Eastern Illinois University
New Course Proposal
MAT 4880, Design and Analysis of Algorithms

1. Catalog Description
   a. **Course number:** MAT 4880
   b. **Title:** Design and Analysis of Algorithms
   c. **Meeting times and credit:** 3-0-3
   d. **Terms to be offered:** Spring
   e. **Short title:** Algorithms
   f. **Course description:** Design paradigms of algorithms such as brute-force, exhaustive search, divide-and-conquer, backtracking, greedy approach and dynamic programming. Mathematical analysis of algorithms; P, NP, and NP-complete problems; heuristics, approximation algorithms, optimality.
   g. **Prerequisite:** MAT 3870 or permission of instructor
   h. **Initial term:** Spring 2007

2. Student Learning Objectives and Evaluation
   a. **Learning Objectives**
      Students will:
      1. apply the mathematical principles of algorithm analysis;
      2. apply algorithm design principles to new problems;
      3. identify algorithm implementation issues; and
      4. describe the theory of the classes P and NP and the impact of NP-complete problems.
   b. **Assessment**
      Course assessment will consist of evaluation of student work in the following areas:
      1. Textbook exercises (15%)
      2. Computer projects involving implementation and subsequent study of algorithms (15%)
      3. 2 in-class exams (40%)
      4. Final exam (30%)
      Student achievement of the stated goals will be assessed in the following manner:

<table>
<thead>
<tr>
<th></th>
<th>Homework Exercises</th>
<th>Programming Projects</th>
<th>Tests</th>
<th>Final Exam</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mathematical analysis of algorithms</td>
<td>X</td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Application to new problems</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Implementation</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>P/NP</td>
<td>X</td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

   c. **Technology**
      This course is not technology-delivered.
   d. **Graduate Student Assessment**
      Graduate students will be asked to do all work expected of undergraduates; in addition, exercises and projects will include more challenging problems. Emphasis will be placed on theoretical exercises requiring additional reading. Each textbook or homework assignment will include at least two additional problems for graduate students only. Each test will include at least one additional question for graduate students. Programming assignments will be identical to those for undergraduates.
3. **Outline of the Course**
   a. **Units of Time**

| Week 1 | Introduction to algorithm design paradigms.  
Specification of algorithms.  
Proving correctness of algorithms.  
Analyzing algorithms.  
Asymptotic behavior. |
|---|---|
| Week 2 | Types of problems: searching and sorting, string processing, graph problems,  
combinatorial problems, geometrical problems, numerical problems.  
Fundamental data structures: linear data types, graphs, trees, and sets.  
Exercise set 1 is due. |
| Week 3 | Analysis of recursive algorithms.  
Analysis of iterative algorithms.  
Empirical Analysis. |
| Week 4 | Brute-force: selection sort, sequential search, brute-force string matching, closest pair,  
convex hull, knapsack problem.  
Exercise set 2 due. |
| Week 5 | Divide-and-conquer: mergesort, quicksort, binary search, Strassen’s matrix multiplication,  
closest pair and convex hull problems.  
Exam 1. |
| Week 6 | Decrease-and-conquer: insertion sort, topological sorting, generating combinatorial  
objects, Josephus problem, finding a median, interpolation search.  
Exercise set 3 due. |
| Week 7 | Transform-and-conquer: presorting, LU decomposition, Horner’s rule, binary  
exponentiation, counting paths in a graph. |
| Week 8 | Dynamic programming: binomial coefficients, Warshall/Floyd algorithms, optimal binary  
search trees, the knapsack problem.  
Exercise set 4 due. |
| Week 9 | Greedy algorithms: Prim’s algorithm, Kruskal’s algorithm, Dijkstra’s algorithm, Huffman  
trees. |
| Week 10 | Lower-bound arguments: information-theoretic and adversary arguments.  
Decision trees: searching and sorting.  
Exercise set 5 due. |
| Week 11 | Space-time tradeoffs: string matching algorithms.  
Exam 2. |
| Week 12 | The classes P and NP; NP-complete problems.  
Solvability.  
Exercise set 6 due. |
| Week 13 | Backtracking algorithms. |
| Week 14 | Branch and bound: assignment problem, knapsack problem, traveling salesperson  
problem.  
Exercise set 7 due. |
| Week 15 | Approximation algorithms. |

b. **Technology-delivered**
   Not applicable

4. **Rationale**
   a. **Purpose and need**
   Algorithm analysis and design is a central focus of the discipline of computer science. Historically, the topics of data structures and algorithm analysis have been
taught as a single course, MAT 4870. However, each of these topics is extraordinarily rich and consequently it is not possible to cover the full range of topics in a single semester. Students interested in pursuing graduate studies in computer science will be expected to have a background in algorithm study.

b. Justification for course level and prerequisites
The proposed course requires maturity in two areas: mathematics and computer science. Students typically attain this maturity by completion of MAT 3870, which provides a strong background in data structures. Meeting the prerequisites of MAT 3870 provides the necessary mathematical background.

c. Similarity to existing courses
MAT 4870 currently examines the design, implementation, and analysis of data structures. This proposal goes along with concurrent requests to add a new course, MAT 3870, and to drop (by executive action) MAT 4870. A significant part of the proposed material for MAT 4880 will be drawn from the algorithm and analysis material formerly included in MAT 4870. This will provide, in MAT 4880, a better focus on design and analysis, especially for those algorithms not directly related to manipulating data structures. Some of this material has previously been offered in topics courses at the graduate level.

d. Impact on program
This course will provide a new and important upper-division elective for the Mathematics and Computer Science major. It could also be included as an elective in some options for the Mathematics major; it might be considered as an elective in graduate programs in Mathematics or in Technology as well as the undergraduate degree in CIS.

It is a required course for the Mathematics and Computer Science major. It could also be included as an elective in some options for the Mathematics major; it might be considered as an elective in graduate programs in Mathematics or in Technology as well as the undergraduate degree in CIS.

5. Implementation
   a. Faculty members to whom this course may be assigned
      Andrews, Mertz, Slough, Van Cleave, or other qualified faculty in the Department of Mathematics and Computer Science.

   b. Additional cost to students
      None.

   c. Text and supplementary materials

6. Community College Transfer
   Not applicable

7. Date approved by the department: October 24, 2005

8. Date approved by the college curriculum committee: November 11, 2005

9. Date approved by CAA January 12, 2006 CGS ____________________