THE UNIVERSITY of ADELAIDE

Optimisation & Operations Research

Haide College, Spring Semester

Practical 5 (1%)

See website for practical and due dates

Work through the below instructions in MATLAB and MATLAB Grader as indicated. Submit to MATLAB Grader by the due date.

Branch and bound

Matt Roughan, Mike Chen

The aim of this practical is to try out Branch and Bound. We will proceed using MATLAB's `linprog` to solve the sub-problems, to avoid all of the minutiae of each relaxation.

Consider the Integer Linear Program (ILP)

$$\begin{array}{rrcrcrcl} \max z = & 2x_1 & + & 10x_2 & + & 1x_3 & & \\ \text{s.t.} & & & & & & & \\ & 6x_1 & + & 4x_2 & + & 4x_3 & \leq & 12 \\ & 4x_1 & + & 7x_2 & + & 3x_3 & \leq & 18 \\ & & & x_2 & + & 4x_3 & \leq & 17 \end{array}$$

with $x_i \geq 0$ and integer for all i.

---

1. Input the matrices and vectors A, **b** and **c** into MATLAB, and use `linprog` to solve the relaxed LP corresponding to the ILP above. **Check your solution in Matlab Grader.**

   Hints:

   - Write your solutions into a MATLAB `.m` script, so that you can add to it as you need for the following questions, and so that you can just cut and paste your solutions into Matlab Grader.
   - The Grader template will give you some ideas about how to format or structure your results.

---

2. Based on your solution to the relaxed problem branch on the variable x_2 by creating two new *IP*s with extra constraints

$$x_2 \leq \lfloor c \rfloor \quad (1)$$
$$x_2 \geq \lceil c \rceil \quad (2)$$

   where c is the value of x_2^* from the previous relaxation.

   Hints:

   - You don't have to change **c** at all, either add to your constraint matrix A and vector **b**. Perhaps call these `A1`, `b1`, `A2`, `b2` to distinguish them from the original problem.
   - Alternatively, you could implement this by adding lower/upper bounds to create the two nex problems (say `lb1`, `ub1`, `lb2`, `ub2`).

---

3. Solve the two new problems, again using `linprog`.

   Check your solutions in Matlab Grader.

   Hints:

- Make sure to get all the output from `linprog`, so you can test if the problems are feasible as well as what the solution is.

---

4. Follow the Branch and Bound Rules to analyse the results, and make appropriate further branches or fathom the branch, following down the Branch and Bound tree until you can determine the solution to the ILP.

   Check your final result in Matlab Grader.
