Recitation 6

Tyler Chan

February 10, 2025

Agenda

- 3 Dafny Examples
- Quiz 4

Dafny Example 1

Java Code:

```
public class Example1 {
    public static int max(int a, int b) {
        if (a > b) {
            return a;
        } else {
            return b;
        }
    }
}
```

Write a dafny file for this code. Starter can be found here.

Dafny Example 1

Dafny Solution:

```
method max(a: int, b: int) returns (m: int)
{
    if a > b {
        return a;
    } else {
        return b;
    }
}
```

Dafny Example 2

Java Code:

```
public class ExampleSumOfDigits {
    /**
     * Returns the sum of the decimal digits
     *   of a nonnegative integer x.
     * For example:
     *   sumOfDigits(0) = 0
     *   sumOfDigits(5) = 5
     *   sumOfDigits(123) = 1+2+3 = 6
     */
    public static int sumOfDigits(int x) {
        if (x < 10) {
            return x;
        } else {
            return (x % 10) + sumOfDigits(x / 10);
        }
    }
}
```

Write a dafny file for this code. Starter can be found here.

Dafny Example 2

Dafny Solution:

```
method SumOfDigits(x: nat) returns (sum: nat)
    requires x >= 0
    ensures sum == SumOfDigitsSpec(x)
    decreases x
{
    if x < 10 {
        return x;
    } else {
        var rest := SumOfDigits(x / 10);
        return (x % 10) + rest;
    }
}

// A pure function that specifies how to compute
//   sum of digits (mathematically)
function SumOfDigitsSpec(x: nat): nat
{
    if x < 10 then x
    else (x % 10) + SumOfDigitsSpec(x / 10)
}
```

Dafny Example 3

Java Code:

```
public class Example3 {
    public static int power(int base, int exponent) {
        int result = 1;
        for (int i = 0; i < exponent; i++) {
            result *= base;
        }
        return result;
    }
}
```

Write a dafny file for this code. Starter can be found here.

Dafny Example 3

Dafny Solution:

```
method power(base: int, exponent: nat) returns (result: int)
    ensures result == pow(base, exponent)
{
    result := 1;
    var i := 0;
    while i < exponent
        invariant 0 <= i <= exponent
        invariant result == pow(base, i)
    {
        result := result * base;
        i := i + 1;
    }
    return result;
}

// A pure function that specifies how to compute powers,
//   this time with recurision
function pow(base: int, exp: nat): int {
 if exp == 0 then 1 else base * pow(base, exp-1)
}
```

Resources

- The full solutions can be found on my repo: https://github.com/DeBestTrap/psoft-recitation-materials/rec6
- Prof. Kuzmin also has dafny examples: https://github.com/KCony/PSoftExamples/tree/master/Dafny

Quiz 4

Do quiz 4 now on Submitty.