

# CS 422 section 1, Winter 2016

**Instructor:** Michael Kowalczyk

**Office:** 2222 Jamrich Hall

**Office Phone:** 227-1600

**Office Hours:** 11:00am – 11:50am MWR, or by appointment

**Email:** mkowalcz@nmu.edu

**Class Meetings:** 10:00am-10:50am MWF in 1706 West Science

**Course Website:** <http://euclid.nmu.edu/~mkowalcz/cs422>

## Overview:

This course involves three main aspects: 1) Designing algorithms to solve computational problems, 2) formulating convincing arguments for algorithm correctness and 3) estimating the time (or space) required to run different algorithms. The first aspect involves mastering certain design techniques, such as *divide-and-conquer*, *greedy algorithms*, and *dynamic programming*. The later two involve the application of logic and mathematical techniques to understand the behavior of algorithms. Finally, we will learn about the famous theory of *NP-Completeness*, around which is centered one of the most important unresolved problems in computer science: “ $P=NP?$ ”!

## Prerequisites:

CS222 and one of MA161 or MA271.

## Textbook:

Our textbook is the 3<sup>rd</sup> edition of *Introduction to Algorithms*, by Cormen, Leiserson, Rivest, and Stein. It's a great textbook and it's also nice to add to your personal “CS library” after you are done with the course; I've used it countless times as a reference over the years. It is also available as an E-book through the NMU library, but note that a limited number of E-copies can be checked out simultaneously (so be mindful of others and sign out when you are done using it).

## Course Objectives:

Upon completion of the course, students should be able to do the following.

- Given a programming problem (for which students have not seen before) that is polynomial-time solvable through either divide and conquer, dynamic programming, or the greedy approach, students should be able to 1) develop an algorithm in pseudocode to solve it efficiently, 2) write a proof of correctness, and 3) provide an asymptotically optimal runtime estimate.
- Given pseudocode for an algorithm based on recursion or iteration, students can determine a runtime estimate expressed in asymptotic notation. In the case of a recursive algorithm, students can write a recurrence relation to model the runtime, and solve it.
- Given any two program runtimes that are likely to occur in practice, students can relate them using asymptotic notation, in such a way that conveys the most information possible.
- Given NP-complete problems  $A$  and  $B$  for which  $A$  is assumed to be NP-complete and  $A$  has a straightforward polynomial-time mapping reduction to  $B$ , students can prove that  $B$  is NP-complete.
- Students can identify at least two ways of coping with computationally intractable problems in real life.
- Students can write out the definition of the complexity class P.

**Grading:**

Grades will be based solely upon (two to seven) exams and (zero or more) quizzes, but I will also assign other work to help solidify your learning. It is important to practice constantly, as we will be learning some fairly challenging material. A quiz is weighted as 1/10<sup>th</sup> of an exam.

**Important Dates & Schedule Conflicts:**

The final exam will be on Thursday, April 28 from 10:00am until 11:50am. Any other exams will be announced ahead of time. Quizzes may happen at any time. Any conflicts with class meetings or the exams (due to religious observances, other coursework, intercollegiate athletics, etc.) should be made known to me as soon as you are aware of the conflict.

**Academic Conduct:**

I work hard, with honesty and integrity; I expect my students to do the same.

**Laptop Use:**

Due to the nature of this course, you won't need to bring your laptop to class (if there is ever an exception to this, I will announce it ahead of time). If you have a laptop with you, I ask that you to keep it closed in the interest of keeping distractions to a minimum.

**Disability Services:**

If you have a need for disability-related accommodations or services, please inform the Coordinator of Disability Services in the Disability Services Office by: coming into the office at 2001 C. B. Hedgcock; calling 227-1700; or e-mailing [disserv@nmu.edu](mailto:disserv@nmu.edu). Reasonable and effective accommodations and services will be provided to students if requests are made in a timely manner, with appropriate documentation, in accordance with federal, state, and University guidelines.