**Software Analysis & Verification** **Due: Mar 24, 2020**

# Homework 2

*Instructor: Fei He* 沈冠霖 (2017013569)

*TA: Jianhui Chen, Fengmin Zhu*

*Read the instructions below carefully before you start working on the assignment:*

- In this assignment, you are asked to both typeset your answers in the attached LaTeX source file, and also complete the missing code in the Dafny file `PA.dfy`. Make sure that the Dafny complier can type check your code! When done, compile this file to a PDF. Compress the PDF and `PA.dfy` to an `.zip` archive and hand it to Tsinghua Web Learning *before the due date.*
- Make sure you fill in your *name* and *Tsinghua ID*, and replace all "`TODO`"s with your solutions.
- Any kind of dishonesty is *strictly prohibited* in the full semester. If you refer to any material that is not provided by us, you *must cite* it.

## Problem 1: Short-Answered Questions

**1-1** Underline all free variables (to be precise, their occurrences) in the following first-order formula:

$$\forall x.(f(x) \land (\exists y.g(x, y, z))) \land (\exists z.g(x, y, z))$$

**Solution** $\forall x.(f(x) \land (\exists y.g(x, y, \underline{z}))) \land (\exists z.g(x, \underline{y}, z))$■

**1-2** Which of the following problems or theories are decidable?

(a) Deciding validity for propositional logic.

(b) Deciding validity for first-order logic.

(c) $T_E$.

(d) $T_{\mathbb{N}}$.

(e) The quantifier-free fragment of $T_A$.

**Solution** (a),(d),(e) ■

**1-3** Find an equivalence relation that is not a congruence relation.

**Solution** 对于集合 S={a,b,c}, 在 S 上定义函数 f:f(a)=c,f(b)=b,f(c)=c, 还有关系 R:{(a,a),(b,b),(a,b),(b,a),(c,c)}

首先，R 是等价关系，可以划分为两个等价类 {a,b} 和 {c}

其次，R 不是共轭关系。一方面，aRb 成立; 另一方面，f(a)=c,f(b)=b,f(a)Rf(b) 不成立;

■

**1-4** Find two distinct equivalence relations such that one refines the other.

**Solution** 对于集合 S={a,b,c}, 在 S 上定义关系 R$_1\{(a,a),(b,b),(c,c)\}$, $R_2 : \{(a,a),(b,b),(a,b),(b,a),(c,c)\}$ R$_1$, $R_2$ 都是等价关系。另一方面，$\hat{R_1} \in \hat{R_2}, R_1 \prec R_2$ ■

**1-5** In the congruence closure algorithm, subterms of a formula are represented by DAGs. Which term does Figure 1 represents? Write it out in a formulaic way.

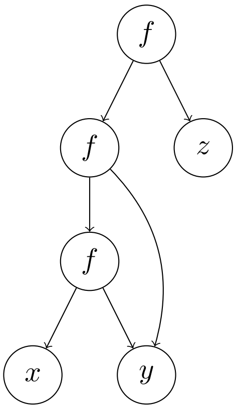

图 1: A subterm.

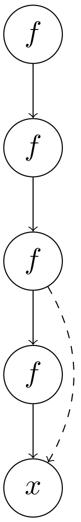

图 2: A DAG.

**Solution** f(f(f(x,y),y),z) ■

**1-6** Figure 2 presents a DAG in an execution of the congruence closure algorithm. The dashed edge was inserted via a union operation. Which congruences classes can you infer from this figure?

**Solution** $\{x, f^2(x), f^4(x)\}$ 和 $\{f^1(x), f^3(x)\}$ 两个共轭类。
首先，有 $\{x, f^2(x)\}, \{f^1(x)\}, \{f^3(x)\}, \{f^4(x)\}$ 四个等价类。
其次, 根据共轭条件, 若有 $xRf^2(x)$, 则必有 $f(x)Rf(f^2(x))$, 也即 $f^1(x)Rf^3(x)$。同理, 有 $f^2(x)Rf^4(x)$
将两条新增关系加入原先的等价关系 R, 再次扩增 R 使其满足等价性和共轭条件,则能得到 $\{x, f^2(x), f^4(x)\}$ 和 $\{f^1(x), f^3(x)\}$ 两个共轭类。■

## Problem 2: Peano Arithmetic

In this problem, we will show that two ways of defining natural numbers are, to some extent, the same – one is using the axioms of Peano Arithmetic, and another is using an inductive set whose elements are what we mean “natural numbers”.

To receive full credit of this problem, you must both:

- complete the proofs in `PA.dfy`, and
- fill in the missing manual proofs in this file.

There is an example that has been done for you. You should read it carefully before you start.

**PA** Recall that *Peano Arithmetic* (PA) is a first-order theory with signature:

$$\Sigma_{PA} : \{0, 1, +, \times, =\}$$

where:

- 0 and 1 are constants
- $+$ and $\times$ are binary functions
- $=$ is a binary predicate

It has the following axioms:

- All of the equality axioms: reflexivity, symmetry, transitivity, and congruence
- *Zero:* $\forall x.\ \neg(x + 1 = 0)$
- *Additive identity:* $\forall x.\ x + 0 = x$
- *Times zero:* $\forall x.\ x \times 0 = 0$
- *Successor:* $\forall x, y.\ (x + 1 = y + 1) \to x = y$
- *Plus successor:* $\forall x, y.\ x + (y + 1) = (x + y) + 1$
- *Times successor:* $\forall x, y.\ x \times (y + 1) = x \times y + x$

It also has an axiom schema for induction:

$$(F[0] \land (\forall x.F[x] \to F[x + 1])) \to \forall x.F[x]$$

The intended interpretation for this theory is the natural numbers with constant symbols 0 and 1, a predicate symbol $=$ taking equality over $\mathbb{N}$, and function symbols $+, \times$ taking the corresponding expected functions over $\mathbb{N}$.

**Natural Numbers as an Inductive Set** On the other hand, we could define natural numbers as an *inductive* set $S$, i.e. a *minimum* set whose elements are generated by using only the following two rules:

- $0 \in S$;
- if $n \in S$, then $\mathsf{Succ}(n) \in S$.

In fact, our familiar $\mathbb{N} = S$ as defined above. Using the above notion, 1 is represented by $\mathsf{Succ}(0)$; 2 is represented by $\mathsf{Succ}(\mathsf{Succ}(0))$; and so on.

The above definition can be easily expressed in Dafny, as an inductive type:

```
datatype Nat = Zero | Succ(n: Nat)
```

We then define the constant 1, as well as the functions for addition and multiplication as follows:

```
function one(): Nat
{
  Succ(Zero)
}

function add(x: Nat, y: Nat): Nat
{
  match(x) {
    case Zero => y
    case Succ(n) => Succ(add(n, y))
  }
}

function mult(x: Nat, y: Nat): Nat
{
  match(x) {
    case Zero => Zero
    case Succ(n) => add(mult(n, y), y)
  }
}
```

In this problem, you will use Dafny to prove that the inductively-defined `Nat` type satisfies the axioms of PA.

**The Calc Statement** Before getting started, you should read section 21.17 of the Dafny manual to understand the basic usage of Calc statements, started with the keyword `calc`.

**Instructions** In `PA.dfy`: Provide the correct pre- and post-conditions for lemmas *Zero* (2-1), *Times zero* (2-3), *Successor* (2-4), *Plus successor* (2-5) and *Time successor* (2-6), and the bodies of your lemmas must satisfy each postcondition. The proof for *Additive identity* (2-2) is provided as an example. If you use an `assume` statement in any of your lemmas, you will receive *partial* (i.e. not full) credits.

Moreover, in this file: Write a *careful manual* proof for each of the lemmas above. Again, the manual proof for *Additive identity* (2-2) is provided as an example. By saying *careful*, we mean that your proof must include all details and you should explain the reasons for each step. Remember that the only thing you know are: (1) the definitions of the functions listed in `PA.dfy`, and (2) a couple of "built-in" proof strategies (supported by Dafny and we admit them) including the (structural) induction, proof-by-contradiction and all equality axioms.

### Proofs

**2-1** *Zero*

证明.

- If x = 0, by definition of 0 and 1, $x + 1 = 0 + 1 = 1 \neq 0$

- Else if x is not 0, then x must be the successor of n, x = Succ(n). By definition of plus, $x + 1 = succ(n) + 1 = succ(n + 1) \neq 0$

  Therefore, $\neg(x + 1 = 0)$ holds for every $x$.

□

**2-2** *Additive identity* (example)

证明. By contradiction. Suppose that $x + 0 = x$ does not held for some $x$. Then, there are only two possible choices:

- $x$ is zero, i.e. $x = 0$.
  By definition of +, we know that $x + 0 = 0 + 0 = 0$. Since $0 = x$, we conclude that $x + 0 = x$, contradiction!
- $x$ is the successor of $n$, i.e. $x = n + 1$.
  By definition of +, we know that $x + 0 = (n + 1) + 0 = (n + 0) + 1$. By inductive hypothesis, $n + 0 = n$. Thus, $(n + 0) + 1 = n + 1 = x$. We again conclude that $x * 0 = 0$, contradiction!

Therefore, $x + 0 = x$ holds for every $x$. □

**2-3** *Times zero*

证明.

- $x$ is zero, i.e. $x = 0$.
  By definition of ×, we know that $x \times 0 = 0 \times 0 = 0$. Since $0 = x$, we conclude that $x \times 0 = 0$!
- $x$ is the successor of $n$, i.e. $x = n + 1$.
  By definition of ×, we know that $x \times 0 = (n \times 0) + 0$. By inductive hypothesis, $n \times 0 = 0$. Thus, $(n \times 0) + 0 = 0 + 0 = 0$. We again conclude that $x + 0 = x$!

  Therefore, $x \times 0 = 0$ holds for every $x$. □

**2-4** *Successor*

证明. - If x = 0, then y has 2 conditions:

  - $y = 0$
    $, x + 1 = y + 1 = 1$
    $, x = y = 0$, right.
  - else, $y = Succ(n)$, then the prerequesite is not right, since
    $y + 1$
    $= Succ(n) + 1$
    $= Succ(n + 1)$, by definition of plus
    but for x,
    $x + 1$
    $= 0 + 1$
    $= Succ(0)$, by lamma 2, which will be proved in task 6

    since$\neg(n + 1 = 0)$,$y + 1 \leq x + 1$, the prerequesite is wrong, then the formula is right.

- else, x = Succ(m),then y has 2 conditions:
  - $y = 0$, the prerequesite is similarly wrong, then the formula is right.
  - else, $y = Succ(n)$,
    $x + 1$
    $= Succ(m) + 1$
    $= Succ(m + 1)$
    $y + 1$
    $= Succ(n) + 1$
    $= Succ(n + 1)$
    Since $x + 1 = y + 1$, then $Succ(m + 1) = Succ(n + 1), m + 1 = n + 1$
    By induction hypothesis, $m = n$. Since$x = Succ(m), y = Succ(n), x = y$

Therefore, $(x + 1 = y + 1) \to x = y$ holds for every $x, y$.

□

**2-5** *Plus successor*

证明. - $x$ is zero, i.e. $x = 0$.

$x + (y + 1)$
$= 0 + (y + 1)$
$= y + 1$,By definition of +
$= (0 + y) + 1$,By definition of +
$= (x + y) + 1$

- $x$ is the successor of $n$,
  $x + (y + 1)$
  $= Succ(n) + (y + 1)$
  $= Succ(n + (y + 1))$,By definition of $+$
  $= Succ((n + y) + 1)$,By inductive hypothesis
  $= Succ(n + y) + 1$,By definition of $+$
  $= Succ(n) + y + 1$,By definition of $+$
  $= (x + y) + 1$

  Therefore, $x + (y + 1) = (x + y) + 1$ holds for every $x, y$. □

**2-6** *Times successor*

**lamma 2** First, we must prove that $x + 1 = Succ(x)$

证明. • $x$ is zero, i.e. $x = 0$.
$x + 1$
$= 0 + 1$
$= 1$
$= Succ(0)$
$= Succ(x)$

- $x$ is the successor of $n$,
  $x + 1$
  $= Succ(n) + 1$
  $= Succ(n + 1)$
  $= Succ(Succ(n))$, by inductive hypothesis
  $= Succ(x)$

  Therefore,$x + 1 = Succ(x)$ is true for all x

**lamma of exchange** Second, we must prove that $x + (y + z) = (x + z) + y$

- $z$ is zero, i.e. $z = 0$.
  $(x + y) + z$
  $= (x + y) + 0 = (x + y)$,by Additive identity
  $= (x + 0) + y$,by Additive identity
  $= (x + z) + y$
- $z$ is the successor of $n$,
  $(x + y) + z$
  $= (x + y) + Succ(n)$

$= (x+y)+(n+1)$,by lamma 2
$= (x+y)+n+1$, by Plus Successor
$= (x+n)+y+1$, by inductive hypothesis
$= Succ((x+n)+y)$, by lamma 2
$= Succ(x+n)+y$
$= (x+n+1)+y$, by lamma 2
$= x+(n+1)+y$, by Plus Successor
$= x+Succ(n)+y$, by lamma 2
$= (x+z)+y$

Therefore,$x+(y+z)=(x+z)+y$ is true for all x,y,z

**Proof** Last, we can prove that $x \times (y+1) = x \times y + x$

- $x$ is zero, i.e. $x = 0$.
  $x \times (y+1)$
  $= 0 \times (y+1)$
  $= 0$, by definition of ×
  $= 0+0$, by definition of +
  $= (0 \times y)+0$, by definition of ×
  $= (x \times y)+x$
- $x$ is the successor of $n$,
  $x \times (y+1)$
  $= Succ(n) \times (y+1)$
  $= (n \times (y+1)) + (y+1)$, by definition of multiply
  $= (n \times y + n) + (y+1)$, by inductive hypothesis
  $= (n \times y + n + y) + 1$, by plus successor
  $= ((n \times y)+y)+n+1$, by exchange lamma
  $= (Succ(n) \times y)+n+1$, by definition of multiply
  $= (Succ(n) \times y)+Succ(n)$, by lamma 2
  $= (x \times y)+x$

Therefore, $x \times (y+1) = x \times y + x$ is true for all x,y □