Department of Mathematical Sciences
Carnegie Mellon University

21-393 Operations Research II
Test 1

Name:____

Problem	Points	Score
1	35	
2	35	
3	30	
Total	100	

Q1: (35pts)

(a) Fill in the last column of the table below for solving the following knapsack problem and produce an optimal solution:

$$\begin{array}{lccccccc} \text{maximise} & 3x_1 & + & 7x_2 & + & 20x_3 & & \\ \text{subject to} & x_1 & + & 2x_2 & + & 3x_3 & \leq & 12 \end{array}$$

$x_1, x_2, x_3 \geq 0$ and integer.

w	$f_1(x_1)$	b_1	$f_2(x_2)$	b_2	$f_3(x_3)$	b_3
0	0	0	0	0		
1	3	1	3	0		
2	6	1	7	1		
3	9	1	10	1		
4	12	1	14	1		
5	15	1	17	1		
6	18	1	21	1		
7	21	1	24	1		
8	24	1	28	1		
9	27	1	31	1		
10	30	1	35	1		
11	33	1	38	1		
12	36	1	42	1		

(b) Read off the solution to the problem below from the above table.

$$\begin{array}{lccccccc} \text{minimise} & x_1 & + & 2x_2 & + & 3x_3 & & \\ \text{subject to} & 3x_1 & + & 7x_2 & + & 20x_3 & \geq & 60 \end{array}$$

$x_1, x_2, x_3 \geq 0$ and integer.

(c)

$B.V.$	x_1	x_2	x_3	x_4	ξ_1	ξ_2	RHS
x_0	1		1				7
x_2	1/3	1	$-1/5$				7/4
x_4	7/2		$-3/4$	1			11/4

Suppose that the above tableau represents the optimal basic feasible solution to a pure interger programming problem. Derive a Gomory cut, add it to the tableau, and indicate where the first dual simplex pivot should be.

Q2: (35pts)

A factory produces two products X,Y. There are monthly demands of d_X, d_Y for these products, respectively. The factory can only make one type of item in a period. It costs $c_X(x), c_Y(x)$ respectively to make x units of X,Y respectively. Assume there is a maximum storage of H allowed and initially there are $H/2$ items of each in store. There is also a storage charge of I per unit stored in any period. Because of cleaning problems, there cannot be three consecutive intervals in which the same product is being produced. Write down a dynamic programming recurrence to solve the problem.

Assume also that H is large enough so that there are no questions of feasibility.

Q3: (30pts)
Formulate the following as an integer program: n jobs need to be done and there are $3n$ people available. Each job requires three people to complete. Let

$$a_{i,j} = \begin{cases} 1 & \text{person } i \text{ and person } j \text{ cannot work together.} \\ 0 & \text{person } i \text{ and person } j \text{ can work together.} \end{cases}$$

$$b_{i,j} = \begin{cases} 1 & \text{person } i \text{ and person } j \text{ are good working togther.} \\ 0 & \text{person } i \text{ and person } j \text{ are only mediocre working togther.} \end{cases}$$

We want to determine a collection of n disjoint teams of three people that can cover every job. No team can contain a pair person i, person j for which $a_{i,j} = 1$. We want to maximise the sum

$$\sum_{i,j} b_{i,j} 1_{\{\text{person } i \text{ and person } j \text{ work together}\}}.$$