Due: Wednesday, September 23rd (noon)

Reading

Sections 1.1 through 1.4 of of Applegate, Bixby, Chvátal and Cook's *The Traveling Salesman Problem : A Computational Study*. This provides some cultural background on the Travelling Salesman Problem, a key example in this course. SFU library link.

Read the Introduction and Chapters 1 and 2 of the `AMPL` book: `http://ampl.com/resources/the-ampl-book/chapter-downloads/`. Note that most of the AMPL book covers *linear*, rather than integer programs. So the mathematical content is closely related to what you have seen in Math 308.

From the textbook, Chapter 1 and Sections 2.1 and 2.2. Sections 1.3 briefly introduces computational complexity, while Section 1.5 explores connection between discrete optimization and number theory. If you are not familiar with those subjects, then you may find these sections quite challenging.

Please make sure you understand the translation of combinatorial optimization problems into integer programs, for instance knapsack, set covering and travelling salesman. Try to convince yourself that it is straightforward to model most natural constraints using mixed integer programs. LP is already powerful, but doesn't easily capture integer variables or constraints of the form "A or B", "satisfying k of n constraints" or "takes a value from a specified discrete set".

Problems for Math 408 and Math 708

All problems to be submitted via Canvas. Please submit a single file names `hw1.pdf` containing all your written work, along with files `hw1.dat` and `hw1.mod` for the AMPL question. Please make sure to write your name on the first page of `hw1.pdf` and in the comments of `hw1.dat` and `hw1.mod`.

0. (Not graded.) Download and install `AMPL` following the instructions given in class.

You can use either the command-line or graphical (Integrated Development Environment) version.

1. Take your nine digit student id number and add 10 to each digit to get a sequence of nine numbers $a_1, a_2, \ldots, a_9$ between 10 and 19. Similarly, add 10 to each of the first nine digits of π to get a sequence $b_1, b_2, \ldots, b_n$. Your *personal knapsack problem* is:

$$\text{Maximize } \sum_{i=1}^{9} b_i x_i \quad \text{subject to } \sum_{i=1}^{9} a_i x_i \leq \frac{1}{2}\sum_{i=1}^{9} a_i \quad \text{and } x_i \in \{0,1\} \text{ for } i = 1, \ldots, 9.$$

a. Solve this integer program using `AMPL`, either via the command line interface or the `AMPL IDE` graphical interface. Use the `Cplex` solver. Please include a screen shot of the final solution in with your written solutions.

b. Use `AMPL` to solve the linear programming relaxation of your knapsack problem, that is:

$$\text{Maximize } \sum_{i=1}^{9} b_i x_i \quad \text{subject to } \sum_{i=1}^{9} a_i x_i \leq \frac{1}{2}\sum_{i=1}^{9} a_i \quad \text{and } 0 \leq x_i \leq 1 \text{ for } i = 1, \ldots, 9.$$

Do not submit the `AMPL` files for this problem. Instead explain briefly what changes you need to make to the files from the previous files to solve this problem.

Warning: If you use the default solver (`Minos`) in `AMPL`, it will *not* take the integrality constraints into account: it uses methods of continuous optimization and is not built to handle them. The two other available solvers are `Cplex` and `Gurobi`, you change to them via:

`option solver cplex;` and

`option solver gurobi;`

c. Now use AMPL to solve a mixed-integer programming version of your knapsack problem, where only variables 5 through 9 are required to be integer:

$$\text{Maximize} \sum_{i=1}^{9} b_i x_i \quad \text{subject to} \sum_{i=1}^{9} a_i x_i \leq \frac{1}{2}\sum_{i=1}^{9} a_i \quad \text{and } 0 \leq x_i \leq 1 \text{ for } i = 1, \ldots, 4;$$

$$\text{and } x_i \in \{0, 1\} \text{ for } i = 5, \ldots, 9.$$

Do not submit a the AMPL files for this problem. Instead explain briefly what changes you need to make to the files from the previous files to solve this problem.

2. Call the results from questions 1. ip^*, lp^* and mip^* respectively. Which of these numbers is largest, and which is smallest? Should this be true for all students in the class? Can some of the numbers be equal? Which ones?

3. Consider a situation where your model includes binary variables $x_1, x_2, \ldots, x_9$, each representing the potential purchase of a particular investment. Show how to represent each of the following constraints as a single linear constraint:

a. You can choose at most 5 of them.

b. If you choose investment 1, you must also choose investment 2.

c. If you choose investment 1, you may not choose investment 7.

d. You must choose either both investments 5 and 6, or neither of them.

4. Draw the convex hull of the following 2-variable sets:

a. $S_1 = \{x \in \mathbb{Z}^2_+ \ : \ 2x_1 + 3x_2 \leq 7,\ x_1 - x_2 \leq 1,\ x_2 - x_1 \leq 1\}$.

b. $S_2 = \{(x, y) \in \mathbb{Z}_+ \times \mathbb{R}_+ \ : \ x + y \geq 1.8,\ x \leq 3,\ y \leq 3\}$.

5. Textbook Exercise 1.15.

Additional Problems for Math 708

6. Consider the problem of colouring the vertices of a graph $G = (V, E)$ using the minimum possible number of colours such that no edge connects vertices of the same colour. Formulate this problem as an integer program. You can assume that you have an a priori upper bound k for the number of colours you will use, perhaps $k = |V|$ or something smaller based on knowledge of the graph. Comment on how the size of your formulation scales with $|V|$, $|E|$ and k.

7. Textbook Exercise 1.16.

8. A *magic square* is an arrangement of the numbers $1, 2, \ldots, n^2$ in an $n \times n$ box such that each row, column and diagonal has a constant sum. Formulate as an integer program the problem of finding a magic square maximizing the sum of the entries in its upper left quadrant. You may assume n is even. Comment on how the size of your formulation scales with n.