CSci 2011H Quiz 2 Solutions

Q1. Sets

Given S = {1, 2, 3}, P = {1, 2, A, B}, and R = P(S). 
R is then equivalent to {∅, {1}, {2}, {3}, {1, 2}, {1, 3}, {2, 3}, {1, 2, 3}}.

The intersection S ∩ P is the set containing all elements that are members of both S and P (elements can be in any order). So S ∩ P = {1, 2}.

The union S ∪ P is the set containing all elements that are members of either S or P or both. Then S ∪ P = {1, 2, 3, A, B}.

S ∪ R is the set containing all elements of S or R or both. Note that S and R do not have any members in common ({1} ≠ 1). So S ∪ R = {1, 2, 3, ∅, {1}, {2}, {3}, {1, 2}, {1, 3}, {2, 3}, {1, 2, 3}} (elements in any order). Thus there are 11 elements in S ∪ R.

Since S and R have no members in common, S ∩ R = ∅.

Q2. Functions

<table>
<thead>
<tr>
<th>Function</th>
<th>1-to-1?</th>
<th>Onto?</th>
</tr>
</thead>
<tbody>
<tr>
<td>f(x) = x - 3</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>g(x) = 2x - 1</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>f(x) = [1.5x] - 1</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>g(x) = x^2 - 3</td>
<td>N</td>
<td>N</td>
</tr>
</tbody>
</table>

For f(x) = x - 3, if f(x) = f(y), then x - 3 = y - 3, which implies that x = y, so the function is one to one. Also, given some element y in the codomain of f(x), we can find an element x in the domain such that f(x) = y; that element is y + 3. So the function is onto.

For g(x) = 2x - 1, f(x) = f(y) implies that 2x - 1 = 2y - 1 which implies that x = y, so one to one. However, since for any even integer, there does not exist an x such that f(x) equals that number, the function is not onto.

For f(x) = f(x) = [1.5x] - 1, take x to be an even integer, some 2n. Then f(x) = 3n - 1 (note that f(x) is one less than a multiple of three). We can see this is a one to one function, so f(x) is one to one for even integers. Now take x to be odd, some 2m - 1. Then f(x) = 3m - 2 - 1 = 3(m - 1) (exactly a multiple of three). This is also one to one, so f(x) is one to one for odd integers. Lastly, because f(even) cannot ever equal f(odd) since they have different remainders when divided by three, we know every element in the domain maps to a unique value in the codomain, so f(x) is one to one. But for every integer that is two less than a multiple of three, there doesn't exist an x such that f(x) equals that integer, so it is not onto.
For $g(x) = x^2 - 3$, look at $f(2) = f(-2) = 1$, though $x \neq y$. Thus it isn't one to one. Also, any integer less than -3 cannot be obtained from the function, so it is not onto either.

**Q3. Counting**

Since the set of all prime numbers less than 1,000,000 is finite with some finite cardinality $n$, then the power set of that set has cardinality $2^n$, and so is also finite.

<table>
<thead>
<tr>
<th>$\frac{1}{2}$</th>
<th>$\frac{1}{3}$</th>
<th>$\frac{1}{4}$</th>
<th>$\frac{1}{5}$</th>
<th>$\frac{1}{6}$</th>
<th>$\frac{1}{7}$</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\frac{2}{3}$</td>
<td>$\frac{2}{4}$</td>
<td>$\frac{2}{5}$</td>
<td>$\frac{2}{6}$</td>
<td>$\frac{2}{7}$</td>
<td>...</td>
<td></td>
</tr>
<tr>
<td>$\frac{3}{4}$</td>
<td>$\frac{3}{5}$</td>
<td>$\frac{3}{6}$</td>
<td>$\frac{3}{7}$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\frac{4}{5}$</td>
<td>$\frac{4}{6}$</td>
<td>$\frac{4}{7}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\frac{5}{6}$</td>
<td>$\frac{5}{7}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\frac{6}{7}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The set of positive rational numbers less than one is countable. We can show this by organizing the numbers as fractions like in the chart to the left, then counting the elements in each column starting with the column containing only $\frac{1}{2}$ and moving to the right, not counting repeats.

The set of all sets of integers that sum to a negative number is uncountable, similar to how the set of all sets of integers is uncountable. A diagonalization proof can be used to show this.

The set of all possible Java programs is countable. To understand this, think about the fact that computer programs in a particular language can be thought of as strings of characters from a finite alphabet.

**Q4. Proofs and Derivation (Lots of examples!!!)**

Many different types of proofs are possible here. Below are several examples of valid proofs:

**Proof 1** (proof by cases)

Let $(x \oplus y) \land (y \oplus z)$.

$$(x \oplus y) \land (y \oplus z) \equiv ((x \land \neg y) \lor (\neg x \land y)) \land ((y \land \neg z) \lor (\neg y \land z))$$

**Case 1** ($y = T$):

$$(x \land F) \lor (\neg x \land T) \land ((T \land \neg z) \lor (F \land z))$$

$$\equiv (F \lor \neg x) \land (\neg z \lor F)$$

$$\equiv (\neg x \land \neg z)$$

$$\equiv (\neg x \land \neg z) \lor (x \land z)$$

$$\equiv (\neg (x \oplus z))$$

$$\equiv \neg (x \oplus z)$$

**Case 2** ($y = F$):

$$(x \land T) \lor (\neg x \land F) \land ((F \land \neg z) \lor (T \land z))$$

$$\equiv ((x \lor F) \land (F \lor z))$$

$$\equiv (x \land z)$$

$$\equiv (x \land z) \lor (\neg x \land \neg z)$$

$$\equiv (x \oplus \neg z)$$
\[ \equiv \neg(x \oplus z) \]
\[ \therefore (x \oplus y) \land (y \oplus z) \rightarrow \neg(x \oplus z) \]

Proof 2 (direct proof)
Let \((x \oplus y) \land (y \oplus z)\).
\[
(x \oplus y) \land (y \oplus z) \equiv ((x \land \neg y) \lor (\neg x \land y)) \land ((y \land \neg z) \lor (\neg y \land z))
\]
\[\equiv ((x \land \neg y) \land ((y \land \neg z) \lor (\neg y \land z))) \lor ((\neg x \land y) \land ((y \land \neg z) \lor (\neg y \land z)))
\]
\[\equiv ((x \land \neg y) \land (y \land \neg z)) \lor ((\neg x \land y) \land (\neg y \land z))
\]
\[\equiv F \lor (x \land \neg y \land z) \lor (\neg x \land y \land \neg z) \lor F
\]
\[\equiv (x \land \neg y \land z) \lor (\neg x \land y \land \neg z)
\]
If \((x \land \neg y \land z) \lor (\neg x \land y \land \neg z)\) is true, then \((x \land z) \lor (\neg x \land \neg z)\) is also true.
\[
(x \land z) \lor (\neg x \land \neg z) \equiv \neg(x \oplus z).
\]
Therefore, \((x \oplus y) \land (y \oplus z) \rightarrow \neg(x \oplus z)\).

Proof 3 (by contradiction)
Let \((x \oplus y) \land (y \oplus z)\). Assume \((x \oplus z)\). Then, either \(x\) is true and \(z\) is false, or \(z\) is true and \(x\) is false. Suppose \(x\) is true and \(z\) false. Then for \((x \oplus y)\) to be true we need \(y\) to be false. But then \((y \oplus z)\) is false, which is a contradiction. Now suppose the other case is true- \(z\) is true and \(x\) is false. For \((x \oplus y)\) to be true, \(y\) must be true this time. But then \((y \oplus z)\) is false again! Again, we have a contradiction. Hence our assumption is false. That is to say, \(\neg(x \oplus z)\). So \((x \oplus y) \land (y \oplus z) \rightarrow \neg(x \oplus z)\).

Proof 4
We want to show that \((x \oplus y) \land (y \oplus z) \rightarrow \neg(x \oplus z)\) is always true.
\[
(x \oplus y) \land (y \oplus z) \rightarrow \neg(x \oplus z) \equiv \neg((x \oplus y) \land (y \oplus z)) \lor \neg(x \oplus z)
\]
\[\equiv \neg(x \oplus y) \lor \neg(y \oplus z) \lor \neg(x \oplus z)
\]
\[\equiv (x \leftrightarrow y) \lor (y \leftrightarrow z) \lor (x \leftrightarrow z)
\]
For the above statement to be true, at least two of the three elements must have the same truth value, because \(T \leftrightarrow T\) and \(F \leftrightarrow F\) are true, and \(T \lor p \lor q\) is always true no matter the truth values of \(p\) and \(q\). With three variables, we have four different possibilities: all three are true, two are true and one false, one true and two false, or all false. In all of these possibilities, at least two of the variables have the same truth value. Thus \((x \leftrightarrow y) \lor (y \leftrightarrow z) \lor (x \leftrightarrow z)\) is always true, which implies that \((x \oplus y) \land (y \oplus z) \rightarrow \neg(x \oplus z)\).