NATIONAL UNIVERSITY OF SINGAPORE

(Semester II : 2009-10)

UIT2201 : COMPUTER SCIENCE AND THE IT REVOLUTION

April 2010 – Time Allowed: 2 Hours

INSTRUCTIONS TO CANDIDATES

1. This examination paper consists of FIVE questions and comprises TEN printed pages including this page.

2. Answer ALL questions.

3. Write ALL your answers in this examination book.

4. This is an OPEN BOOK examination.

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>15</td>
<td></td>
</tr>
<tr>
<td>Q5</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>TOTAL</td>
<td>80</td>
<td></td>
</tr>
</tbody>
</table>
Question 1: (20 marks)

True-False questions. (2 marks each)

(a) An algorithm that contains an infinite loop cannot be a correct algorithm to a computational problem. ______

(b) Whenever we can write an algorithm to solve a computational problem, $P$ say, we can program the algorithm on a computer and have a fast and efficient solution to the problem $P$. ______

(c) An algorithm $K$ with running time $2010n$ is faster than an algorithm $L$ with running time $0.2201n^2$ for large values of $n$. ______

(d) When searching a sorted array of size $n$ using binary search, if $n$ doubles, then the number of name-comparisons also approximately doubles. ______

(e) In database query processing, the join operation is a computationally expensive operation even when we have very fast machines. ______

(f) In the design of Wide Area Networks (WANs), an important property is that there should be multiple paths between any pair of hosts. ______

(g) The early AI program ELIZA is a program that passes the Turing test. ______

(h) The study of algorithms lies at the heart of computer science. ______

(i) When analyzing the time complexity of an algorithm, we usually consider the worst-case performance of the algorithm over all possible input instances of a given input size. ______

(j) In AI, an example of a recognition task is the task of locating faces in a digital image, such as the image taken with a digital camera. ______

Fun Question: (1 bonus mark)

Give an example of the “repeated doubling” phenomena from a domain other than those we have already covered during the course.

Your Answer:
Question 2: (15 marks)

Consider a database with the following 3 tables: {SI, CI, EN}. We assume that |SI| = 30,000, |CI| = 1000, |EN| = 100,000. (Use the blank reverse pages, if necessary. To save space and writing, you should use the short table names.)

<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>
</tbody>
</table>

(a) (6 Marks) You want to list the Course-ID, Student-ID of all courses and students taught by the instructor “H. T. Gersting”. Give the appropriate (i) SQL query, and (ii) sequence of basic database primitives operations (using e-project, e-select, e-join) to accomplish the task.

(i) SQL Query:

(ii) A Sequence of Basic Primitives:
Question 2: (continued…)

In the USP computer lab, Prof. S. Harp came across a printout on the floor with the following sequence of DB operations as answer to some (unknown) tutorial problem.

\[
\begin{align*}
X_1 & \leftarrow \text{join SI and EN where } (\text{SI.Student-ID} = \text{EN.Student-ID}); \\
X_2 & \leftarrow \text{select from } X_1 \\
& \quad \text{where } (\text{Course-ID} = "UIT2201") \text{ and } (\text{Faculty} = "FOE");
\end{align*}
\]

The code was obviously not written by a UIT2201 student. Upon seeing it, Prof. S. Harp deduced that the code was correct and it solved a given problem, but still, he was not happy at all. Prof. S. Harp then took the printout and showed it to you.

(b) (2 Marks) State the problem that this code fragment was supposed to solve.

Answer: List

(c) (2 Marks) Explain why Prof. S. Harp was still not happy at all.

(e) (5 Marks) What would you do (to the code) to make Prof. S. Harp happy?
Question 3: (15 marks)

(a) (3 marks) In the circuit below, the rectangles Q represent the same type of gate. Based on the input and output information given, identify whether Q can be any one of the following: AND, OR, or XOR gate.

![Circuit Diagram]

Circle the correct answer below:

AND-gate: Yes No  
OR-gate: Yes No  
XOR-gate: Yes No

(b) (4 marks) In your logic design laboratory, you are given a “mystery” gate X (shown below) with the truth table defined on the right.

![Truth Table]

Using a truth table (or otherwise), give the logical formula for the output Z of the circuit shown below.

![Circuit Diagram with Z]

Answer: \( Z = \) ____________________________

(c) (2 marks) In the memory unit, the MAR and MDR registers are used to support two basic operations. Name these two basic operations.
Question 3: (continued…)

(d) (3 points) You purchase a 2MB (2 x 2^{20} bytes) random access memory (RAM) and it is rectangular in shape where the column width is 8 times the row width.

How many bits must be used to specify the address of each cell in this RAM?

Answer: ___________________

How many bits are there in the row selector and how many in the column selector?

Row Selector: _____________ Column Selector: _________________

(e) (3 marks) It is known that the set \{+, *, ~\} (of logical OR, AND, and NOT operations/gates) is logically complete, namely, we can implement any logical formula with a combination of these three operations/gates. Explain why the set \{+, ~\} is also logically complete.
Question 4: (15 marks)

You are given a list \( A[1..n] \) of \( n \) positive integers (assume that \( n \) is a multiple of 2). You want to partition the list \( A \) into two subsets, \( B \) and \( C \), each of size \( n/2 \). Define \( \text{sum}(B) \) to be the \textit{sum of the numbers in the set} \( B \) (and define \( \text{sum}(C) \) similarly). Also define the \textit{sum-difference}, namely \( \text{sum-diff}(B,C) = |\text{sum}(B) - \text{sum}(C)| \).

For example, consider \( A = [9, 1, 3, 4, 2, 5] \).
- If \( B=[9,1,3] \) and \( C=[4,2,5] \), then \( \text{sum}(B)=13 \), \( \text{sum}(C)=11 \), and \( \text{sum-diff}(B,C)=2 \).
- If \( B=[9,1,5] \) and \( C=[4,2,3] \), then \( \text{sum}(B)=15 \), \( \text{sum}(C)=9 \), and \( \text{sum-diff}(B,C)=6 \).

The goal is to compute a \textit{max sum-diff partition} of \( A \), which is defined as a partition of \( A \) into \( B \) and \( C \) in such a way that the difference in the sums is \textit{as large as possible}, namely, \( \text{sum-diff}(B,C) \), is maximized.

(a) (2 marks) For the given example, where \( A = [9, 1, 3, 4, 2, 5] \), find a \textit{max sum-diff partition} of \( A \), namely \( B \) and \( C \) that maximizes \( \text{sum-diff}(B,C) \).

Answer: \( B = \) ____________________________  \( C = \) ____________________________

(b) (6 marks) Design an algorithm to compute a \textit{max sum-diff partition}. Give your algorithm in pseudo-code. (You are free to quote any algorithm covered in the course. Quote them as \textit{high level primitives} and clearly state \textit{what} they do.)

(c) (2 marks) Give the time complexity (running time) of your algorithm.

Answer: ____________________________
Question 4: (continued…)

(d) (2 marks) For the given example, where \( A = [9, 1, 3, 4, 2, 5] \), find a min sum-diff partition of \( A \), namely a partition of \( A \) into subsets \( B \) and \( C \) in such a way that the difference in the sums is as small as possible, namely, \( \text{sum-diff}(B,C) \), is minimized.

Answer: \( B = \) _______________________  \( C = \) _______________________

(e) (3 marks) Give a rough sketch of an algorithm to compute a min sum-diff partition for small problem instances (where the list \( A \) has fewer than 20 elements).
Question 5: (15 marks)

You are given a knowledge based system with the following knowledge base,

**Knowledge/Fact Base:**

- LChild(David, John)
- LChild(Tom, Diana)
- RChild(Mary, David)
- RChild(Peter, Tom)
- RChild(Fish, Ruby)

where LChild(X,Y) means “X is left child of Y” and
RChild(X,Y) means “X is right child of Y”.

**Inference Rules:**

R1. Child(X,Y) if LChild(X,Y)
R2. Child(X,Y) if RChild(X,Y)

(a) (3 marks) Draw a “family tree” based on the fact base given. (In the “family tree”, the “child” is drawn below the “parent” with arrow from parent to child.)

(b) (3 marks) Answer the following queries: (no need to show the steps)

- ?LChild(Mary, David)  
  Answer: ______________________
- ?RChild(X, Diana)  
  X = ______________________
- ?Child (Y, John)  
  Y = ______________________

(c) (2 marks) Give inference rule(s) for the Descendant relationship where Descendant(X,Y) means “X is descendant of Y” and

R3: Descendant(X,Y) if

(d) (3 marks) Answer the following query: (no need to show the steps)

- ?Descendant(Peter, Tom)  
  Answer: ______________________
- ?Descendant(W, David)  
  W = ______________________
- ?Descendant(Fish, Z)  
  Z = ______________________
Question 5: (continued…)

(e) (4 marks) In the course, we discussed several recurring principles. One of them is the “divide and conquer” approach. Give two different examples in this course where this recurring “divide and conquer” approach is at play.

~~~ END OF QUESTIONS ~~~