Homework #1

CS477 - Fall 2019
Due Feb 22, 2019

Consider the Dafny program provided on the website: *Product1, Product2, Half,* and *DividesIt.*

Prove them correct using Dafny by writing down appropriate loop invariants. You are allowed to only add (side-effect free) annotations to the code— you cannot change the executing code in any way (not even in trivial ways that may seem correct to you).

Submit your program (with proof that Dafny verified it) using a hardcopy printout, and also email your (verified) programs to madhu@illinois.edu in a tarball.

A few notes:

- Product1 and Product2 are inefficient ways to compute the product. Aim is to get you started thinking about inductive invariants and writing them formally in logic and in Dafny. Note that we are using the type nat (natural numbers), a subtype of integers, rather than integers here. Dafny will hence check that every assignment to a natural number necessarily assigns a non-negative number. Also, in Dafny, int and nat types are pure mathematical integers and natural numbers, and don't overflow, etc.

- Half is a program that uses integer division and asks you to write invariants using them. Note the use of "/" (integer division) and "%" (modulo) operators.

- In DividesIt, Dafny needs a lemma to prove the program correct. Lemmas are written as procedures. We have already provided the lemma (make sure you understand it and you agree it's valid). We don't *prove* the lemma but just assume it; later in the course, we will see how to prove lemmas. Also, this program has a `decreases` clause to prove the loop terminates—— ignore understanding this for now. But keep it there so that Dafny does not complain. In the other programs above, Dafny actually uses a simple heuristic to reason that the loops terminate, and hence doesn't require a decreases clause.