I101 Review Notes

helpful hints

Mehmet M. Dalkilic

December 10, 2006

Review

The final exam is cumulative. Please try to answer all questions! If you find you can’t complete a problem, at least provide some way for us to give you partial credit—so write something pertaining the problem. Answers left blank, while easy to grade, are not helpful to you. Please make sure you do your own work. Take your time too—thoughtful, complete answers are better than providing a correct answer without showing work. There are some questions in which we’re more interested in what you have to say and how you reason, than a particular answer.

There are three different exams containing nine problems each. Some problems have three, four, or five small subproblems.

Vocabulary

Make sure you are familiar with the vocabulary of Informatics. I’m including what I believe to be important words, but this list is in no way exhaustive: Informatics, model, Information paradox, logic, satisficing, HTML, algorithm, heuristic, digital convergence, kilo, mega, giga, tera, peta, bit, Byte, bug, material implication, gate, circuit, syntax, semantics, tautology, contradiction, contingency, entropy, networks, OSI, peer-to-peer, client/server, luddite, Bus

Concepts

There are a number of important concepts you should be familiar with. I’m including again some of the most important:

• Analog vs. Digital
• Adding structure to problem solving
• Social consequences, often unforeseen, of technology on society: ownership, privacy, security
• Digitally representing analog information
• Structure
• Digital truth vs. Reality
• Logic
• Propositional Logic (PL) \( (\land, \lor, \neg, \rightarrow) \)
• Circuits AND-gate, OR-gate, NOT-gate and building circuits
• Truth tables for PL statements
• Bugs (This one is for Schyler)
• Modeling (relationally, statistically—specifically linear, OSI and its layers)
• Entropy
• SQL, Unix commands (basic)
• Administration on peer-to-peer, client/server
• Base Conversions

Sample Questions

1. Consider the following propositional statement:

\[ G = (\neg A \land B) \lor (\neg B \land A) \]

(a) What is the truth table for the expression?

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>G</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

(b) Draw the circuit for this propositional statement.

(c) Assume \( A \) is True and \( B \) is False. What is the value of statement? From the truth table it’s True

(d) Is the statement a tautology or contradiction or contingency? It’s a contingency, because it’s neither all true (tautology) or false (contradiction).

2. Assume you’ve been hired by an agricultural firm to model the relationship between how much fertilizer used vs. how many ears of corn a corn plant yields. The table you’ve been given access to is given below:

<table>
<thead>
<tr>
<th>TAccident</th>
<th>Fertilizer</th>
<th>EarsOfCorn</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>9</td>
<td></td>
</tr>
</tbody>
</table>

(a) Will there always be a meaningful relationship between the number of manure and vegetable?
(b) Find a linear model where EarsOfCorn is a function of Fertilizer.

(c) Hypothetically, you can find how much manure you’d need to produce 100 ears of corn.

(d) Would you advise the firm to put 100 lbs. of manure per plant? Why or Why not?

3. For this problem assume you have the relation below—the Team_1 attribute has beaten Team_2 attribute.

<table>
<thead>
<tr>
<th>FTems</th>
<th>Team_1</th>
<th>Team_2</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>D</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>E</td>
<td></td>
</tr>
</tbody>
</table>

(a) Find all the teams that lost to A (SQL)
(b) Find all the teams that beat C (SQL)
(c) Find all the teams that lost to the team that lost to team A (SQL)