CIS521
Homework 3

Nathan Fraenkel

2.1 – LP & IP Question 1

1. Maximize:

$$5x1 + 1.5x2 + x3 + 2x4 + 7x5$$

Subject to:

$$8x1 + 5x2 + 3x3 + 6x4 + 10x5 \leq 400$$
$$x1, x3, x5 \geq 0$$
$$x2 \geq 10$$
$$x4 \geq 15$$
$$x1, x2, x3, x4, x5 \leq 24$$

➔ x1 = 2.5, x2 = 10, x3 = 0, x4 = 15, x5 = 24

➔ Maximum revenue = 5(2.5) + 1.5(10) + 0 + 2(15) + 7(24) = $ 225.50

➔ Weight used = x1 + x2 + x3 + x4 + x5 = 51.5 kg

2. The solution in the previous equation is not valid because we contain a weight that is not an integer, namely, x1 which is equal to 2.5 kg. To ensure that each package only weighs 1kg and to maximize our revenue, we must reevaluate x1 and x5 and say x1=5 and x5=22.

➔ Maximum revenue = 5(5) + 1.5(10) + 0 + 2(15) + 7(22) = $224.00

➔ Weight used = 52 kg

3. Rounding our solutions to question 1 would result in x1 = 3 kg, which would not be valid because we would violate the electric power constraint. Our total electric power used per day would be 8(3) + 5(10) + 0 + 6(15) + 10(24) = 404 kW/hr.
Therefore, we would need to add 4 to our original electric power constraint as "slack" so that our new, rounded solution would be valid.

4. Question 1 is easier to solve because it is a linear program, so it can be solved in polynomial time.
Question 2 is an integer programming problem, and since it cannot be cast as an LP, we would be stuck guessing and checking our 5 solutions for who knows how long or using complex search algorithms that can get up to exponential time.

2.2 – More LP & IP

1.

$x_i = n$	if I have n meetings with person i
$s_i = m$	if the meeting with person i starts at m o'clock
$e_i = k$	if the meeting with person i ends at k o'clock

➔ $\max \sum_i x_i$

Such that:

- $(s_i, e_i) \in S_i^{(av)}$ for all p_i

(the tuple must be an element of p_i's available time slots)

- $s_i \geq e_j$ AND $s_i \leq s_j$ for all i, j

(p_i's start time cannot be within a time slot of p_j)

2.
$\delta_{ijk} = 1$ if (i,j) on Sudoku board has value k

➔ max $\sum_{i,j,k} \delta_{ijk} = 1$

Such that:

- $\sum_i \delta_{ijk} = 1$ for all j, k only 1 k value per row
- $\sum_j \delta_{ijk} = 1$ for all i, k only 1 k value per column
- $\sum_k \delta_{ijk} = 1$ for all i, j all k values must be filled
- $\sum_{i=3x-2}^{3x} \sum_{j=3y-2}^{3y} \delta_{ijk} = 1$ for all k; x,y = [1:3] only 1 k value per sub square

2.3 – Adversarial Search

1. a) Nodes J, L and M will be pruned.
   b)

Node Letter	α-value	β-value
A	4	∞
B	-∞	4
C	4	3
D	4	1
E	4	4
F	8	8
G	6	6
H	7	7
I	3	3
J	-	-
K	1	1
L	-	-
M	-	-

2. a) The best move is to go left.

D will choose 2, E takes value 1, so then B takes value $(0.8)2 + (0.2)1 = 1.8$.

1.8 will be propagated up as the alpha value for A, and then back down as the alpha value for C and F.

F will take value -2, G will take value 2.

-2 wil propagate up as F's beta value, violating the alpha beta rule, so then we can prune F's right child.

C will take value $-2(0.6) + 2(0.4) = -0.4$.

A will take the max of B and C's values, taking value 1.8.

b) <u>First six</u>: Yes we will have to see the 7^{th} and 8^{th}.

B will take value 1.8, and $C = -1.2 + 0.4x$, where x = G's value.

G's two children could have values 1,000 and 1,001 for all we know, in which case C = 398.8, where A would take C's value, not B's.

<u>First seven</u>: No, we do not have to see the 8^{th} leaf.

$B = 1.8$ and $C = -1.2 + 0.4(\min\{2, x\})$ where x = value of 8^{th} leaf.

If 2 is the min, then $C = -0.4$, which is still less than B.

If x is the min, we know it is less than 2, so C will have a value less than or equal to -0.4.

In either case, A will always choose V's value given the first 7 leaves.

c)

$1.6 + -2(0.2) = 1.2$

$1.6 + 2(0.2) = 2.0$

So: $1.2 \leq C \text{ value} \leq 2.0$

d) Under the assumption from C, leaves 6, 7, and 8 do not need to be evaluated.

Leaf 6 does not need to get evaluated because leaf 5 has value -2, which is the smallest possible value, so F will take that value no matter what leaf 6 has as its value.

Leaves 7 and 8 also do not need to evaluated because given that leaf 5 has value -2, there is no way C can have a value greater than B's under the assumption in (c).