1. **Warm-up Virtual Lab**

Last week, you were asked to setup your Virtual Machine for the labs. Now it’s time to actually get your feet wet! Please download the IPython starter code from Piazza or the course webpage, and answer the following questions:

(a) In Python, you can represent propositions as follows:

\[
x = a \lor \neg b \\
y = \neg a \land b
\]

Listing 1: Propositions

```python
def x(a, b):
    return a or not b
def y(a, b):
    return not a and b
```

Implement the functions \(f\), \(g\), and \(h\), each takes in three parameters, based on the following propositions. Comment on the relationship between each pair.

\[
f = (a \land \neg b) \lor c \\
g = (\neg a \lor b) \land \neg c \\
h = (a \lor c) \land (\neg b \lor c)
\]

(b) Implement the `forall` and `exists` functions, where each function takes in a non-empty list and a predicate (a function that takes in one parameter and returns either True or False). For example, if we apply the function `is_positive` to each element in `lst`, we get False and True, respectively.

Listing 2: Predicate

```python
lst = [-1, 1]
def is_positive(x):
    return x > 0
```

`forall` should return True if all elements in the list satisfy the predicate and False otherwise. Similarly, `exists` should return True if there exists an element in the list that satisfies the predicate and False otherwise. Do not use Python’s built-in `any` and `all` functions.

(c) Implement the `forall` function again, this time using the `exists` function only. Similarly, implement the `exists` function again using only `forall`.

*Hint: you may find higher-order functions and anonymous functions from EECS 61A helpful for this question. For an example, take a look at the IPython Notebook.*
Reminder: When you finish, don’t forget to convert the notebook to pdf and merge it with your written homework. Please also zip the ipynb file and submit it as hw2.zip.

2. Inductive Charging
There are $n$ cars on a circular track. Among all of them, they have exactly enough fuel (in total) for one car to circle the track. Two cars at the same location may transfer fuel between them.

(a) Prove, using whatever method you want, that there exists at least one car that has enough fuel to reach the next car along the track.

(b) Prove that there exists one car that can circle the track, by gathering fuel from other cars along the way. (That is, one car moving and all others stopped). Hint: Use the previous part.

3. Algorithm Correctness
This problem is a gentle introduction to rigorously proving correctness of algorithms.

Algorithm 1: Recursive Sum

```python
sum_of_list(X):
    if X is empty:
        return 0
    else:
        return X[0] + sum_of_list(X[1:])
```

Algorithm 2: Iterative Sum

```python
sum_of_list(X):
    total = 0
    for i = 0 to len(X)-1:
        total = total + X[i]
    return total
```

Algorithm 3: Find a maximal value from an array

```python
find_max(X):
    front = 0;
    end = length(X) - 1;
    while(front != end ) {
        if X[front] <= X[end]:
            front = front +1;
        else:
            end = end - 1;
    }
    return X[front];
```

1 To think about, after you complete this problem: Is your proof constructive or non-constructive? (That is, does it actually point to the exact car that can complete the track, or does it just prove that one such car must exist?) If it’s non-constructive, then how do we actually find this car? (Can we write a program to do this, faster than actually trying every car?)
(a) Prove that Algorithm 1 returns the sum of the list $X$. That is, prove that for all lists $X$ (of length $n$):

$$\text{sum}_\text{of}_\text{list}(X) = \sum_{i=0}^{n-1} X[i]$$

(b) Prove that Algorithm 2 returns the sum of the list $X$. Notice, this algorithm is not recursive, so the previous method of proof won’t work. Hint: Try proving that the loop iteration performs as expected.

(c) Prove that Algorithm 3 returns the maximum value of array $X$.

4. Losing Marbles
Two EECS70 GSIs are playing a game, where there is an urn that contains some number of red marbles (R), green marbles (G), and blue marbles (B). There is also an infinite supply of marbles outside the urn. When it is a player’s turn, the player may either:

(i) Remove one red marble from the urn, and add 3 green marbles.
(ii) Remove two green marbles from the urn, and add 7 blue marbles.
(iii) Remove one blue marble from the urn.

These are the only legal moves. The last player that can make a legal move wins. We play optimally, of course – meaning we always play one of the best possible legal moves.

(a) Prove by induction that, if the urn initially contains a finite number of marbles at the start of the game, then the game will end after a finite number of moves.

(b) If the urn contains 2 green marbles and $B$ blue marbles initially, then who will win the game? Prove it. In this case, does it matter what strategy the players use?

(c) If the urn contains $(R, G, B)$ red, green, and blue marbles initially, then who will win the game? Prove it. In this case, does it matter what strategy the players use?

5. Induction for Real
Induction is always done over objects like natural numbers, but in some cases we can leverage induction to prove things about real numbers (with the appropriate mapping). For example:

Bob the Bug is on a window, trying to escape Sally the Spider. Sally has built her web from the ground to 2 inches up the window. Every second, Bob jumps 1 inch vertically up the window, then loses grip and falls to half his vertical height.

Prove that no matter how high Bob starts up the window, he will always fall into Sally’s net in a finite number of seconds.

6. Hit or Miss?
State which of the proofs below is correct or incorrect. For the incorrect ones, please explain clearly where the logical error in the proof lies. Simply saying that the claim or the induction hypothesis is false is not a valid explanation of what is wrong with the proof. You do not need to elaborate if you think the proof is correct.

(a) Claim: For all positive numbers $n \in \mathbb{R}$, $n^2 \geq n$.

Proof: The proof will be by induction on $n$.

Base Case: $1^2 \geq 1$. It is true for $n = 1$. 

EECS 70, Fall 2014, Homework 2
Inductive Hypothesis: Assume that \( n^2 \geq n \).

Inductive Step: We must prove that \((n+1)^2 \geq n + 1\). Starting from the left hand side,
\[
(n + 1)^2 = n^2 + 2n + 1 \\
\geq n + 1.
\]

Therefore, the statement is true. \(\Box\)

(b) Claim: For all negative integers \( n, -1 - 3 - \ldots + (2n + 1) = -n^2 \).

Proof: The proof will be by induction on \( n \).

Base Case: \(-1 = -(-1)^2 \). It is true for \( n = -1 \).

Inductive Hypothesis: Assume that \(-1 - 3 - \ldots + (2n + 1) = -n^2 \).

Inductive Step: We need to prove that the statement is also true for \( n - 1 \) if it is true for \( n \), that is, \(-1 - 3 - \ldots + (2(n - 1) + 1) = -(n - 1)^2 \). Starting from the left hand side,
\[
-1 - 3 - \ldots + (2(n - 1) + 1) = (-1 - 3 - \ldots (2n + 1)) + (2(n - 1) + 1) \\
= -n^2 + (2(n - 1) + 1) \quad \text{(Inductive Hypothesis)} \\
= -n^2 + 2n - 1 \\
= -(n - 1)^2
\]

Therefore, the statement is true. \(\Box\)

(c) Claim: For all positive integers \( n, \sum_{i=0}^{n} 2^{-i} \leq 2 \).

Proof: We will prove a stronger statement, that is, \( \sum_{i=0}^{n} 2^{-i} = 2 - 2^{-n} \), by induction on \( n \).

Base Case: \( n = 1 \leq 2 - 1 \). It is true for \( n = 1 \).

Inductive Hypothesis: Assume that \( \sum_{i=0}^{n} 2^{-i} = 2 - 2^{-n} \).

Inductive Step: We must show that \( \sum_{i=0}^{n+1} 2^{-i} = 2 - 2^{-(n+1)} \). Starting from the left hand side,
\[
\sum_{i=0}^{n+1} 2^{-i} = \sum_{i=0}^{n} 2^{-i} + 2^{-(n+1)} \\
= (2 - 2^{-n}) + 2^{-(n+1)} \quad \text{(Inductive Hypothesis)} \\
= 2 - 2^{-(n+1)}.
\]

Since \( \sum_{i=0}^{n} 2^{-i} = 2 - 2^{-n} \leq 2 \), the claim is true. \(\Box\)

(d) Claim: For all nonnegative integers \( n, 2n = 0 \).

Proof: We will prove by strong induction on \( n \).

Base Case: \( 2 \times 0 = 0 \). It is true for \( n = 0 \).

Inductive Hypothesis: Assume that \( 2k = 0 \) for all \( 0 \leq k \leq n \).

Inductive Step: We must show that \( 2(n+1) = 0 \). Write \( n + 1 = a + b \) where \( 0 < a, b \leq n \). From the inductive hypothesis, we know \( 2a = 0 \) and \( 2b = 0 \), therefore,
\[
2(n + 1) = 2(a + b) = 2a + 2b = 0 + 0 = 0.
\]

The statement is true. \(\Box\)
(e) **Claim:** Every positive integer \( n \geq 2 \) has a unique prime factorization.

In other words, let \( 2 \leq p_1, p_2, \ldots, p_i \leq n \) be all prime numbers that divide \( n \), there is only one unique way to write \( n \) as a product of primes,

\[
n = p_1^{d_1} \cdot p_2^{d_2} \cdots p_i^{d_i},
\]

where \( d_1, d_2, \ldots, d_i \in \mathbb{N} \).

**Proof:** We will prove by strong induction on \( n \).

**Base Case:** 2 is a prime itself. It is true for \( n = 2 \).

**Inductive Hypothesis:** Assume that the statement is true for all \( 2 \leq k \leq n \).

**Inductive Step:** We must prove that the statement is true for \( n + 1 \). If \( n + 1 \) is prime, then it itself is a unique prime factorization. Otherwise, \( n + 1 \) can be written as \( x \times y \) where \( 2 \leq x, y \leq n \). From the inductive hypothesis, both \( x \) and \( y \) have unique prime factorizations. The product of unique prime factorizations is unique, therefore, \( n + 1 \) has a unique prime factorization. \( \square \)

7. **Magic Lawn Mower**

There is a magic lawn mower which can remove a perfect square foot of lawn underneath it in the blink of an eye. To make lawn-mowing less boring, the owner decides to cut each \( 3 \times 3 \) square feet of his lawn in a ‘2’ or ‘5’ pattern. Can you prove that he can cut through any \( 3 \times 3 \) and \( 3^2 \times 3^2 \) square feet of lawn \((n \in \mathbb{Z}^+)\) with one continuous walk without cutting any same square foot twice? Figure 1 shows his walking patterns for \( 3 \times 3 \) and \( 3^2 \times 3^2 \) square feet.

(a) \( 3 \times 3 \) ft\(^2\)

(b) \( 9 \times 9 \) ft\(^2\)

![Figure 1: Example walking patterns](image)

8. **Series Induction**

For all \( n \in \mathbb{N} \), let \( a_n \) be the number of subsets of \( \{1, 2, \cdots, n\} \) that do not contain any two consecutive numbers (including the empty set).

(a) Show that \( a_n \) is the \((n+2)\)-th Fibonacci number.

(b) Prove that \( a_n = \frac{\phi^{n+2} - \bar{\phi}^{n+2}}{\sqrt{5}} \), where \( \phi = \frac{1+\sqrt{5}}{2} \) is the golden ratio and \( \bar{\phi} = \frac{1-\sqrt{5}}{2} \) is the conjugate of \( \phi \).

9. **Write Your Own Problem**

Write your own problem related to this week’s material and solve it. You may still work in groups to brainstorm problems, but each student should submit a unique problem. What is the problem? How to formulate it? How to solve it? What is the solution?