Grundzüge der Informatik 1

Vorlesung 5 - flipped classroom

Korrektheitsbeweise
- Schleifeninvarianten

Schleifeninvariante

- A(n) ist eine Aussage über den Zustand des Algorithmus vor dem n-ten Durchlauf einer Schleife
- Eine Schleifeninvariante ist korrekt, wenn Sie zu Beginn jedes Schleifendurchlaufs erfüllt ist.
- A(1) wird auch als Initialisierung bezeichnet.

Korrektheitsbeweis für Invarianten

- Induktionsanfang: Die Aussage A(1) gilt
- Induktionsschluss: Gilt A(n) und ist die Eintrittsbedingung der Schleife erfüllt so gilt auch A(n+1)

Universität zu Köln

Korrektheitsbeweise
- Schleifeninvarianten

Aufgabe 1

- Schreiben Sie einen Algorithmus, der n! mit Hilfe einer **for**-Schleife berechnet

Korrektheitsbeweise
- Schleifeninvarianten

Aufgabe 2

- Formulieren Sie eine Schleifeninvariante für die **for**-Schleife

Korrektheitsbeweise
- Schleifeninvarianten

Aufgabe 3

- Zeigen Sie die Korrektheit der Schleifeninvariante mit Hilfe von Induktion
  - Was ist der Induktionsanfang?
  - Was ist die Induktionsannahme?
  - Was ist der Induktionsschritt?

Korrektheitsbeweise
- Schleifeninvarianten

```
Fib1(n)
1.  F = new array[1…n]
2.  F[1] = 1
3.  F[2] = 1
4.  for i=3 to n do
5.     F[i] = F[i-1] + F[i-2]
6.  return F[n]
```

Aufgabe 4

- Wir wollen die Korrektheit des Algorithmus zur Berechnung der Fibonacci Zahlen aus der ersten Vorlesung zeigen
- Formulieren Sie dazu zunächst eine geeignete Schleifeninvariante

Korrektheitsbeweise
- Schleifeninvarianten

```
Fib1(n)
F = new array[1…n]
F[1] = 1
F[2] = 1
for i=3 to n do
   F[i] = F[i-1] + F[i-2]
return F[n]
```

Aufgabe 5

- Zeigen Sie die Korrektheit der Invariante mit Hilfe von Induktion

Korrektheitsbeweise
- Schleifeninvarianten

```
Fib1(n)
F = new array[1…n]
F[1] = 1
F[2] = 1
for i=3 to n do
   F[i] = F[i-1] + F[i-2]
return F[n]
```

Aufgabe 6

- Zeigen Sie nun mit Hilfe der Invariante die Korrektheit von Algorithmus Fib1