CISC/CMPE-365*
Test #3
November 4, 2016

Student Number (Required) ______________________

Name (Optional)________________________________

This is a closed book test. You may not refer to any resources.

This is a 50 minute test.

Please write your answers in ink. Pencil answers will be marked, but will not be reconsidered after the test papers have been returned.

The test will be marked out of 50.

<table>
<thead>
<tr>
<th>Question 1</th>
<th>/34</th>
</tr>
</thead>
<tbody>
<tr>
<td>Question 2</td>
<td>/14</td>
</tr>
<tr>
<td>Question 3</td>
<td>/2</td>
</tr>
<tr>
<td>TOTAL</td>
<td>/50</td>
</tr>
</tbody>
</table>

The more dynamic you are, the more happens in your life, all the time.

Jaggu Vasudev
**Question 1 (30 Marks)**

You are attending a one-day bagpipe and cowbell festival which runs from 8 AM to 5 PM. Each hour two performances start – one 1-hour performance, and one 2-hour performance (there is no 2-hour performance starting at 4 PM), so the schedule looks something like this (note that this table shows time in “24-hour clock” form)

<table>
<thead>
<tr>
<th>Start Time:</th>
<th>08</th>
<th>09</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-hour</td>
<td>P1.9</td>
<td>P1.10</td>
<td>P1.11</td>
<td>P1.12</td>
<td>P1.13</td>
<td>P1.14</td>
<td>P1.15</td>
<td>P1.16</td>
<td>P1.17</td>
</tr>
<tr>
<td>performances</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2-hour</td>
<td>P2.10</td>
<td>P2.12</td>
<td>P2.14</td>
<td>P2.16</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>performances</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>P2.11</td>
<td>P2.13</td>
<td>P2.15</td>
<td>P2.17</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

where the P1.i are the 1-hour performances, and the P2.i are the 2-hour performances. The name of each performance indicates its **length** and **end time**.

You have a function \( f(P) \) that precisely predicts how much you will enjoy performance \( P \) for each performance. All the \( f(P) \) values are positive – you love that bagpipe/cowbell music!

Your goal is to choose how to fill your day with attending performances so as to maximize your total pleasure.

We can build a dynamic programming solution based on this observation: at 17 o’clock (5 PM), you have just finished attending either P1.17 or P2.17. So your total pleasure is either

\[
\text{f(P1.17) + “max pleasure obtainable from performances up to 16 o’clock”}
\]

or

\[
\text{f(P2.17) + “max pleasure obtainable from performances up to 15 o’clock”}
\]

... whichever is greater

This suggests a recurrence something like this:

\[
\text{MP(t) = max \{ } f(P1.t) + \text{MP(t-1)}, \text{f(P2.t) + MP(t-2)} \}
\]

... question continues on the next page ...
(a) [6 marks] Complete this recurrence by adding base cases as appropriate

(b) [6 marks] Describe the table you will use to store partial results

(c) [6 marks] Describe the order in which you will fill in the table

... question continues on the next page ...
(d) [8 marks] Describe how you will use the completed table to determine the list of performances you will attend.

(e) [8 marks] If we generalize this to a festival that runs for $n$ consecutive hours, what is the complexity of computing your optimal selection of performances? Explain.
Question 2 (14 Marks)

A sequence of integers is called **non-decreasing** if each number in the sequence is ≥ the one before it. We can abbreviate non-decreasing as **nd**.

For example, \( S = \{3,4,4,5,8,12,23,23,23,90\} \) is nd.

Let \( S = \{s_1, \ldots, s_n\} \) be a sequence of integers. Consider the problem of finding the longest nd subsequence of \( S \).

For example:

if \( S = \{6, 1, 5, 3, 0, 4\} \) then the longest nd subsequence of \( S \) is \( \{1, 3, 4\} \).

**10 marks** Give a recurrence relation for the length of the longest nd subsequence of a sequence \( S \).

Hint: the largest possible value in the nd subsequence is the largest value in \( S \). But if we use \( S[n] \), then its value becomes the upper limit for all other values we choose. This suggests basing our recurrence on something like this:

\[
M(k, x) = \text{the length of the longest nd subsequence of } S[1..k] \text{ in which all chosen values are } \leq x.
\]

Now show how \( M(k, x) \) is related to \( M(k-1, x) \) and \( M(k-1, S[k]) \), based on whether we use \( S[k] \) or not.

Explain your reasoning.

Feel free to use another approach to finding a valid recurrence relation if you prefer.

**4 marks** Don’t forget to include base case(s) as needed.

.... answer space continues on next page ....
.... more space to answer Question 2 if needed ....
Question 3: (2 marks)

True or false: if \( S = \{ s_1, \ldots, s_n \} \) and \( k \leq n^3 \) then the question

“Does S contain a subset that sums to k?” can be answered in polynomial time.

\[ \text{TRUE} \]