Dafny
28/12/2016
Kalev Alpernas
Method Pre and Post-Conditions

• Remember design-by-contract you learned in Software 1?
Method Pre and Post-Conditions

- Remember design-by-contract you learned in Software 1?

- We annotate methods with pre-conditions:

```java
method m(x: int, y: int) returns (r: int)
  requires 0 <= x && 0 <= y

{...}
```
Method Pre and Post-Conditions

• Remember design-by-contract you learned in Software 1?

• We annotate methods with pre-conditions and post-conditions:

```java
method m(x: int, y: int) returns (r: int)
    requires 0 <= x && 0 <= y
    ensures r == 2*x + y
{...}
```
Method Pre and Post-Conditions

• Dafny helps us prove that:

If pre-condition $\varphi$ holds at the start of the method

And the method terminates

Then post-condition $\psi$ holds when the method terminates
Method Pre and Post-Conditions

• Let’s consider method m again:

```plaintext
method m(x: int, y: int) returns (r: int)
  requires 0 <= x && 0 <= y
  ensures r == 2*x + y
{...}
```

• We can use dafny to prove that:
  • If \( x \geq 0 \) and \( y \geq 0 \) when m is invoked
  • And the run of m terminates
  • At the end of the run \( r = 2x + y \)
Loop Invariants

• Loops are hard

• Need to provide Dafny with loop invariants
  
  ```
  while n != 0
      invariant r == x+y-n && 0 <= n
  { ... }
  ```
More

• Assertions
  assert 2 * x + x / x > 3;

• Assumptions
  assume x > 1;

• Predicates and (Pure) Functions
  function min(a: nat, b: nat): nat
  {if a < b then a else b}

  predicate twice(a: nat, b: nat)
  {a==2*b}