CSci 2041 - Advanced Programming Principles

Spring 2017

Homework 3

Name: Tiannan Zhou

Email: zhou0745@umn.edu ID:5232494

**Question 1:**

We would like to show that

The principle of induction for natural numbers is

In this case, our property P is defined as

is

Our induction proof would have two cases:

* : show that

by definition of

* : show that where holds by the inductive hypothesis.

by definition of

by property of addition and subtraction

by inductive hypothesis

by property of multiplication and power

has been proven.

**Question 2:**

We would like to show that

The principle of induction for this typeis

In this case, our property P is defined as

is

Our induction proof would have two cases:

* : show that

by definition of

by definition of power

by definition of

* : show that when holds by the inductive hypothesis.

by the definition of

by induction hypothesis

by the definition of multiplication and power

by the definition of

has been proven.

**Question 3:**

We would like to show that

The principle of induction for list is

In this case, our property is defined as

Our induction proof would have two cases:

by properties of lists

by properties of addition

by definition of

by definition of

by induction hypothesis

by definition of

has been proven.

**Question 4:**

We would like to show that

The principle of induction for list is

In this case, our property is defined as

Our induction proof would have two cases:



by definition of



by the definition of

by the property ofproven in Question 3

by induction hypothesis

by definition of

by property of addition

by definition of

has been proven.

**Question 5:**

The principle of induction for list is

We would like to prove first while

is

Our induction proof for would have two cases:



by definition of

by properties of lists



by definition of

by induction hypothesis

by properties of lists

has been proven.

Now we would like to prove while

is

Our induction proof for would have two cases:



by property ofproven by

­ by properties of lists

by properties of lists

by definition of

by property of proven by



by property ofproven by

­ by properties of lists

by definition of

by property of proven by

by induction hypothesis

by property of proven by

by property of lists

by definition of

by property proven by

has been proved.

**Question 6:**

We would like to show

The principle of induction for list is

In this case, our property is defined as

We just need to prove the situation that B is also true when A is true in order to prove that is true (by property of implication). Our induction proof lists below:

Base case:

to prove

by definition of

by definition of sorted

by property of lists

by definition of

Thus, is true, P([ ]) has been proven.

to prove is

by definition of

Case 1:

by definition of

by definition of

by

by definition of

by property of Boolean expression

Case 2:

by definition of

by definition of

by property of lists

by definition of

by

by definition of

by property of Boolean expressions

Thus, is true, P([x]) has been proven.g

Inductive case:

to prove the statement

We assume to make sure can be true. (We don’t care the situation when is false because the implication statement would always be true if the left-hand side is false (by property of Propositional Logic)) Thus, my proof below would just show the cases under condition that the left-hand is always true.

Case 1:

Left hand:

by property of Boolean expression

by

by definition of

by definition of

Thus, is true under the condition of

Case 2:

Right hand:

by definition of place

by definition of and assumption

by

by inductive hypothesis and assumption

by property of Boolean expressions

Thus, has been proven.

**Question 7:**

1. Since the function would place the element into the list where it found the first element currently in list which is greater than x. Thus, in the result list from function , all the elements on the left of new inserted element are smaller than . Therefore, function ‘s assumption does not matter in this scenario due to the premise mentioned above made by the function . Function would skip all the elements on the left of the target element (since they are smaller than the element we’re looking for), then find and return true.
2. It doesn’t need. We can directly assume is true when we prove the statement. Because of the property of implication statement that when the left-hand is false, the full single-direction implication statement is always true, we just need to prove the situation that the right-hand side is true as well while the left-hand side is true to make sure the full statement is true. The situation of left-hand side being false does not matter.