Midterm (Sample)

CS105: Great Insights in Computer Science
Michael L. Littman, Fall 2007

Rules

• Please write your answers directly on the pages.
• You may use your notes, your book, a calculator.
• You may **not** discuss the test with anyone else, including online sources (via cell phone, computer).
• You may ask clarification questions during the exam.
1. Pictures

A. A 4MP digital camera has an image size of 2289 x 1712 pixels. How many bits is a picture from this camera?

B. The Octopus Song is 3 minutes and 1 second long. If the audio is stored at “CD quality” (192 Kbps), how many bits does it take to store the audio portion of the song?

2. Logical Construction

Write a logical expression (C = something that can include As, Bs, nots, ands, and ors) to match the truth table.

A. | A | B | C |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>False</td>
<td>False</td>
<td>True</td>
</tr>
<tr>
<td>False</td>
<td>True</td>
<td>True</td>
</tr>
<tr>
<td>True</td>
<td>False</td>
<td>False</td>
</tr>
<tr>
<td>True</td>
<td>True</td>
<td>False</td>
</tr>
</tbody>
</table>

B. | A | B | C |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>False</td>
<td>False</td>
<td>True</td>
</tr>
<tr>
<td>False</td>
<td>True</td>
<td>False</td>
</tr>
<tr>
<td>True</td>
<td>False</td>
<td>True</td>
</tr>
<tr>
<td>True</td>
<td>True</td>
<td>True</td>
</tr>
</tbody>
</table>
3. Binary Numbers

(A) What is the decimal number 157 in binary?

(B) What is 11100111 in decimal?

(C) What is the two’s complement of 11100111?

4. Binary Addition

Add these numbers, show the carries.

\[
\begin{array}{c}
00100111 \\
+ 11001010 \\
\hline
10000011 \\
+ 11010111
\end{array}
\]
5. Relay Circuit

Fill in the truth table for this relay circuit.

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>False</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>False</td>
<td>True</td>
<td></td>
</tr>
<tr>
<td>True</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>True</td>
<td>True</td>
<td></td>
</tr>
</tbody>
</table>

6. If-Then-Else-32

An “if-then-else” gate takes three bits as input, A, B, and C. It produces one bit of output, which is equal to B’s value if A is True and C’s if A is False. It can be constructed out of 1 not, 2 ands, and an or.

If we generalize to an “if-then-else-k” gate, A is still one bit, but B and C are now k bits each. It produces k bits of output, which are equal to B if A is True and C if A is False. It can be constructed out of k separate “if-then-else” gates.

How many nots, ands, and ors does it take to make an “if-then-else-32” gate?
7. How Many Pops?
A. When the flag is clicked, how many times will the pop sound happen before everything stops?

B. What value will A have at the end?

![Image showing Scratch code logic for the problem.

8. What’s Missing?

(A) 0-70420-00710-? (gefilte fish)

(B) 0-53?39-65583-4 (CHiPS movie)

(C) ?-52695-23657-0 (Halloween kit)

What will $E$ be after each of these short machine-language programs are executed?

(A) $\begin{align*}
\text{acc} &= \text{not } A \\
E &= \text{acc} \\
\text{acc} &= B \\
\text{acc} &= \text{acc or } C \\
E &= \text{acc and } E
\end{align*}$

(B) $\begin{align*}
\text{acc} &= C \\
\text{acc} &= \text{acc or } D \\
\text{acc} &= \text{acc and } A \\
\text{acc} &= \text{acc and } B \\
E &= \text{acc}
\end{align*}$

10. Halt or Not?

For what values of “?” does each of these Scratch scripts halt? (“*” means multiplication.)

(A) $\begin{align*}
\text{repeat until } \text{count} < 1 \\
\text{play sound } \text{pop} \text{ and wait} \\
\text{change count by } 2
\end{align*}$

(B) $\begin{align*}
\text{repeat until } \text{count} < \text{count} - 1 \\
\text{play sound } \text{pop} \text{ and wait} \\
\text{change count by } 2
\end{align*}$

(C) $\begin{align*}
\text{repeat until } \text{count} < 64 \\
\text{play sound } \text{pop} \text{ and wait} \\
\text{set count to } \text{count} + 2
\end{align*}$

(D) $\begin{align*}
\text{repeat until } \text{count} > 1024 \\
\text{play sound } \text{pop} \text{ and wait} \\
\text{set count to } \text{count} - 1
\end{align*}$