Practice Quiz 6 Solutions

CSCI 136: Spring 2022

Your name: ____________________

Consider the following code to compute the square of a number $n \geq 0$.

```
/** computes the square of a number
 * @param n the number to be squared
 * @return n*n
 * @pre n >= 0
 * @post returns n*n
 */
public static int square(int n) {
  int ret = 0;
  int odd = 1;
  for(int i = 1; i <= n; i++) {
    ret += odd;
    odd += 2;
  }
  return ret;
}
```

1. What can you say about the value of `ret` after the loop has iterated k times?

   Hint: trace the loop execution for some small values of `n`

   Hint #2: we're looking for a very simple statement about the loop

   Your answer: After the loop iterates k times, `ret` holds the value k^2.

2. Prove by induction that the above method correctly returns the square of `n`. Remember to write all parts of an inductive proof. (You can use the back of the paper if you run out of room)

   Hint: you may want to use the fact that $(k+1)^2 = k^2 + (2k+1)$

   Answer below:

We'll prove by induction that after the loop iterates n times, `ret` stores the value n^2. Since we then return `ret`, the correct value is returned.

Base case: after the loop iterates 0 times, `ret` stores 0, which is 0^2.

Inductive Hypothesis: For some $k \geq 0$, after the loop iterates k times, `ret` stores the value k^2.

Inductive Step: By the inductive hypothesis, after the loop iterates k times, `ret` stores the value k^2. Let's say it iterates one more time.

Since the loop has iterated k times, `odd` stores the value $1 + 2k$ at the beginning of the $k+1$st loop iteration. In the $k+1$st iteration, the only change to `ret` is `ret += odd`. Since `ret` is k^2 (by the inductive hypothesis) and `odd` is $1 + 2k$ at the beginning of the loop, after this line is run `ret` stores $k^2 + 2k + 1 = (k+1)^2$. Therefore, after the $k+1$st iteration, `ret` stores the value $(k+1)^2$, and the inductive hypothesis is proven for $k+1$.