15-122: Principles of Imperative Computation

QuickCheck 1 Solutions

Sometimes on Friday, we'll give you one of these quick-checks. These are NOT graded. We're exprimenting with how exactly these will work; part of their purpose is to give the course staff feedback on your performance, part of their purpose is to maintain a record of recitation attendance. Please state your name and recitation section clearly, give it your best shot and hand this paper in to your TA when you're done.

Name:

Andrew ID:

Section (circle one): A B C D E F G H

Simplicio and *Sagredo* (you) are a couple of 122 students. They are trying to complete an exercise, but are a bit stuck. Help them out!

Here's some code that *Simplicio* wrote, but he's unable to fill in the blanks. Show him that *Sagredo* is much smarter than he is!

```
1  int factorial (int n)
2  //@requires n >= 0;
3  {
4     int result = 1;
5     while(n > 0)
6     //@loop_invariant n >= 0;
7     {
8        result = result * n;
9        n = n - 1;
10    }
11    return result;
12 }
```

Salviati, one of the TAs, now gives them a 'magic' function `fact` with the signature:

```
int fact (int n);
```

Salviati: Now you guys can add a post-condition!

Simplicio: Where will the postcondition go?

Sagredo: It's between lines 2 and 3.

Simplicio: I have no idea what it could possibly be, though!

Sagredo: `//@ensures \result == fact(n);`