CPSC-354 Report

Emma Garofalo
Chapman University

September 8, 2024

Abstract

This document outlines what has been learned week by week through this class. For now, it only contains the information learned for week one.

Contents

1 Introduction

Welcome to my class report! As the class progresses, I will add my learning from each week. For now, it only has week one.

2 Week by Week

2.1 Week 1

Notes

This week we discussed the foundation of what the class is about. The main idea was that this class is largely about the intersection of math and programming, beginning with revisiting the principles we learned in discrete mathematics. We also learned about LaTeX, which we will be using throughout the semester to edit documents like this one. The general idea of the week was setting all of us students up for what to expect throughout the semester.

We also covered the topics of Formal Systems, which are explained in more detail in the below section. And how to determine the relations between a Lean proof and a Math proof.

Homework

The reading that we had to cover for homework discussed the MUI problem, which helps us break down what a formal system is, and how these attributes can be seen in mathematics like discrete math. A formal system carries the requirement of formality, which states that you must not do anything outside of the set rules. Theorems, axioms, rules of production, and the decision procedure are all key parts of a formal system.

We also had to complete the tutorial world of the natural number game. Here are the solutions for levels 5-8.

Level 5

Goal: a+(b+0)+(c+0)=a+b+c

Solution:

```
rw[add_zero]
rfl
```

Level 6

Goal: a+(b+0)+(c+0)=a+b+c

Solution:

```
rw[add_zero c]
rw[add_zero]
rfl
```

Level 7

Goal: succ n = n + 1

Solution:

```
rw[one_eq_succ_zero]
rw[add_succ]
rw[add_zero]
rfl
```

How is this lean proof related to mathematics proofs?:

This Lean solution demonstrates the definition of Natural Numbers. Natural numbers are defined by two principles: 1. "There is a special natural number, called zero, denoted by 0." 2. "For any natural number n, there is a unique next natural number, called the successor of n."

The first step completed in the Lean proof uses rule 2 by breaking the number 1 on the right side of the equation into the successor of 0. The next line uses the property of addition for successors. Showing that n + succ(m) = succ(n+m) The additive identity property is then used to simplify the equation so that both sides are now equal to each other, proving the validity of the statement that the succ n = n + 1

Level 8

Goal: 2 + 2 = 4

Solution:

```
nth_rewrite 2[two_eq_succ_one]
rw[one_eq_succ_zero]
rw[four_eq_succ_three]
rw[three_eq_succ_two]
```

```
nth_rewrite 2[two_eq_succ_one]
rw[one_eq_succ_zero]
rw[two_eq_succ_one]
rw[one_eq_succ_zero]
rw[add_succ]
rw[add_succ]
rw[add_zero]
rfl
```

I learned a lot from this homework. It basically acted as a refresher for discrete mathematics, and how what seems like such a simple solution is much more complicated than you think it is. It also shows the various ways that you can derive the same solutions.

Comments and Questions

This week provided me with a good refresher of the discrete mathematics class that I took a while ago. It brought to my attention how much there is a crossover between math and code, and I am so excited to explore that in this class.

My question for the week relates to the foundation of mathematics, and where it has all evolved from there. We refreshed on discrete math, which shows us why even the simplest mathematic proofs are valid. And it makes me wonder how mathematics has evolved so much since then. We have such calculated math that has all built off of these proofs. Seeing how much math has evolved since then, does that mean that math will continue to evolve in complexity forever? As we create new technologies and understand our world better, will more complicated relationships continue to be found?

2.2 Week 2: Recursion

Notes

The main topic of this week was recursion, and how it can be visualized with the "Tower of Hanoi" exercise. We learned that the definition of recursion is nesting, and variations on nesting. Recursive definitions are defined in simpler terms of itself, but they never lead to infinite regress or paradox. The "Tower of Hanoi" showed us this well by demonstrating that once you find a base case (a simpler way to break the problem down), then you can use that and the successor of that to solve recursively. We considered the problem from the bottom disk up, and observed how both a stack machine and a rewriting machine can find the efficient recursive solutions.

Homework

The reading that we had for homework was about the "Little Harmonic Labyrinth". This story was a complex example of how recursion is nesting, and variations on nesting. Another real life example that this reading provided was postponing the completion of a task in favor of completing a simpler task, which I think is something that humans naturally do all the time.

We also completed the addition world of the Natural Number Game, and some solutions are outlined below.

Level 1

Goal: 0 + n = n

Solution:

```
induction n with d hd
rw[add_zero]
```

```
rfl

rw[add_succ]
rw[hd]
rfl
```

Level 2

Goal: succ a + b = succ (a + b)

Solution:

```
induction b with n hn
rw[add_zero]
rw[add_zero]
rfl

rw[add_succ]
rw[hn]
rw[add_succ]
rfl
```

Level 3

Goal: a + b = b + a

Solution:

```
induction b with d hd
rw[add_zero]
rw[zero_add]
rfl

rw[add_succ]
rw[hd]
rw[succ_add]
rfl
```

Level 4

Goal: a + b + c = a + (b + c)

Solution:

```
induction c with d hd
rw[add_zero]
rw[add_zero]
rfl

rw[add_succ]
rw[hd]
rw[add_succ]
rw[add_succ]
rfl
```

How is this lean proof related to mathematics proofs?:

This lean proof begins with induction on natural numbers, performed on the varuable c. This mathematics proof involves breaking the problem into a base case, showing that the property holds for the smallest number, usually being 0. Then the inductive step is performed, which assumes that the property holds for the successor of the arbitrary natural number. It uses the inductive step performed before.

Then the identity property for addition is performed, showing that a + 0 = a. Once the base case is proven, the successor must be proven as well. The property of addition for successors simplifies the goal so that the inductive hypothesis can be substituted. With another simplification from the addition of successors, both sides of the equation are shown to be equivalent.

Level 5

Goal: a + b + c = a + c + b

Solution:

```
induction c with d hd
rw[add_zero]
rw[add_zero]
rfl

rw[add_succ]
rw[add_succ]
rw[hd]
rw[succ_add]
rfl
```

This homework was all about recursion. It refreshed my knowledge on how to apply induction to reach the goal.

Comments and Questions

The "Tower of Hanoi" problem reminded me the give and take when it comes to finding recursive solutions. Oftentimes, a recursive solution is a more efficient one, but the time it takes to think up this solution and the possible complexity of it needs to be taken into consideration.

My question of the week relates to this. Is there a key way to always identify that the most efficient solution to a particular problem is recursion? I feel like when a problem becomes more complicated, I struggle to identify if it can be solved recursively, and because of this, I try multiple other solutions before and waste a lot of time coming to a recursive solution.

2.3 ...

...