Vorlesung vom 04.07.06: Programmkonstruktion I

Wo sind wir?

- Hoare-Kalkül
- Jenseits von Hoare
- *Korrekte Programmentwicklung*
  - *Programmkonstruktion*
  - Programmentwicklung durch Transformation, Algorithmendesign

Programmkonstruktion

- Idee: *Programm* immer zusammen mit *Korrektheitsbeweis* entwicklen
- Genau *Spezifikation* nötig
- *Formalisierung* des Problems, *Vor-* und *Nachbedingung*
- Problem: *Gegeben* A, B. *Finde* P, so dass

$$\models \{A\}P\{B\}$$

- *Lösung*: *Rückwärtsbeweis* mit Hoare-Regeln

Drei einfache Prinzipien

1. *Sequentielle* Zerlegung
2. *Fallunterscheidung*
3. *Abschwächung* der Vorbedingung, *Stärkung* der Nachbedingung

- *Problem:* Schleifen und Invarianten

Schleifen und Invarianten

- *Gesucht*: P so dass $P \models \{A\}P\{B\}$

1. *Finde* I, c so dass $I \wedge \neg c \Longrightarrow B$
2. *Zeige* $A \Longrightarrow I$
3. *Finde* b so dass $\{I \wedge c\}b\{I\}$

- *Dann*: $\models \{A\}$`while` b `do` c `od`$\{B\}$

Vier Wege, ein Prädikat abzuschwächen

1. Konjunkt streichen:

$$A \wedge B \wedge C \Longrightarrow A \wedge B$$

2. Konstante durch Variable ersetzen:

$$x < b[1:10] \quad\Longrightarrow\quad x < b[1:i] \wedge 1 \leq i \leq 10$$

3. Wertebereich vergrößern:

$$5 \leq i < 10 \Longrightarrow 0 \leq i < 10$$

4. Disjunkt hinzufügen:

$$A \Longrightarrow A \vee B$$

Beispiel I: Suche

- Generelles Problem: finde i so dass $R_0(i) \Longrightarrow P(i)$
  - Notation: $E(i : R_0 : P)$

1. Abschwächung von R_0 zu R:

$$E(i : R : P) \Longleftrightarrow E(i : R_0 : P), A(i : R_0 : P)$$

   Termination mit $\neg R(i) \vee P(i)$

2. Test vereinfachen: suche Q, R_0', P' so dass

$$E(i : R_0 : P) = Q(V(i : R_0' : P'))$$

- Beispiel: lineare Suche, binäre Suche