UNIVERSITY *of* LIMERICK
OLLSCOIL LUIMNIGH

Faculty of Science and Engineering

END OF SEMESTER ASSESSMENT PAPER

MODULE CODE: MS4315

SEMESTER: Autumn 2014

MODULE TITLE: Operations Research 2

DURATION OF EXAMINATION: 2 1/2 hours

LECTURER: Dr. M. Burke/Dr. J. Kinsella

GRADING SCHEME:

EXTERNAL EXAMINER: Prof. J. King

INSTRUCTIONS TO CANDIDATES:

- **Answer four questions correctly for full marks.**
- **Answer two questions from Q.1–Q.3 and two questions from Q.4–Q.6**
- **50% of the marks are for the two questions from Q1–Q3 and 45% for the two questions from Q4–Q6.**

1 (a) (i) Explain briefly why the following strategy for the solution of Integer Linear Programs (IP's) is not useful: "Solve the LP relaxation then round off the components of the solution to the nearest integers". 1

(ii) Explain carefully how lower bounds for the Standard Form IP $z = \max\{c^T x | Ax \le b, x \ge 0, x \in \mathbb{Z}^n\}$ may be found. 1

(iii) Explain carefully how LP Relaxation may be used to find an upper bound for z in a Standard Form IP. 1

(b) Given an LP (the **Primal** problem) we can write a closely related LP, its **Dual**:

$$\begin{aligned} z &= \max\{c^T x : Ax \le b, x \in \mathbb{R}^n, x \ge 0\} \quad \textbf{Primal} \\ w &= \min\{b^T y : A^T y \ge c, y \in \mathbb{R}^m, y \ge 0\}. \quad \textbf{Dual} \end{aligned}$$

(i) Prove the Weak Duality Theorem: for **any** primal feasible point x and **any** dual feasible point y, $b^T y \ge c^T x$. 2

(ii) Show that the Weak Duality Theorem implies that **any** feasible point y for the Dual problem gives an upper bound for the optimal solution z of the original IP, namely $z \le b^T y$. 2

(c) Consider the IP:

$$\begin{aligned} \max\ 16x_1 + 7x_2 & \\ 7x_1 + 3x_2 &\le 26 \\ 5x_1 + 2x_2 &\le 22 \\ x_1, x_2 \ge 0 \text{ and integer.} & \end{aligned}$$

You will be asked to partly solve the IP, using the Branch and Bound Method. **You should draw an enumeration tree/diagram to keep track of your progress — draw the enumeration tree on an otherwise blank page.**

The corresponding Simplex Tableau (transforming the max problem into a min problem) is:

$$T_0 = \begin{array}{|c|cccc|} \hline 0 & -16 & -7 & 0 & 0 \\ \hline 26 & 7 & 3 & 1 & 0 \\ 22 & 5 & 2 & 0 & 1 \\ \hline \end{array}$$

N.B. The Simplex Method and the Dual Simplex Method are stated on the last page of this paper.

You will partly solve the IP, following the steps on the next pages — remember to draw and fill in an enumeration tree.

(i) Apply one iteration of the Simplex Method to the starting tableau (T_0) and show that the Simplex Tableau now takes the form (**just calculate the first 3 columns**): 3

$$T_1 = \begin{array}{|c|cccc|}\hline 59\frac{3}{7} & 0 & -\frac{1}{7} & 2\frac{2}{7} & 0 \\ \hline 3\frac{5}{7} & 1 & \frac{3}{7} & \frac{1}{7} & 0 \\ 3\frac{3}{7} & 0 & -\frac{1}{7} & -\frac{5}{7} & 1 \\ \hline \end{array} \equiv \begin{array}{|c|cccc|}\hline 59.43 & 0.00 & -0.14 & 2.29 & 0.00 \\ \hline 3.71 & 1.00 & 0.43 & 0.14 & 0.00 \\ 3.43 & 0.00 & -0.14 & -0.71 & 1.00 \\ \hline \end{array}$$

(ii) After a second iteration of the Simplex Method the Simplex Tableau now takes the form: (**N.B. Do not perform the arithmetic!**)

$$T_2 = \begin{array}{|c|cccc|}\hline 60\frac{2}{3} & \frac{1}{3} & 0 & 2\frac{1}{3} & 0 \\ \hline 8\frac{2}{3} & 2\frac{1}{3} & 1 & \frac{1}{3} & 0 \\ 4\frac{2}{3} & \frac{1}{3} & 0 & -\frac{2}{3} & 1 \\ \hline \end{array} \equiv \begin{array}{|c|cccc|}\hline 60.67 & 0.33 & 0.00 & 2.33 & 0.00 \\ \hline 8.67 & 2.33 & 1.00 & 0.33 & 0.00 \\ 4.67 & 0.33 & 0.00 & -0.67 & 1.00 \\ \hline \end{array}.$$

Explain why this Tableau is LP optimal. 1/2

(iii) What is the solution to the LP Relaxation of the IP? 1/2

(iv) Why must we now branch on x_2 and what are the two branches? 1

(v) Consider the **right** branch corresponding to a lower bound on x_2 ($x_2 \geq$?).

A. Amend the LP-optimal tableau T_2 by adding an extra row & column corresponding to the extra constraint — creating a new tableau T_3 with 4 rows and 6 columns. 2

B. Perform one pivot on the appropriate element of T_3 to eliminate x_2 from the new row. **N.B. Just update the last row, not the full tableau.** 1

C. Explain why the resulting tableau is infeasible. Update your enumeration tree. 1

(vi) Starting at T_2, the **left** branch, $x_2 \leq 8$ (T_3, say) (when the new constraint is added and two pivots are performed) has the LP–optimal tableau T_3: (**N.B. Do not perform the arithmetic!**)

$$T_3 = \begin{array}{|c|ccccc|}\hline 60\frac{4}{7} & 0 & 0 & 2\frac{2}{7} & 0 & \frac{1}{7} \\ \hline 8 & 0 & 1 & 0 & 0 & 1 \\ 4\frac{4}{7} & 0 & 0 & -\frac{5}{7} & 1 & \frac{1}{7} \\ \frac{2}{7} & 1 & 0 & \frac{1}{7} & -0 & -\frac{3}{7} \\ \hline \end{array} \equiv \begin{array}{|c|ccccc|}\hline 60.57 & 0.00 & 0.00 & 2.29 & 0.00 & 0.14 \\ \hline 8.00 & 0.00 & 1.00 & 0.00 & 0.00 & 1.00 \\ 4.57 & 0.00 & 0.00 & -0.71 & 1.00 & 0.14 \\ 0.29 & 1.00 & 0.00 & 0.14 & 0.00 & -0.43 \\ \hline \end{array}.$$

Update your enumeration tree including the new upper bound on the overall problem. Why must we now branch on x_1 and what are the two branches? 2

(vii) Starting with T_3, add a row corresponding to the **left** branch. Why may we treat the new constraint as an equality constraint? 1

(viii) Explain why we must pivot on the entry in row 4 and column 2 of the resulting tableau. 1

(ix) The tableau (T_4 say) resulting from the extra row and the pivot is: **N.B. Do not perform the arithmetic!**

$$T_4 = \begin{array}{|r|rrrrr|} \hline 60\frac{4}{7} & 0 & 0 & 2\frac{2}{7} & 0 & \frac{1}{7} \\ \hline 8 & 0 & 1 & 0 & 0 & 1 \\ 4\frac{4}{7} & 0 & 0 & -\frac{5}{7} & 1 & \frac{1}{7} \\ \frac{2}{7} & 1 & 0 & \frac{1}{7} & 0 & -\frac{3}{7} \\ -\frac{2}{7} & 0 & 0 & -\frac{1}{7} & 0 & \frac{3}{7} \\ \hline \end{array} \equiv \begin{array}{|r|rrrrr|} \hline 60.57 & 0.00 & 0.00 & 2.29 & 0.00 & 0.14 \\ \hline 8.00 & 0.00 & 1.00 & 0.00 & 0.00 & 1.00 \\ 4.57 & 0.00 & 0.00 & -0.71 & 1.00 & 0.14 \\ 0.29 & 1.00 & 0.00 & 0.14 & 0.00 & -0.43 \\ -0.29 & 0.00 & 0.00 & -0.14 & 0.00 & 0.43 \\ \hline \end{array}$$

Perform one step of the Dual Simplex Method and show that the resulting tableau is

$$T_5 = \begin{array}{|l|rrrrr|} \hline 56 & 0 & 0 & 0 & 0 & 7 \\ \hline 8 & 0 & 1 & 0 & 0 & 1 \\ 6 & 0 & 0 & 0 & 1 & -2 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 1 & 0 & -3 \\ \hline \end{array}.$$

(**Just calculate the first and fourth columns, columns 2 & 3 do not change. Columns 5 & 6 are not needed.**) 2

(x) Finally; answer the following questions related to T_5, briefly explaining your answer to each.

A. What is the solution? 1/2

B. Is it IP optimal? 1/2

C. If possible update the global upper or lower bound on z? Explain which can be updated. 1

D. Is the IP solved? If not, what other branch or branches must be searched/fathomed/pruned? 1

2 This Question is on the Branch & Bound Method for **Binary** Integer (BIP's) programs. Pseudo-code for the Look-ahead version of the Branch & Bound Method for BIP's is given at the end of this Paper.

(a) The B & B method for BIP's assumes that the problem is a minimisation (min) BIP with all non-negative cost coefficients, where the cost coefficients are non-decreasing from left to right. Demonstrate how this can be achieved using the following example: 4

$$\begin{aligned} \max \quad & -x_1 + 2x_2 - 3x_3 - x_4 + 2x_5 \\ & -3x_1 + 2x_2 + x_3 + x_4 + 0x_5 \geq 1 \\ & -1x_1 - 3x_2 + x_3 + x_4 + x_5 \geq -1 \\ & x \in \mathbb{B}^4 \end{aligned}$$

(b) Given the transformed version of the BIP in part (a) (satisfying the above conditions, note the constant –4 in the objective)

$$\begin{aligned} \min y_1 + y_2 + 2y_3 + 2y_4 + 3y_5 - 4 \\ -3y_1 + y_2 - 2y_3 + 0y_4 + y_5 \geq -2 \\ -y_1 + y_2 + 3y_3 - y_4 + y_5 \geq 1 \\ y \in \mathbb{B}^4 \end{aligned}$$

(i) Generate an upper bound z_U for the optimal z. 1

(ii) Is the all-zeroes point $x = (0, 0, 0, 0)$ feasible for the two constraints? If it were, what would you conclude? 1

(iii) Use the look-ahead rule to find a non-zero lower bound z_L on the optimal z. 2

(c) (i) Now branch on $x_1 = 0/1$. Draw the enumeration tree. 2

(ii) Find updated lower bounds for the sub-problems L_1 & R_1. 2

(iii) Check that neither node can be pruned by infeasibility. 2

(iv) Now check that one (you work out which) of $x_L^{(1)}$ or $x_R^{(1)}$ is feasible for the two constraints. 2

(v) Explain why this node can be pruned for optimality. 2

(vi) Update the value of the incumbent. 1

(vii) Update the enumeration tree. 2

(viii) Can you prune the remaining node by bound — explain carefully. 2

(d) What is the optimal solution to the **original** BIP? 2

3 In this Question you will solve the following DP.

- An IT company must decide the number of specialist staff it needs over the next 3 months. The company estimates that the minimum number of such staff needed for each month is:

Month n	1	2	3
Min. Staff D_n	3	2	4

- The problem is stated in € rather than thousands of € for convenience.
- At least the minimum number of staff (see table) must be in place at the start of every month (following hiring/firing).
- Some or all staff can be "let go" at the beginning of each month.
- Additional staff can be hired at the start of each month at a fixed cost F of €4 plus a cost h of €2 per additional person hired.
- There is an excess cost E to the company of €3 per staff member at the beginning of each month in excess of the number needed (see the table).
- There is no cost incurred when a staff member is "let go".

(a) (i) Identify the states of the system. 1

(ii) Identify the stages or iterations. 1

(iii) Identify the "choices" x_n to be made at each iteration n. Hint: they may be negative as well as positive or zero. 1

(iv) Define $f_n(s)$ for any stage n and state s. 1

(v) Define $f_n(s, x_n)$ for any stage n and state s. 1

(vi) Justify writing the recurrence relation for $f_n(s, x_n)$ in terms of $f_{n+1}(.)$ and the various costs as

$$f_n(s, x_n) = f_{n+1}(s + x_n) + \Delta_n$$

where $\Delta_n = T_1 + T_2$ and $T_1 = (F + hx_n)$ if $x_n > 0$ and $T_2 = E(s + x_n - D_n)$ if $s + x_n - D_n > 0$ 4

(vii) For what value of n & s is $f_n(s)$ to be calculated so that the problem may be solved? 1

(b) Solve the problem using the information provided.

(i) Find the value of $f_n(s)$ for each "possible" value for s, working backwards from $n = 3$.
Hint: think carefully before filling in your tables — most entries are "impossible". 12

(ii) What is the minimum cost of meeting the required staff numbers? 1

(iii) What is the hiring/firing plan that attains the minimum cost? 2

4 An American call option (right to buy a share) has a strike price of €100 with 3 periods left to maturity. The current share price is €98. Analysis of the market indicates that the share price volatility per period is described by: the stock will go up by €2 with probability 0.4, stay the same with probability 0.3, decrease by €2 with probability 0.3.

Use a Dynamic Programming formulation to price the option. In particular, if $f_i(x)$ is the expected value of the option at time i for a share price of x, write down the *Bellman* equation for the scheme. 10
Find an expression for $f_2(x)$ as a function of x. 10
It can be shown that $f_0(98) = €0.544$ and $f_0(102) = €2.87$. Comment on the significance of these. 5

5 (a) In the context of a 2-person game, define what a *Nash* equilibrium is. 3

(b) What is a pure strategy ? What is a mixed strategy ? 3

(c) For strategies define strict dominance and weak dominance. 3

(d) By removing all strategies which are dominated by strict pure or mixed strategies, derive the reduced version of the following 2-player matrix game:

	D	E	F
A	(4,-2)	(3,0)	(-3,-1)
B	(-1,1)	(2,2)	(2,3)
C	(2,1)	(-1,-1)	(0,4)

8

(e) Derive the *Nash* equilibria and values of this game. 8

6 (a) The costs incurred by a firm in a production period are

$$c = 100 + 2x$$

where x is the number of items produced in that period. The items sell at a price of

$$p = 10 - \frac{x}{50}$$

each. Find the level of production that maximises the firm's profits when the firm has a monopoly. 6

(b) If two identical firms supply the market with $x_i,\ i = 1, 2$ items each at a cost per period of

$$c_i = 100 + 2x_i$$

respectively and sell each item at a price of

$$p = 10 - \frac{x_1 + x_2}{50},$$

analyse the resulting one shot *Cournot* game. 9

(c) If this 2-firm game is to be played repeatedly, consider the following "cooperative" strategy: a firm produces half of the optimal level associated with a monopoly (see part (a) for as long as the other firm does the same, and if the other firm deviates, it reverts to the single shot *Cournot* strategy thereafter. Does it ever pay to defect from the cooperative strategy? In particular, using the discount factor ω per period, when is this cooperative strategy a *Nash* equilibrium ? 10

Algorithm 1 (Simplex Method)

begin (Start with a Canonical tableau s.t. $\mathbf{b} \geq \mathbf{0}$.)
  while NOT finished do
    if $c_j \geq 0$ for all j
      then **STOP** (Tableau is optimal.)
      else Select j s.t. $c_j < 0$.
    fi
    if $a_{ij} \leq 0$ for all $i = 1, \ldots, m$
      then **STOP** (Problem is unbounded.)
    fi
    Select k such that:
    $\frac{b_k}{a_{kj}} = \min_i \left\{ \frac{b_i}{a_{ij}} \text{such that } a_{ij} > 0 \right\}$ (k attains the min.)
    Pivot on a_{kj}. (Divide Row k across by a_{kj} and add
  end … multiples of Row k to the rows above & below
end … … … introducing zeros into column j.)

Algorithm 2 (Dual Simplex Method)

begin (Start with a tableau s.t. $\mathbf{c} \geq \mathbf{0}$.)
  while NOT finished do
    if $b_i \geq 0$ for all i
      then **STOP** (Tableau is optimal.)
      else Select i s.t. $b_i < 0$.
    fi
    if $-a_{ij} \leq 0$ for all $j = 1, \ldots, n$
      then **STOP** (Dual unbounded $\equiv$ Primal infeasible.)
    fi
    Select k such that:
    $\frac{c_k}{a_{ik}} = \max_j \left\{ \frac{c_j}{a_{ij}} \text{such that } a_{ij} < 0 \right\}$ (k attains max.)
    Pivot on a_{ik}. (Divide Row k across by a_{ik} and add
  end … multiples of Row k to the rows above & below
end … … … introducing zeros into column i.)

Algorithm 3 (Look-ahead Binary Branch & Bound)

(1) begin
(2) **Initialise** : if $x = 0$ is feasible then $x = 0$ is opt, STOP fi
(3) Set bounds on z^*: $z_U = \sum_j c_j$, $z_L = z_1$, $x_L = (1, 0, \ldots, 0)$
(4) Initialise NodeList to x_L.
(5) if x_L is feasible then x_L is opt, STOP fi
(6) $k \leftarrow 1.\mathrm{NodeListEmpty} := \mathrm{FALSE}$
(7) while $\neg\mathrm{NodeListEmpty}$ do
(8) begin
(9) **Branch**: Select a remaining subset of feasible solutions (node)
(10) (the whole feasible set at the first iteration) and partition the subset
(11) into two smaller subsets by adding constraints $x_k = 0$ and $x_k = 1$.
(12) **Bound**: For each new subset, set x_L to the completion having its
(13) $(k+1)$st component $= 1$ and all subsequent components $= 0$.
(14) Use x_L to determine a lower bound z_L on z over that subset.
(15) **Prune/fathom**: Examine each unpruned node and prune it if
(16) $z_L \geq z_U$ **OR** one or more constraints cannot be satisfied
(17) by any completion in the subset **OR** x_L is feasible.
(18) if x_L feasible then x_L is new **incumbent**, $z_U \leftarrow z_L$.
(19) GOTO (15) and Check if other nodes can be pruned.
(20) fi
(21) **Test**:if no unpruned nodes left then STOP, **incumbent** is opt. else $k \rightarrow k+1$. fi
(22) end
(23) end