INEG 5613- Introduction to Optimization Theory
Online Course Syllabus (text 2nd edition update)

Instructor:
Dr. Ronald L (Ron) Rardin
Distinguished Professor Emeritus
Department of Industrial Engineering
Skype: ron.rardin
Telephone: cell/text (479) 236-6402
land line (469) 294-6253
Preferences: Email for short, easy-to-state questions and homework submissions
Skype or cell phone [times pre-agreed by email] for less structured questions
[9am to 9pm U.S. Central time, 7 days]

Course Schedule Summer 2017:
August 21 – October 15, 2017 (note: end-dates slightly extended beyond registrar schedule)

Description
Graduate-level introduction to the foundational rationales of numerical optimization methods and
model forms including linear programming, integer programming, network flows, and discrete dynamic
programming. Model formulation and tractability, search strategies, characterization of optimal
solutions, duality and sensitivity, outcome justification. Prerequisite: Graduate standing.

Emphasis
Although development will include frequent formulation and solving of example cases, emphasis is on
developing intuition about algorithmic strategies and underlying theory. Students will learn how and
why various algorithms and analysis tools work, how model forms affect performance, and which
application settings are best suited to each. The goal is to provide the background students need to
conduct sound optimization studies and to succeed in later coursework involving either applied
optimization topics or deeper theoretical issues.

Required Background
Some familiarity with matrix algebra (vector/matrix representation of linear equation systems), and
with formulation of elementary optimization models. (See text Primers 1, p.89, and 3, p.203)

Required Text
(Numerous topics, homeworks and exams will sync to the 2nd edition, so the 1st edition is not a
satisfactory substitute.)
Software Usage
Students will be required to make use of **AMPL commercial optimization software** available on INEG computers and downloadable from the Internet (see notes p. 6 below) at http://www.ampl.com
There will be no need for programming in C++, Java or similar general-purpose languages.

Course Structure and Outline
The course will be divided into 2 halves – one covering introductory material and Linear Programming optimization, and the second addressing Integer Programming and related material on Network Flows and Dynamic Programming. Each half consists of several major **blocks** of material, offered in 1-4 online lecture **modules**, followed by a **Midterm Exam**. The following two tables show the blocks, modules, assigned text materials, and approximate schedule for each half. The instructor reserves the right to make modest adjustments as the course evolves.

<table>
<thead>
<tr>
<th>WEEK</th>
<th>BLOCK</th>
<th>MODULE</th>
<th>TEXT MATERIALS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Aug 21-27</td>
<td>A Startup</td>
<td>1 Introduction</td>
<td>Syllabus, 2.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2 Graphic solution and outcomes</td>
<td>2.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3 Indexing, MP forms and computer solution</td>
<td>2.3-2.5, 2.7-2.8</td>
</tr>
<tr>
<td>2 Aug 28-Sep 3</td>
<td>B Search &amp; Convexity</td>
<td>1 Improving search</td>
<td>3.1-3.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2 Algebraic conditions for improving and feasible directions</td>
<td>3.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3 Linearity, convexity and global solutions</td>
<td>3.4</td>
</tr>
<tr>
<td></td>
<td>C LP Modeling</td>
<td>1 Modeling of LPs</td>
<td>4.1-4.5</td>
</tr>
<tr>
<td>3 Sep 4-10</td>
<td>D LP Structure &amp; Optima</td>
<td>1 LP optimal solutions</td>
<td>5.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2 LP standard form, basic solutions and extreme-point solutions</td>
<td>5.2</td>
</tr>
<tr>
<td></td>
<td>E Simplex Search for LPs</td>
<td>1 Rudimentary simplex</td>
<td>5.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2 Sufficiency, degeneracy, cycling and finiteness</td>
<td>5.6-5.7</td>
</tr>
<tr>
<td>4 Sep 11-17</td>
<td>F Duality &amp; Sensitivity</td>
<td>1 Qualitative sensitivity</td>
<td>6.1-6.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2 Quantitative sensitivity and dual definition</td>
<td>6.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3 Complementary slackness, what if questions, and multi-form dual taking</td>
<td>6.4-6.5</td>
</tr>
<tr>
<td>5 Sep 18-24</td>
<td>G Primal vs Dual &amp; KKT Optimality</td>
<td>1 Primal to dual relationships</td>
<td>6.7 (pp. 344-350)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2 LP optimality conditions and Primal strategy</td>
<td>Rest of 6.7</td>
</tr>
</tbody>
</table>

**MIDTERM 1 (three hours arranged within Sep 22-24)**
<table>
<thead>
<tr>
<th>WEEK</th>
<th>BLOCK</th>
<th>MODULE</th>
<th>TEXT MATERIALS</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>H</td>
<td>IP Modeling</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Modeling of ILPs</td>
<td>11.1-11.3, 11.5 (pp. 687-692), 11.6</td>
</tr>
<tr>
<td></td>
<td>I</td>
<td>Enumeration &amp; Relaxations</td>
<td></td>
</tr>
<tr>
<td></td>
<td>J</td>
<td>LP Branch &amp; Bound Search</td>
<td>12.1, 12.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>12.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>12.4</td>
</tr>
<tr>
<td>7</td>
<td>K</td>
<td>Cutting Planes and Branch and Cut</td>
<td>12.5 (pp 777-788)</td>
</tr>
<tr>
<td></td>
<td>L</td>
<td>Network Flow Modeling</td>
<td>10.1, 10.6</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>Network Flow Algorithms &amp; Integrality</td>
<td>10.2, 10.5</td>
</tr>
<tr>
<td></td>
<td>N</td>
<td>Shortest Paths &amp; Dynamic Programming</td>
<td>9.3-9.5</td>
</tr>
<tr>
<td></td>
<td>O</td>
<td>Computational Complexity</td>
<td>14.1-14.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>14.4-14.6</td>
</tr>
</tbody>
</table>

**MIDTERM 2 (three hours arranged within Oct 13-15)**

**Lesson Master Pages**

There is a multi-part master page posted for each block of the course under the Lessons tab of the Blackboard material for the course.

- **Things You Should Be Able to Do** comprises the first part of the learning objectives for the block – the skills and problem-solving capabilities you should learn. These are the first step in realizing the course's goal of introducing students to modeling and analysis in optimization.

- **Principles You Should be Able to Apply and Justify** comprises the second part of the learning objectives for the block – the key theoretical insights you should understand sufficiently to explain, apply and justify as part of studying the block. These add the second step of learning the rationales behind methods presented as a start on mastering related optimization theory.

- **Lecture Materials** for each module of the block include buttons to play the recorded lecture(s) for the module, along with reference material students will need to follow the lectures including "Screens" PowerPoints where available, and/or "Exhibits" to which the instructor refers during the lecture.
Assigned Readings for each module of the block detail text (2nd edition) materials students should study to gain a full understanding of the module.

Homework Exercises for each block assign tasks utilizing the “Things” skills part of learning objectives, and applying the “Principles” part. It is the instructor’s strongly held belief from long experience in teaching this material that the only way for students to master math-heavy material like this course is to require them to work with it in exercises.

How to Study Blocks and Modules of the Course
It is recommended that students study each of the blocks and modules of the course in approximately this sequence
1. Skim through the assigned text/reference materials for the module to capture the main ideas and tools involved
2. Download, and print or save to a write-on electronic pad, the “Screens” or “Exhibits” material for the module so that they will be available for note-taking as the lecture is played
3. Play and watch the recorded lecture for the module, taking notes on or around the downloaded lecture materials
4. Do and submit solutions for the assigned homework exercises, reviewing as needed relevant parts of the text materials, or replaying parts of lectures
5. Return to the learning objective “Things” and “Principles” for the block before you move on. Take each in turn, be sure you have learned what is required, and make notes to help you study later for the exams. Often this will require you to again go back for more detailed reading of text materials, or replaying parts of lectures

Submitting Homeworks
Homework answers for each block of the course should be submitted by email to the instructor (rrardin@uark.edu) after scanning as electronic attachments. Homework papers should be neat, but there is no requirement that they be typewritten. Where AMPL is required, submit the final log file for each model solved.

Students are strongly advised not to do their homework casually or depend on published/pirated answers, notes, etc. that may be available. Serious effort on the exercises directly prepares students for the “Things” part of the learning objectives, which will be the source of a major part of exams.

A recommended submission date is listed on the block master page for homework of each block. However, submissions will be accepted through the last-chance date, which is the Sunday ending the associated week of the course. Notice that this implies more than one block may be due on the same last-chance day. No results will be accepted after the last-chance date for the week

Homework submissions will be lightly reviewed for completeness and main ideas by the instructor, scored on the basis of 20-60 points per assignment, and emailed back to the student. Assignments will not count equally; total points are shown with each block. Shortly after the last-chance date/time for each block has passed, solutions will be available through Blackboard class software.
Examinations
Two proctored Midterm examinations are planned at the ends of the two halves of the course, each covering only material in that half. Exam questions will closely follow the Learning Objectives for covered blocks, requiring students to execute algorithms, apply relevant theory, formulate simple applications, and justify underlying principles. Copies of typical midterms from earlier offerings are available (without answers) through Blackboard class software.

The first Midterm exam for each student will take place sometime Fri-Sun, Sep 22-24, and the second sometime Fri-Sun Oct 13-15. Students are responsible for recruiting appropriate proctors and submitting the request form downloadable from the course Required Material in Blackboard. Both exams will provide students up to 3 hours to prepare their solutions under the watch of proctors in order to allow balanced coverage of all the learning objectives without excessive time pressure.

Both exams will be almost closed book, i.e., students may use no books/notes except two 8 1/2" by 11" sheets of handwritten notes (both sides). Multi-function calculators, pads, laptops, and wireless/telephone communication devices are prohibited during exams. Instead, students should arrange to bring a simple calculator without programmable or wireless communication features for use during exams. Proctors will be responsible for approving both the notes and the calculators used, and electronically scanning, then emailing student papers to the instructor at rrardin@uark.edu.

Partial Credit
Some homework and most exam questions will involve answers for which it is appropriate to award partial credit. To avoid confrontations over a few points, partial credit scoring will never be discussed one-on-one. In any case where the student believes scoring is inappropriate, he/she should attach a covering note briefly explaining why scoring seems wrong, and submit the subject item for a regrade by email within one week after papers are returned. The instructor will then reconsider the scoring in light of those written comments and determine what, if any adjustment is appropriate. He reserves the right to review all parts of the exam/homework if regrade of any part is sought.

Academic Honesty
Discussion of course content among students aids learning and is encouraged. However, each student is expected to submit his/her own work. No two submitted homework papers should ever be identical on any major part. No cooperation of any kind, or use of unauthorized notes and devices, is allowed during proctored examinations. A discussion of Academic Honesty policies, and processes is available at the University website http://honesty.uark.edu/policy/index.php.

Grading
Although the instructor reserves the right to deviate from strict mathematical weighting, grades will normally be derived as follows:

- After each exam is graded, the instructor will announce a (nonnegative) calibrating boost to be added to every student’s score in final course grade weighting.
- Then adjusted major grades will be weighted 40% for each exam, and 20% for homework average (total points received divided by total points possible) to obtain weighted averages.
- Those students with a final weighted average at least 90% are guaranteed an ‘A’, 80% a ‘B’, 70% a ‘C’, and 60% a ‘D’, etc.
Obtaining Access to AMPL

- AMPL is installed on all computers in INEG teaching labs. Use the AMPL directory of the C:\ drive.
  - Then access the code through a DOS command window as in course Module A3, slide 18
- A free student version can also be downloaded to your own computer from URL http://www.ampl.com
  - There select
    • AMPL for Teaching
    • FOR STUDENTS
    • Demo Version
    • Download a Free Demo
- Roll down to AMPL Command Line download for Windows
- Select ampl.mswin64.zip
- Create a new folder ampl
- Drag and drop ampl.mswin64.zip in the ampl folder
- Within the ampl folder, select ampl.mswin64.zip to unzip its contents into new folder ampl/ampl.mswin64/ampl.mswin64
- Next code all filename.txt ampl input files described above within that new folder
- Then access the code through a DOS command window as in course Module A3, slide 18

Further Note: Campus AMPL can also be accessed remotely as explained in http://industrial-engineering.uark.edu/technical-support/it-faq.php