

Logik für Informatiker

Logic for computer scientists

Induction and Program Correctness

Till Mossakowski

WiSe 2007/08



A recursive program computing

$$0 + 1 + 2 + \dots + n$$

```
public natural sumToRec(natural n) {  
    if(n == 0) return 0;  
    else return n + sumToRec(n - 1);  
}
```

An imperative program

```
public natural sumUpTo(natural n) {
  natural sum = 0;
  natural count = 0;
  while(count < n) {
    count += 1;
    sum += count;
  }
  return sum;
}
```

An imperative program

```
public natural sumUpTo(natural n) {  
    natural sum = 0;  
    natural count = 0;  
    while(count < n) {  
        count += 1;  
        sum += count;  
    }  
    return sum;  
}
```

Invariant: $sum = (0 + 1 + 2 + \dots + k) \wedge count = k$

A second imperative implementation

```
public natural sumDownFrom(natural n) {  
    natural sum = 0;  
    natural count = n;  
    while(count > 0) {  
        sum += count;  
        count -= 1;  
    }  
    return sum;  
}
```

A second imperative implementation

```
public natural sumDownFrom(natural n) {  
    natural sum = 0;  
    natural count = n;  
    while(count > 0) {  
        sum += count;  
        count -= 1;  
    }  
    return sum;  
}
```

Invariant: $sum = (count + 1) + \dots + n$

The Java Modeling Language (JML)

`http://www.cs.ucf.edu/~leavens/JML`

Exercises

- Language, proof and logic, 16.27 - 16.31
- write a small Java program annotated with JML, such that universal and existential quantifiers are used