Department of Mathematical Sciences
Carnegie Mellon University
21-393 Operations Research II
Test 1

Name:______________________________

<table>
<thead>
<tr>
<th>Problem</th>
<th>Points</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>100</td>
<td></td>
</tr>
</tbody>
</table>
Q1: (40pts)
(a) Solve the following knapsack problem, writing the results of the dynamic programming recursion in a table. You will not score any points for just writing down the answer:

maximise \( 3x_1 + 7x_2 + 17x_3 \)
subject to
\[ 2x_1 + 3x_2 + 6x_3 \leq 10 \]
\[ x_1, x_2, x_3 \geq 0 \text{ and integer}. \]
Your answer should consist of a table.
(b) Using the answer to part (a), solve the following problem:

minimise \( 2x_1 + 3x_2 + 6x_3 \)
subject to
\[ 3x_1 + 7x_2 + 17x_3 \geq 20 \]
\[ x_1, x_2, x_3 \geq 0 \text{ and integer}. \]
(This does not require any new computations!)
Q2: (40pts) A system can be in 3 states 1, 2, 3 and the cost of moving from state \( i \) to state \( j \) in one period is \( c(i, j) \), where the \( c(i, j) \) are given in the matrix below. The one period discount factor \( \alpha \) is 1/2. The aim is to find a policy which simultaneously minimises the discounted cost of operating from any starting state. Start with the policy

\[
\pi(1) = 2, \pi(2) = 1, \pi(3) = 2.
\]

Evaluate this policy. Is it optimal? If not find an improved policy.

YOU DO NOT NEED TO EVALUATE THIS NEW POLICY OR FIND AN OPTIMAL STRATEGY.

The matrix of costs is

\[
\begin{bmatrix}
5 & 10 & 1 \\
8 & 2 & 2 \\
1 & 10 & 2
\end{bmatrix}
\]
Q3: (20pts)

Woody the woodcutter will cut a given log of wood, at any place you choose, for a price equal to the length of the given log. Suppose you have a log of length $L$, marked to be cut in $n$ different locations labeled 1, 2, \ldots, n. For simplicity, let indices 0 and $n + 1$ denote the left and right endpoints of the original log of length $L$. Let $d_i$ denote the distance of mark $i$ from the left end of the log, and assume that $0 = d_0 < d_1 < d_2 < \cdots < d_n < d_{n+1} = L$. The wood-cutting problem is the problem of determining the sequence of cuts to the log that will cut the log at all the marked places and minimize your total payment. Give a dynamic programming formulation to solve this problem.