UIT2201: Computer Science and Information Technology Revolution  
(Spring Semester 2012)

MID TERM TEST   (1 hour)

INSTRUCTIONS:

1. This Mid-Term Test is CLOSED BOOK / CLOSED NOTES.
2. Answer ALL questions in this answer book.
3. Make sure you write down your NAME and MATRIC NUMBER

Name: __________________________________________________________

Matric Number: __________________________

<table>
<thead>
<tr>
<th>QUESTION</th>
<th>POSSIBLE</th>
<th>SCORE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>Q2</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>Q3</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>Q4</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>TOTAL</td>
<td>60+1</td>
<td></td>
</tr>
</tbody>
</table>

Fun Question: (1 bonus point)

Name the Nobel Laurette who is associated with the famous “Piano Tuner Problem?  
__________________________________________________________________

(Now, please relax and enjoy the Quiz, OK?)
Question 1: (20 points)

True-False (2 point each)

(z) (Sample) The course UIT2201 is taught by Prof. Leong Hon Wai _TRUE_

(a) An algorithm X with running time 2012n will be faster than another algorithm Y with running time 0.2012n² for all values of n. _______

(b) In Scratch, we can define a variable that is capable of storing many data values, under one variable name (thus, reducing the need to “invent” many variable names). _______

(c) The binary search algorithm is very fast because it keeps reducing the problem size by a factor of two. _______

(d) The Find-Max(A, n) primitive that finds the maximum in an array A[1..n], of n unsorted numbers has time complexity Θ(lg n). _______

(e) Query processing (for example, answering SQL query) is no longer a complex research issue since we have very fast computers nowadays. _______

(f) Each column of a database table is called a schema. _______

Not True-False

(g) (4 points each) Rank the following time complexity in increasing rate of growth (from slowest to fastest).

\[ \Theta(n^2), \ \Theta(n), \ \Theta(n \ lg \ n), \ \Theta(2^n). \]

Answer: ____________________________________________

Fill in the blanks: (2 points each)

(h) Give a common application of the Pattern-Match algorithm covered in class.

Answer: ____________________________________________

(i) Name your favourite recurring principle covered in class.

Answer: ____________________________________________
Question 2: (15 points)

You are given two lists (arrays) of characters, \( T[1..n] \) and \( P[1..m] \) (where \( n \gg m \)). You can assume that \( T \) and \( P \) have already been filled with data (i.e. data has been read in).

You are given the following algorithm called \( \text{U-Count}(T, k, P, m) \):

\[
\begin{align*}
\text{U-Count}(T, k, P, m) & : \\
(* \text{ Compute some unknown quantity from } *) & \\
(* \text{ array } T[k..k+m-1] \text{ and array } P[1..m] *) & \\
\text{begin} & \\
1 & s\text{-count} \leftarrow 0; \\
2 & j \leftarrow 1; \\
3 & \text{repeat until } (j > m); \\
4 & \quad \text{if } (P[j] = T[k+j-1]) \text{ then} \\
5 & \quad \quad \text{s\text{-count} } \leftarrow \text{s\text{-count} } + 1; \\
6 & \quad \text{endif} \\
7 & \quad j \leftarrow j + 1; \\
8 & \text{endrepeat} \\
9 & u\text{-count} \leftarrow m - s\text{-count}; \\
10 & \text{print } u\text{-count}; \\
\text{end;} 
\end{align*}
\]

(a) (6 points) Suppose that \( T[1..10] = \{\text{R E V O L U T I O N}\} \) and \( P[1..4] = \{\text{V O L T}\} \). If we run algorithm \( \text{U-Count} \) with different inputs, what is the output (of line 10), and what is the final value of the variable \( s\text{-count} \) (at the end of the algorithm)?

<table>
<thead>
<tr>
<th>Output produced</th>
<th>Value of ( s\text{-count} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \text{U-Count}(T, 1, P, 4) )</td>
<td>___________________________</td>
</tr>
<tr>
<td>( \text{U-Count}(T, 3, P, 4) )</td>
<td>___________________________</td>
</tr>
<tr>
<td>( \text{U-Count}(T, 8, P, 3) )</td>
<td>___________________________</td>
</tr>
</tbody>
</table>

(b) (3 points) Describe in plain English, what \( \text{U-Count} \) actually counts?

(b) (2 points) What is a dominant operation for this algorithm \( \text{U-Count} \)?

Dominant Operation: ___________________________

(b) (4 points) Give the worst-case time complexity (expressed in the \( \Theta \)-notation) of the algorithm \( \text{U-Count}(T, k, P, m) \) for general \( T \) and \( P \)?

Time Complexity: ___________________________
Question 3: (15 points)

You use the binary search algorithm `Bin-Search(SL, 9, aName)` given in class to search for `aName` (a given name) in a sorted list `SL` of 9 names, given below.

\[ SL = (\text{Alvin, Beth, Cathy, David, Eve, Faith, Gail, Hon, John}) \]

(a) [5] Draw the search tree that is used to visualize this binary search algorithm applied to the sorted list `SL` of size 9. (Include the “square” nodes for unsuccessful searches).

(b) [4] How many name-comparisons are needed to search (successfully or unsuccessfully) for each of the names given in the table below:

<table>
<thead>
<tr>
<th>aName</th>
<th>Number of Name-Comparisons</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eve</td>
<td>________________________</td>
</tr>
<tr>
<td>Cathy</td>
<td>________________________</td>
</tr>
<tr>
<td>Bunny</td>
<td>________________________</td>
</tr>
<tr>
<td>John</td>
<td>________________________</td>
</tr>
</tbody>
</table>

(c) [3] Assuming that each name is equally likely to be searched, what is the average number of name-comparisons used in a successful search?

(d) [3] Estimate the average number of name-comparisons used in an unsuccessful search. (You can assume that each “unsuccessful search node” is equally likely.)
Question 4: (10 points)

Given a database with the following 3 tables: \{SI, CI, EN\}. You should use these short table names to save space and writing. (Use the reverse blank pages, if necessary)

<table>
<thead>
<tr>
<th>SI (STUDENT-INFO)</th>
<th>CI (COURSE-INFO)</th>
<th>EN (ENROLMENT)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Student-ID</td>
<td>Name</td>
<td>NRIC.No</td>
</tr>
<tr>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
</tbody>
</table>

For (a) and (b) below, give the appropriate (i) SQL query, and (ii) a sequence of basic database primitives (using `e-project, e-select, e-join`):

(a) [4 pts] List the Course-ID, Day, Hour of all courses taught in the venue “USP–SR1”.

(i) SQL Query:

(ii) Using Basic Primitives:

(b) [6 pts] List the Student-ID, Major of FASS students who have classes in “USP–SR1”.

(i) SQL Query:

(ii) Using Basic Primitives:

~~~ END OF QUIZ ~~~