UIT2201: Computer Science and Information Technology Revolution
Spring 2012 – Final Exam (Solution Sketch)

(NOT TO BE GIVEN TO FUTURE UIT2201 STUDENTS)

Question 1: (20 marks)

(a) -- (j) T T F F F F F T T F

Fun Question: (1 bonus mark) \(_{(10)}_2 = (2)_{10}\)

Question 2: (15 marks)

(a) (3 marks)
SELECT Student-ID, Name, Tel-No
FROM SI, EN
WHERE (SI.Major = "Origami") AND
(EN.Course-ID ="UIT2201") AND
(SI.Student-ID = EN.Student-ID)

(b) (3 marks)
It is computationally very expensive to do the e-join operations
And in this case, there is no need to involve the CI table at all.

(c) (3 marks)
H1 \leftarrow e-select from SI where (Major="Origami");
H2 \leftarrow e-select from EN where (Course-ID="UIT2201");
H3 \leftarrow e-join H1 and H2 where (H1.Student-ID = H2.Student-ID);
Ans \leftarrow e-project Student-ID, Name, Tel-No

(d) (2 marks)
The same information appearing in many records (duplicated).
For example, fields such as \{NRIC-No, Address, Tel-No, Faculty, Major\}
appearing in all records of a particular student;

(e) (4 marks)
The deletion problem:
When some info about a student is deleted, the student
also "disappear" from the database.
Or any other such examples will also do.

Question 3: (15 marks)

(a) (3 marks) AND-gate: Yes OR-gate: No XOR-gate: Yes [No]#
(b) (4 marks) Truth Table: DIY \[ Z = \neg P \land \neg Q + P \land Q \]
(c) (2 marks) Size of Memory Unit: \(2^{32} = 4\text{Gigabytes}\)
(d) (2 marks) Row Selector: 16 bits Column Selector: 16 bits
(e) (4 marks) Binary Search (recursive halving)
Decoder Circuit \# of output doubles with each address line
And any other correct examples…
# Mistake found by KT (Kristen Tang) and DC (Davin Choo)
Question 4: (15 marks)

(a) (2 marks)
Easy to implement -- reuse the code from MeSM and MeBG. The two different search procedures have already been tested in their respective companies.

(b) (4 marks)  Worst Case: _1000_ Average Case: _(1001)/2 = 500.5_

(c) (3 marks)  Worst Case: _1000+20___

(d) (4 marks)
One-time pre-processing: Sort the combined list.
(Time taken: O(n log n) or O(n^2) if using Selection Sort.)
Then, can use Binary Search on combined list  (length = 1,001,000)

(e) (2 marks)  Worst Case: __20____

Question 5: (15 marks)

(a) (3 marks) “road network” -- DIY

(b) (5 marks)

\[ ?\text{ROAD} (B, C) \quad \text{Answer: YES} \]
\[ ?\text{ROAD} (X, D) \quad X = A, \quad X=B, \quad X=E \]
\[ ?\text{PATH} (A, B) \quad \text{Answer: YES } \text{(}\text{ROAD} (A, B) \text{)} \]
\[ ?\text{PATH} (A, E) \quad \text{Answer: \_ROAD} (A, B), \quad \text{ROAD} (B, D), \quad \text{ROAD} (D, E) \quad \_\]
\[ \_\text{ROAD} (A, D), \quad \text{ROAD} (D, E) \_\]

(c) (2 points)
The maximum sum is obtained when the two numbers are the largest possible.
1. Use the algorithm for Largest and 2\text{nd} Largest.
2. Add sum of these two numbers.

(d) (5 points)
Idea: First sort array \(A[1..n]\):
Then, if we include \(A[k]\) , then use binary search to find \(X - A[k]\);

Algorithm:
\[ k \leftarrow 1; \]
While (\(k <= n\)) do
\[ T \leftarrow X - A[k]; \]
\[ m \leftarrow \text{Binary-Search} (A, 1, n, T) \quad \text{(* binary-search for } T \text{ in } A[1..n] \text{ *)} \]
if (found) then { Print \(A[k]\) and \(A[m]\) and Exit while-loop; }
\[ k \leftarrow k + 1; \]
endwhile

~~~ END OF QUESTIONS ~~~