Instructor:	Tamon Stephen
Meeting Time:	MWF 9:30–10:20 in SUR 5080
Office:	2886 Podium 2
Office Phone:	778–782–7429
E-mail:	tamon at sfu ca
Web page:	`http://www.math.sfu.ca/~tamon/Teaching/1147_Math408/`
Office Hours:	Monday 10:30–11:20 (tentative) and by appointment.
Text:	Integer Programming by Laurence Wolsey
Grading:	25% Homework, 25% Midterm, 50% Final.
	748: 20% Homework, 20% Presentation, 20% Midterm, 40% Final.

1. **Syllabus.** This course is an introduction to discrete optimization. The focus is on modelling problems as integer programs and polyhedral methods for solving these programs. Topics that we plan to cover include:

   Model building using integer, binary and mixed integer variables. Computer solution of integer programming models, linear programming relaxations, Lagrangian relations, duality, simple upper bounds using greedy algorithms. Branch and bound algorithms, implicit enumeration, LP based branch and bound.

   Valid inequalities, Gomory's fractional cut, mixed integer cuts, strong valid inequalities, simple facets for 0-1 knapsack polytope and the traveling salesman polytope, branch and cut algorithms.

   Lagrangian relaxation, strength of the Lagrangian dual, Lagrangian heuristics.

   Column generation algorithm, solving symmetric traveling salesman problem using column generation.

   Greedy and local search algorithms, construction heuristics, worst case analysis of heuristics.

2. **Graduate student projects.** Near the end of the term, each graduate student will present a brief (30 to 40 minute) introductory lecture on a current topic in integer programming. The topic will be selected in conjunction with the instructor. An possible source of topics are the surveys of current topics in the book *50 Years of Integer Programming 1958-2008*, which is on reserve. There may be an option to give these presentations in the SFU Operations Research Seminar series rather than in class.

3. **Homework.** There will be five homework assignments during the term. Late homework will not be accepted.

   You are encouraged to talk with each other and the instructor about the homework, but you must write up the solutions yourself, using your own words. Solutions copied from other students, textbooks or the Internet are **not** acceptable.

4. **Computing.** Integer programming is by its nature a computational subject, and students are encouraged to experiment with software for integer programming. Some integer programming capability is now available even in general purpose software such as the `Microsoft Excel` spreadsheet. There are also many specialized free and commercial packages for mathematical optimization. For instance, you can obtain student versions of the `AMPL` modelling language with several (limited) solvers from `http://www.ampl.com`. Depending on interest and availability of computing resources, we may also try out a full-featured version of the `CPLEX` solver. The assignments may involve the use of such software, however no prior computing background is required.

5. **Exams.** Books, notes and calculators cannot be used on the exams. Students **must** plan to take the tests at their scheduled times.

   The tentative dates and times for the tests are:

   ```
   Midterm: Friday, October 17th, 9:30-10:20 AM  (in class)
   Final: Friday, December 12th, 12:00-3:00 PM
   ```

6. **Reserve Books.** There is a copy of the course text on reserve at the SFU Surrey library. Additionally, there are two textbooks that cover similar ground: Bertsimas and Weismantel's *Optimization Over Integers* and Lee's *A First Course in Combinatorial Optimization*.

   A nice overview of the development of integer programming is contained in the book *50 Years of Integer Programming 1958-2008*, edited by Jünger et al. Schrijver's *Combinatorial Optimization* is an excellent reference book in this area. For a refresher on linear programming, Chvatál's book is available. The books of Papadimitriou and Papadimitriou and Steiglitz provide background on computational complexity theory.

Have a great term!