Exam 2

Revisiting Exam 1

1. (Average: 6.1, Median: 7) Explain why the set of real numbers x where $-1 \leq x \leq 1$ is not well ordered by $<$. (Expected answers will give a good intuitive reason; better-than-expected answers will provide a convincing proof.)

   A well-written intuitive argument that there is no smallest real number above -1 would be worth 10 points. For 10 points, the intuitive answer must mention that there is a subset of the given set that has no minimum.

Proof by contradiction: Assume by contradiction that, R_1, the set of real numbers between -1 and 1 is well ordered. By the definition of a well-ordered set, all non-empty subsets of R_1 have a minimum element. Consider $S = R_1 - -1$. We know $S \subset R_1$ since every element of S is an element of R_1.

We show that R_1 is not well-ordered by contradiction. By the assumption that R_1 is well ordered, since S is a non-empty subset of R_1 it must have a minimum, m.

Define $m^* = (-1 + m)/2$. Then $m^* < m$ since $m > -1$ and the average of -1 and m must be greater than -1. But $m^* \in S$ since it is a real number between -1 and 1. This contradicts the assumption that m is the minimum of S. This contradicts the assumption that R_1 is well-ordered, since we have shown a nonempty subset of R_1 that has no minimum.

State Machines and Invariants

2. (Average 6.8, Median 7) Suppose $M = (S, G \subseteq S \times S, q_0 \in S)$ is a state machine, and R is the set of reachable states in M. For each of the statements below, indicate if the implication stated is valid (must always be true) or invalid (might be false). Provide a short justification supporting your answer.

a. If $R = M$ (namely all the states are reachable), then $G \cup \{(q_0, q_0)\}$ is a surjective relation.

   Circle one: ***Valid*** or *Invalid*

   Justification: Since all the states are reachable, there is at least one arrow into every state in G (except the start state q_0). Thus, $G \cup \{(q_0, q_0)\}$ is a surjective relation because it includes an arrow into q_0, as well as all the arrows in G which must include at least one arrow into every other state.

b. If $P(q)$ is a *preserved invariant* for machine $M = (S, G, q_0)$, and $P(q)$ is false for some reachable state $q \in R$, then $P(q_0)$ is false.

   Circle one: **Valid** or *Invalid*

   Justification: If $P(q_0)$ is true, by the invariant principle, $P(q)$ must hold for all $q \in R$. So, it must be the case that $P(q_0)$ is false if there is any reachable state where $P(q)$ is false.

c. If R is a finite set, then G is a finite set as well. (Note that as a subset of $S \times S$, G is a set itself.)

Circle one: *Valid* or **Invalid**

Justification: Here's one counterexample: $S = \mathbb{N}, G = \{(n, 2n) \,|n \in \mathbb{N}, n \geq 1\}$. G is infinite, but $R = \{0\}$ since $q_0 = 0$ and there are no edges from 0.

Recursive Data Types

Consider the recursive data types, *NBF* (short for Nand-Based Formula) defined by:

- **Base:** *True* is a *NBF*.
- **Base:** *False* is a *NBF*.
- **Constructor:** for all *NBF* objects f_1, f_2, *NAND*(f_1, f_2) is a *NBF*.

The Value of a *NBF*, f, is logical Boolean value of it when evaluated as a logical formula based on NAND gates. So, for example,

$$\text{Value}(\textit{NAND}(\textit{NAND}(True, True), False)) = \textbf{True}.$$

3. (Average 5.9, Median 6) Provide a precise and complete definition of Value for all *NBF* objects.

   Value(*True*) = **True**
   Value(*False*) = **False**
   $\textit{Value}(\textit{NAND}(f_1, f_2)) = \text{NAND}(\textit{Value}(f_1), \textit{Value}(f_2))$

4. (Average 6.8, Median 8) Prove by structural induction that for all *NBF* objects f it holds that

   $$\text{Value}(\textit{NAND}(f, f)) = \neg(\text{Value}(f))$$

   where $\neg(Z)$ is simply the logical negation of the logical (**True** or **False**) variable Z. Note that for this problem it is important that you specify all the required steps of the structural induction.

   Prove by structural induction:

   1. Base cases: *True, False*
      a. $f = \textit{True}$: $\textit{Value}(\textit{NAND}(\textit{True}, \textit{True})) = \textbf{False} = \neg(\textit{Value}(\textit{True})) = \neg(\textbf{True})$
      b. $f = \textit{False}$: $\textit{Value}(\textit{NAND}(\textit{False}, \textit{False})) = \textbf{True} = \neg(\textit{Value}(\textit{False})) = \neg(\textbf{False})$
   2. Constructor case: $f = \textit{NAND}(f_1, f_2)$.
      We need to show that $\textit{Value}(\textit{NAND}(f, f)) = \neg(\textit{Value}(f))$. By the definition of *Value*, *Value*(f) could either be **False** or **True**. By the definition of *Value*, $\textit{Value}(\textit{NAND}(f, f)) = \text{NAND}(\textit{Value}(f), \textit{Value}(f))$. The value of $\textit{Value}(f)$ is either **True** and **False**. So, we need to cover two cases, and show the equality holds for both: NAND(**False**, **False**) $= \neg$(**False**) is **True** since NAND(**False**, **False**) $=$ **True**, and NAND(**True**, **True**) $= \neg$(**True**) is also true.

   Note that we did not need to use the induction hypothesis at all in the constructor case proof (that is, it never depends on the property holding for f_1 or f_2, only knowing that $\textit{Value}(f)$ for any *NBF* object is True or False. So, we didn't really need to prove the base case!

Program Verification

Consider the Python program below, that returns the sum of the elements of the input list `p`. You may assume `p` is a non-empty list of natural numbers.

```
def sum_elements(p):
    x = 0
    i = len(p)
    while i > 0:
        i = i - 1
        x = x + p[i]
    return x
```

5. (Average 8.9, Median 9) Complete the definition of the state machine below that models `sum_elements`.

$$
\begin{aligned}
S &= \{(x, i) \mid x, i \in \mathbb{N}\} \\
q_0 &= (0, \texttt{len(p)}) \\
G &= \{(x, i) \rightarrow (x', i') \mid \\
&\quad i > 0 \wedge \\
&\quad i' = i - 1 \;\wedge \\
&\quad x' = x + \texttt{p[i - 1]} \;\}
\end{aligned}
$$

6. (Average 9.8, Median 10) Prove that the state machine (from problem 5) always terminates.

   Since we start in state $(0, \texttt{len(p)})$ and each transition decreases the value of the i part of the state by one, it will reach $i = 0$ in `len(p)` steps. There are no transitions from a state where $i \leq 0$, hence the machine must eventually terminate.

7. (Average 7.1, Median 7) Prove that `sum_elements`, as modeled by the state machine from Problem 6, always returns the sum of the elements of that list. (Hint: for this goal, you need to (1) find an appropriate invariant property (and prove that it is indeed a preserved invariant), (2) show that the property at a final ending state implies correctness, and (3) it also holds over the original state.)

   We prove the sum correctness using the Invariant Principle. For the preserved invariant we choose:

   $$P(q = (x, i)) ::= x = p[i] + p[i+1] + \ldots + p[len(p) - 1].$$

   Our goal is to prove that in all final states, $x = p[0] + p[1] + \ldots + p[len(p) - 1]$. So, we need to show that (1) P is a preserved invariant for M, that (2) $P(\textit{final}) \implies x =$ the sum of all elements in p, and (3) the $P(q_0)$ is true.

   (1) We need to show P is a preserved invariant. All transitions are from $(x, i) \rightarrow (x + p[i-1], i-1)$ when $i > 0$. So, $P(q = (x, i)) = x = p[i] + p[i+1] + \ldots + p[len(p) - 1]$. Because the transition is $q \rightarrow r = (x + p[i-1], i-1])$, we can add $p[i-1]$ to both sides: $x + p[i-1] = (p[i] + p[i+1] + \ldots + p[len(p) - 1]) + p[i-1] = p[i-1] + p[i] + \ldots + p[len(p) - 1)$. This is $P(r = (x + p[i-1], i-1))$, so we have shown the invariant P is preserved.

(2) As we argued for 6, the final states are states where $i = 0$. So, if we are in a final state $q_f = (x, 0)$, $P(q_f) = x = p[0] + p[1] + \ldots + p[len(p) - 1]$. This is the correctness property we need, so we have satisfied $P(\mathit{final}) \implies x =$ the sum of all elements in p.

(3) $P(q_0 = (0, len(p)))$ is true since $x = 0 =$ empty sum. The sume is empty since it is $p[i] + p[i + 1] + \ldots + p[len(p) - 1]$, and $i = len(p)$ which is greater than the last index.

Infinite Cardinalities

8. (Average 9.0, Median 10) For each set defined below, answer if the set is = *Finite* or *Countably Infinite* or *Uncountable*. Support your answer with a convincing and concise justification. Recall that $\mathbb{N}$ is the set of natural numbers, $\mathbb{Z}$ is the set of integers, $\mathbb{R}$ is the set of real numbers, and $pow(A)$ is the set of subsets of A.

a. $pow(Z)$ where Z is the set of all electrons in the Milky Way galaxy.

Circle one: **Finite** or *Countably Infinite* or *Uncountable*

Justification: The number of all electrons finite, and the power set of any finte set is also finite. (We gave full credit to answers that interpreted Z here as $\mathbb{Z}$ (the integers), and explained that $pow(\mathbb{Z})$ is uncountable, so long your justification made it clear, since the question used a confusing notation (and mispelled the name of our galaxy).

b. $pow(\mathbb{R})$

Circle one: *Finite* or *Countably Infinite* or **Uncountable**

Justification: $\mathbb{R}$ is uncountable and we know for all sets $|pow(S)| > |S|$, so $pow(\mathbb{R})$ must be uncountable.

c. $\{(a, b) \mid a \in \mathbb{N}, b \in \mathbb{R}, b = a/2\}$

Circle one: *Finite* or **Countably Infinite** or *Uncountable*

Justification: $\mathbb{N}$ is countably infinite. There is a bijection between $S = \{(a, b) \mid a \in \mathbb{N}, b \in \mathbb{R}, b = a/2\}$ and $\mathbb{N}$: simply map each element of $S = (a, b)$ to the element of $\mathbb{N}$ that matches its a value. Since there is a bijection, the sets have the same cardinality, and S is countably infinite.

