etc. Although MS Windows is very common, some familiarity with UNIX/Linux commands would probably be invaluable in the event the new environment is not Windows-based. Further, early on when testing and evaluating ideas, it is highly likely that scientific software is available, and perhaps required. Experience with at least one package significantly improves the learning curve to learn any scientific software package.

For working at the OR/CS interface, it is relatively obvious that someone will need at least some computer programming skills. Fluency in at least one programming language is generally sufficient to substantially hasten the ability to learn other languages. The basic concepts of data structures are universal and highly valuable for anyone doing any programming. Further, the basic ability to analyze an algorithm is essential for any programmer when trying to improve the speed and efficiency of some code, which is usually very important to OR endeavors.

From the industrial point of view, when hiring any candidate for a position that would involve working at the OR/CS interface, the most important attribute would be to have not only a strong skill set, but also the potential to easily and quickly learn any new skills that are necessary. From the mathematical perspective, having a balanced breadth and depth in various topics is probably the best combination. At the level of OR modeling, it is probably better for a student to have a very wide breadth in different kinds of models and solution methods, such that when approaching a new problem a good approach is attempted early on in the process. Learning more about a particular topic can be done over the course of a project, but this is not going to happen when someone is unaware of the topic in the first place. When a topical expert without much breadth has a “hammer” in their toolbox, every problem tends to look like a “nail” to them. In terms of computing competencies, the rule of thumb is that more is better from an industrial perspective. It is typically harder to come up to speed in industry from the computing standpoint – a good skill set is usually assumed. Further, those with weaker computing skills would probably find the competition with their peers very harsh. Generally scientific software is more commonly used and prevalent in industry for looking at OR problems than programming applications from scratch, with the probable exception of the software industry. However, computer programming is still a necessary tool, and it is very hard to become an efficient, structured and clean programmer if one does not learn to do things that way from the start.

## **4. Tier Two and Tier Three**

### **4.1 Skills and Awarenesses**

As previously noted, these skills and awarenesses have not been subject to careful scrutiny and discussion. This will take place as the Committee proceeds with work in 2008.

### ***Mathematical Competencies:***

- Differential equations
  - ODEs
  - PDEs
  - solution methods
  
- Advanced/Numerical Linear Algebra
  - conditioning and stability
  - SVD
  - QR factorization
  - Gaussian elimination
  - Cholesky factorization
  
- Numerical analysis / scientific computing
  - floating point calculations
  - numerical integration
  - numerical solution of differential equations
  
- Real analysis
- More advanced graph theory
- Knowledge of mathematical proof writing techniques

### ***Operations Research Theory, Modeling & Algorithms:***

- Understand the differences between exact methods and heuristics
- Understand the differences between feasibility, local optimality and global optimality
- Basic ability in formulating models (mathematical programming, simulation, probabilistic, etc.)
- Some experience in designing solution methods for models
  
- Linear programming
  - duality
  - simplex algorithm
  
- Familiarity with various algorithm types and corresponding model-types for application
  - branch-and-bound / enumeration / tree search
  - gradient-based methods (for nonlinear)
  - dynamic programming
  - stochastic optimization methods
  - different types of heuristics

- Some exposure to IE topics such as supply chain management, scheduling, logistics, production, controls, etc.

***Computing Competencies:***

- Proficiency in using both Windows and UNIX/Linux operation systems
- Experience w/ a mathematical programming language (GAMS, AMPL, AIMMS, etc.)

***Programming Competencies:***

- Fluency in object-oriented computer programming
- Compiling code in different environments / operating systems
- Scripting language experience (Perl, Python, etc.)
- Basic conceptual knowledge of parallel computing
- Exposure to software design principles
- More advanced competency in algorithmic analysis & computational complexity
  - Big-O notation
  - algorithm design
  - algorithmic analysis

## **5. Additional Context**

### **5.1 Review of Non-academic Job Postings**

The Tier One skills and awarenesses support the general qualifications for a non-academic career as observed by reviewing three months of the 2007 job descriptions in OR/MS Today. Beyond technical abilities, nearly all postings required strong communication skills. Most positions request a degree in Computer Science, Industrial Engineering, Mathematics or Operations Research or a related area such as Econometrics, Finance, Physics or Statistics. All but one position included a proficiency with software - this advertisement instead focused on mathematical tools and even requested scores on the Putnam exam<sup>1</sup> and/or the Math Olympiad, so it is arguable whether or not this position should be included since it is not likely at the OR/CS interface. For the remaining advertisements, the following software recommendations were explicitly stated:

- 62% listed experience with a programming language such as C/C++/C\#, Java, Visual Basic, Perl, etc....
- 38% desired proficiency with mathematical programming software such as CPLEX, GAMS, AMPL, MPL, SAILS, etc....
- 46% stated a need for experience with statistical software like SAS or S-Plus.
- 58% wanted proficiency with database software such as Access, SQL, Excel or Oracle.

Detailed knowledge of a targeted sub-discipline is often required. For example, instead of saying that a candidate should have a knowledge of mathematical programming, the desired skills were instead focused on topics like dynamic programming or column generation, and instead of graph theory and/or network optimization, advertisements asked for an understanding of social network theory or network flows. This level of detail is difficult to support within the general guidelines of the ICS Education Committee since the union of all possible job skills would include the entirety of OR & CS. However, the breadth of such expertise is significant, and this review highlights the fact that there are quality careers for individuals who are broadly trained in OR/CS, who have programming and software skills, who have specific knowledge of a sub-discipline, and who are good communicators. Anecdotal evidence from the members of the ICS Education Committee supports this claim.

## 5.2 Review of Undergraduate and Graduate Programs in OR/CS-related Fields

Using the list of 141 U.S. educational programs found on the INFORMS web site (<http://www.informs.org/index.php?c=737&kat=ORMS+Educational+Programs>), a review of the educational requirements for graduate programs in the OR/CS interface and other OR-related fields was done. These programs came from Business Schools, IE programs, Mathematical Sciences programs, and independent OR or OR-Statistics groups. In addition, a review of the graduation requirements for various undergraduate programs in OR-related fields was performed. Roughly 50% of the education programs listed were examined for this study.

When examining the requirements for graduate program admittance, programs in Mathematical Science departments were not included, since their prerequisites would already include material we are considering.

Many of the undergraduate programs surveyed had similar requirements; it did not matter whether the programs were in business, engineering, or mathematics departments. They required the following courses for graduation:

1. Calculus through multivariable calculus
2. Linear Algebra
3. One or two computer programming courses

Many undergraduate programs also required a course in economics and probability and statistics (calculus-based).

Not all graduate programs listed prerequisite coursework for incoming students. However, those that did mimicked the above list in recommended coursework, also including courses in probability and statistics. Some programs required that a course in real analysis also be taken (at least at an undergraduate level) so that the student is exposed to the notion of a mathematical proof; this requirement, when found, primarily occurred for incoming Ph.D. students.

### ***Specific Examples from Programs***

This is a selection of the programs found on the INFORMS web site. These programs specifically noted their prerequisite requirements for incoming graduate students. Phrases and sentences in quotes come directly from the program's web site.

- Northwestern University - Industrial Engineering*  
 Northwestern expects incoming students to have a solid background in Linear Algebra (both matrix algebra and vector spaces), calculus-based probability and statistics, the idea of a mathematical proof, and exposure to a computer programming language (no specific language is required, but they recommend C++ and some knowledge in Matlab). They recommend both textbooks for review material and courses (undergraduate) to take in order to meet expectations.
- Auburn University - Industrial and Systems Engineering*  
 Undergraduate students are required to take courses in calculus, differential equations, probability and statistics, and a computer programming course using Matlab prior to beginning their major courses. Auburn expects incoming graduate students to have taken courses in calculus (through multivariable calculus), differential equations, probability and statistics, and both deterministic and stochastic operations research. They also expect a course in engineering economics.
- College of William and Mary - Computational OR program*  
 Incoming students “should possess a reasonable mathematical background, including calculus, linear algebra, and probability though they need not have obtained an undergraduate degree in mathematics. Applicants should have completed at least one course in a high-level programming language and are encouraged to have completed additional computer science courses such as data structures and algorithms.”
- Cornell University - Master of Engineering program in Operations Research and Information Engineering*  
 “Applicants should have four semesters of college calculus, through functions of several variables, as well as introductory calculus-based courses in probability and statistics. They should have taken a second-level computer science course in Pascal, C, or Java (a background in C or C++ is particularly valuable in seeking employment in financial services).”
- Cornell University – Ph.D. program in Operations Research and Information Engineering*  
 “Ph.D. students should have a strong background in engineering, mathematics, computer science, mathematical economics or one of the physical sciences. An undergraduate calculus-based probability course is advisable and students should have some background in computing.”
- Case-Western Reserve University - Department of Operations*  
 In the Master of Science in Management - Operations Research (MSM-OR) program, “the courses in this program require a working knowledge of undergraduate calculus that includes single and multivariate differentiation and integration, one semester of undergraduate linear algebra, and a working knowledge of C++.”

In the Ph.D. in Operations Research program, “the courses in this program require a working knowledge of three semesters of undergraduate calculus (including multivariate differentiation and integration) and one semester of undergraduate linear algebra. Entering students are given an assessment exam just prior to the beginning of their first semester to

determine their knowledge of the material in calculus and linear algebra used in our courses. Students are counseled individually to identify topics in these areas that they need to review and they are encouraged to take advantage of the graduate math consultant. It is also assumed that the student has previously had the equivalent of a one-semester undergraduate course in probability and one in structured programming (preferably with object oriented programming, such as C++).”

- *Columbia University – Department of Industrial Engineering and Operations Research*  
“Students entering the doctoral program must demonstrate proficiency in the following areas: Linear Algebra, Real Analysis, Probability, Statistics, and a programming language (e.g., C, C++, PASCAL, JAVA; either a course or substantial use in some project). A student lacking background in any of these prerequisites is expected to remedy the deficiency by the end of the first year.”
  
- *Ga. Tech - School of Industrial and Systems Engineering*  
“At the masters level, ... it is to be understood that candidates pursuing any of the Stewart School's Master's degrees should have or be willing to attain a mathematics background essentially equivalent to the first two years of an engineering degree, including exposure to a course in linear algebra. In addition, solid undergraduate-level courses in probability, statistics, and the fundamental methodologies of operations research will be required before enrolling in any of the respective graduate courses called for in the stated degree programs.”
  
- *Purdue University – Department of Industrial Engineering*  
For both the Masters and Ph.D. programs, “the following prerequisites are recommended for all incoming students:
  - Mathematics through multivariate calculus, differential equations, and linear algebra
  - Applied probability and engineering statistics
  - Basic techniques of operations research
  - Proficiency in computer programming”
  
- *Texas A&M University – Department of Industrial and Systems Engineering*  
“The department prerequisites listed below are intended to ensure that every graduate student has a background in mathematics, computing, and engineering adequate to the pursuit of graduate study in industrial engineering. Most entering students have a bachelor's degree in an engineering field. A degree in another discipline is acceptable provided the following course material has been taken or is taken concurrently with graduate studies.
  - A course in engineering economics.
  - Two courses in engineering calculus.
  - A course in linear algebra.
  - One additional mathematics course that requires first-semester calculus as a prerequisite.
  - One course in engineering mechanics or calculus-based physics.
  - One course in calculus-based probability and statistics.
  - Demonstrated proficiency in scientific computing.
  - A course in mathematical optimization.

In addition, an application for the Master of Science and Ph.D Programs should have one semester of probabilistic operations research covering stochastic processes.”

- *University of Delaware*

For incoming students to the Master's Degree (M.S.) (in Operations Research), the prerequisites for admission are the equivalent of: multivariable calculus, “Data Analysis and Nonparametric Statistics, Linear Algebra, and Computer Programming Languages.”

### 5.3 Review of Curriculum Recommendations from Other Professional Organizations

Curriculum recommendations from the Association for Computing Machinery (ACM), the Mathematical Association of America (MAA), the Society for Industrial and Applied Mathematics (SIAM), and the IEEE Computer Society (IEEECS) were examined. Most of the reports examined focus on making recommendations to programs and departments about development of specific degree programs, which as noted above is somewhat different from our aim. Documents available from MAA and SIAM barely mention operations research at all, while documents produced jointly by ACM and IEEECS mention operations research as a valuable elective course. No report outlining curriculum recommendations was available on the website for the American Mathematical Society (AMS). Study of these various reports is ongoing, and the Committee plans to speak directly to individuals in these organizations involved in producing their respective reports to determine how our own recommendations complement them.

---

<sup>1</sup>The Putnam Exam is an annual mathematical exam designed to identify mathematical talent. The median score is commonly zero.