Übung “Grundbegriffe der Informatik”

Karlsruher Institut für Technologie

Matthias Schulz, Gebäude 50.34, Raum 247

email: schulz@ira.uka.de

Matthias Janke, Gebäude 50.34, Raum 249

email: matthias.janke@kit.edu

Invarianten

Gegeben: Ein Fass Salz, ein Fass Zucker.

Wiederhole folgenden Prozess:

Einen Löffel Inhalt des Salzfasses ins Zuckerfass.
Einen Löffel Inhalt des Zuckerfasses ins Salzfass.

Ist am Ende mehr Zucker im Salzfass oder mehr Salz im Zuckerfass?

.

Invarianten

Gegeben: Ein Fass Salz, ein Fass Zucker.

Wiederhole folgenden Prozess:

Einen Löffel Inhalt des Salzfasses ins Zuckerfass.
Einen Löffel Inhalt des Zuckerfasses ins Salzfass.

Ist am Ende mehr Zucker im Salzfass oder mehr Salz im Zuckerfass?
Und was hat das mit Invarianten zu tun?

.

Invarianten

Gegeben: Ein Fass Salz, ein Fass Zucker.

Wiederhole folgenden Prozess:

Einen Löffel Inhalt des Salzfasses ins Zuckerfass.
Einen Löffel Inhalt des Zuckerfasses ins Salzfass.

Genau so viel Zucker im Salzfass wie Salz im Zuckerfass.

.

Invarianten

Gegeben: Ein Fass Salz, ein Fass Zucker.

Wiederhole folgenden Prozess:

Einen Löffel Inhalt des Salzfasses ins Zuckerfass.
Einen Löffel Inhalt des Zuckerfasses ins Salzfass.

Invariant: (Volumen-)Menge in Salzfass.

.

Invarianten

Gegeben: Ein Fass Salz, ein Fass Zucker.

Wiederhole folgenden Prozess:

Einen Löffel Inhalt des Salzfasses ins Zuckerfass.
Einen Löffel Inhalt des Zuckerfasses ins Salzfass.

Invariant: (Volumen-)Menge in Salzfass.
Invariant: Menge an Salz.

.

Invarianten

Gegeben: Ein Fass Salz, ein Fass Zucker.

Wiederhole folgenden Prozess:

Einen Löffel Inhalt des Salzfasses ins Zuckerfass.
Einen Löffel Inhalt des Zuckerfasses ins Salzfass.

Anfangs gilt: Genau so viel Zucker in Salzfass wie Salz in Zuckerfass.

.

Invarianten

Gegeben: Ein Fass Salz, ein Fass Zucker.

Wiederhole folgenden Prozess:

Einen Löffel Inhalt des Salzfasses ins Zuckerfass.
Einen Löffel Inhalt des Zuckerfasses ins Salzfass.

Zucker in Zuckerfass: z_Z, Zucker in Salzfass: z_S
Salz in Zuckerfass: s_Z, Salz in Salzfass: s_S

.

Invarianten

Gegeben: Ein Fass Salz, ein Fass Zucker.

Wiederhole folgenden Prozess:

Einen Löffel Inhalt des Salzfasses ins Zuckerfass.
Einen Löffel Inhalt des Zuckerfasses ins Salzfass.

Zucker in Zuckerfass: z_z, Zucker in Salzfass: z_s
Salz in Zuckerfass: s_z, Salz in Salzfass: s_s

Es gilt immer: $s_z + s_s = z_s + s_s$

.

Invarianten

Gegeben: Ein Fass Salz, ein Fass Zucker.

Wiederhole folgenden Prozess:

Einen Löffel Inhalt des Salzfasses ins Zuckerfass.
Einen Löffel Inhalt des Zuckerfasses ins Salzfass.

Zucker in Zuckerfass: z_z, Zucker in Salzfass: z_s
Salz in Zuckerfass: s_z, Salz in Salzfass: s_s

Es gilt immer: $s_z + s_s = z_s + s_s$

Also $s_z = z_s$

.

Invarianten

Schale mit a blauen, b roten und c grünen Kugeln.

Wiederhole: Nimm zwei verschiedenfarbige Kugeln und ersetze sie durch eine Kugel der dritten Farbe.

.

Invarianten

Schale mit a blauen, b roten und c grünen Kugeln.

Wiederhole: Nimm zwei verschiedenfarbige Kugeln und ersetze sie durch eine Kugel der dritten Farbe.

$(3,2,4) \rightarrow (2,1,5) \rightarrow (1,2,4) \rightarrow (2,1,3) \rightarrow (1,2,2) \rightarrow (2,1,1) \rightarrow (1,0,2) \rightarrow (0,1,1) \rightarrow (1,0,0)$

.

Invarianten

Schale mit a blauen, b roten und c grünen Kugeln.

Wiederhole: Nimm zwei verschiedenfarbige Kugeln und ersetze sie durch eine Kugel der dritten Farbe.

Bekannt: Am Ende bleibe eine einzelne Kugel übrig. Welche Farbe?

.

Invarianten

Schale mit a blauen, b roten und c grünen Kugeln.

Wiederhole: Nimm zwei verschiedenfarbige Kugeln und ersetze sie durch eine Kugel der dritten Farbe.

Bekannt: Am Ende bleibe eine einzelne Kugel übrig. Welche Farbe?

Suche Invariante ...

.

Invarianten

Schale mit a blauen, b roten und c grünen Kugeln.

Wiederhole: Nimm zwei verschiedenfarbige Kugeln und ersetze sie durch eine Kugel der dritten Farbe.

Bekannt: Am Ende bleibe eine einzelne Kugel übrig. Welche Farbe?

$(a, b, c) \to (a - 1, b - 1, c + 1)$

.

Invarianten

Schale mit a blauen, b roten und c grünen Kugeln.

Wiederhole: Nimm zwei verschiedenfarbige Kugeln und ersetze sie durch eine Kugel der dritten Farbe.

Bekannt: Am Ende bleibe eine einzelne Kugel übrig. Welche Farbe?

$(a, b, c) \to (a - 1, b - 1, c + 1)$

Alles ändert sich!

.

Invarianten

$$(a, b, c) \to (a-1, b-1, c+1)$$

Wie siehts mit Differenzen aus?

$$(a-1)-(b-1) = a-b$$
$$(a-1)-(c+1) = a-c-2$$
$$(b-1)-(c+1) = b-c-2$$

.

Invarianten

Schale mit a blauen, b roten und c grünen Kugeln.

Wiederhole: Nimm zwei verschiedenfarbige Kugeln und ersetze sie durch eine Kugel der dritten Farbe.

Bekannt: Am Ende bleibe eine einzelne Kugel übrig. Welche Farbe?

$(a, b, c) \to (a - 1, b - 1, c + 1)$

Differenzen modulo 2 bleiben gleich.

.

Invarianten

Schale mit a blauen, b roten und c grünen Kugeln.

Wiederhole: Nimm zwei verschiedenfarbige Kugeln und ersetze sie durch eine Kugel der dritten Farbe.

Bekannt: Am Ende bleibe eine einzelne Kugel übrig. Welche Farbe?

Ende: $(0, 1, 0)$:

Für nicht vorhandene Kugelfarben gilt $(x - y) \bmod 2 = 0$
Für vorhandene Kugelfarbe gilt $(y - x) \bmod 2 = 1$

.

Invarianten

Schale mit a blauen, b roten und c grünen Kugeln.

Wiederhole: Nimm zwei verschiedenfarbige Kugeln und ersetze sie durch eine Kugel der dritten Farbe.

Bekannt: Am Ende bleibe eine einzelne Kugel übrig. Welche Farbe?

Also: Farbe mit Anzahl modulo 2 ungleich Anzahl andere Farben modulo 2 bleibt übrig.

.

Aussagenlogik per Arithmetik

0 falsch, 1 wahr:

Term mit A, B, dessen Ergebnis stets der Wahrheitswert von $A \wedge B$ ist.

.

Aussagenlogik per Arithmetik

0 falsch, 1 wahr:

Term mit A, B, dessen Ergebnis stets der Wahrheitswert von $A \wedge B$ ist.

$A = 0 \Rightarrow$ Ergebnis ist 0
$B = 0 \Rightarrow$ Ergebnis ist 0

.

Aussagenlogik per Arithmetik

0 falsch, 1 wahr:

Term mit A, B, dessen Ergebnis stets der Wahrheitswert von $A \wedge B$ ist.

$A = 0 \Rightarrow$ Ergebnis ist 0
$B = 0 \Rightarrow$ Ergebnis ist 0

Also: $A \cdot B$

.

Aussagenlogik per Arithmetik

0 falsch, 1 wahr:

Term mit A, dessen Ergebnis stets der Wahrheitswert von $\neg A$ ist.

$A = 0 \Rightarrow$ Ergebnis ist 1
$A = 1 \Rightarrow$ Ergebnis ist 0

.

Aussagenlogik per Arithmetik

0 falsch, 1 wahr:

Term mit A, dessen Ergebnis stets der Wahrheitswert von $\neg A$ ist.

$A = 0 \Rightarrow$ Ergebnis ist 1
$A = 1 \Rightarrow$ Ergebnis ist 0

Summe (Ergebnis + A) ist immer 1.

.

Aussagenlogik per Arithmetik

0 falsch, 1 wahr:

Term mit A, dessen Ergebnis stets der Wahrheitswert von $\neg A$ ist.

$A = 0 \Rightarrow$ Ergebnis ist 1
$A = 1 \Rightarrow$ Ergebnis ist 0

Also: $1 - A$

.

Aussagenlogik per Arithmetik

0 falsch, 1 wahr:

Weitere Formeln: Auf Übungsblatt

.

Algorithmen

Gegeben Funktion $f : A \times A \to \mathbb{G}_2$,
$\forall x, y \in A : f(x, y) = 1 \iff x = y$.

Programm mit Eingabe $w \in A^*$:

$k \leftarrow 1$;
for $i = 0$ to $|w| - 1$ do
  $k \leftarrow k * f(w(i), w(|w| - 1 - i))$;
od

.

Algorithmen

Gegeben Funktion $f : A \times A \to \mathbb{G}_2$,
$\forall x, y \in A : f(x, y) = 1 \iff x = y$.

Programm mit Eingabe $w \in A^*$:

$k \leftarrow 1$;
for $i = 0$ to $|w| - 1$ do
  $k \leftarrow k * f(w(i), w(|w| - 1 - i))$;
od

Wann gilt am Ende $k = 1$?

.

Algorithmen

Gegeben Funktion $f : A \times A \to \mathbb{G}_2$,
$\forall x, y \in A : f(x, y) = 1 \iff x = y$.

Programm mit Eingabe $w \in A^*$:

$k \leftarrow 1$;
for $i = 0$ to $|w| - 1$ do
  $k \leftarrow k * f(w(i), w(|w| - 1 - i))$;
od

$k = 0 \Rightarrow \exists i \in \mathbb{G}_{|w|} : f(w(i), w(|w| - 1 - i)) = 0$

.

Algorithmen

Gegeben Funktion $f : A \times A \to \mathbb{G}_2$,
$\forall x, y \in A : f(x, y) = 1 \iff x = y$.

Programm mit Eingabe $w \in A^*$:

$k \leftarrow 1$;
for $i = 0$ to $|w| - 1$ do
  $k \leftarrow k * f(w(i), w(|w| - 1 - i))$;
od

Also $k = 1 \Rightarrow \forall i \in \mathbb{G}_{|w|} : f(w(i), w(|w| - 1 - i)) = 1$

.

Beispiele: ANNA, OTTO, RELIEFPFEILER ...

Palindrome!

.

Algorithmen

Gegeben Funktion $f : A \times A \to \mathbb{G}_2$,
$\forall x, y \in A : f(x,y) = 1 \iff x = y$.

Programm mit Eingabe $w \in A^*$:

$k \leftarrow 1$;
for $i = 0$ to $|w| - 1$ do
  $k \leftarrow k * f(w(i), w(|w| - 1 - i))$;
od

Schleifeninvariante: $k = 1 \Rightarrow \forall j < i : w(j) = w(|w| - 1 - j)$.

.

Algorithmen

Gegeben Funktion $f : A \times A \to \mathbb{G}_2$,
$\forall x, y \in A : f(x, y) = 1 \iff x = y$.

Programm mit Eingabe $w \in A^*$:

$k \leftarrow 1$;
for $i = 0$ to $|w| - 1$ do
$\langle$Schleifeninvariante gilt$\rangle$
    $k \leftarrow k * f(w(i), w(|w| - 1 - i))$;
$\langle$Schleifeninvariante gilt$\rangle$
od

Schleifeninvariante: $k = 1 \Rightarrow \forall j < i : w(j) = w(|w| - 1 - j)$.

.

Algorithmen

Gegeben Funktion $f : A \times A \to \mathbb{G}_2$,
$\forall x, y \in A : f(x, y) = 1 \iff x = y$.

Programm mit Eingabe $w \in A^*$:

$k \leftarrow 1$;
for $i = 0$ to $|w| - 1$ do
$\langle$Schleifeninvariante gilt für $i\rangle$
  $k \leftarrow k * f(w(i), w(|w| - 1 - i))$;
$\langle$Schleifeninvariante gilt für $i + 1\rangle$
od

Schleifeninvariante: $k = 1 \Rightarrow \forall j < i : w(j) = w(|w| - 1 - j)$.

.

Gegeben Funktion $f : A \times A \to \mathbb{G}_2$,
$\forall x, y \in A : f(x, y) = 1 \iff x = y$.

Programm mit Eingabe $w \in A^*$:

$k \leftarrow 1$;
for $i = 0$ to $|w| - 1$ do
$\langle$Schleifeninvariante gilt für $i\rangle$
  $k \leftarrow k * f(w(i), w(|w| - 1 - i))$;
$\langle$Schleifeninvariante gilt für $i + 1\rangle$
od

Problem: Schleifeninvariante soll vor Durchlauf gilt und nach Durchlauf gelten.

$k \leftarrow 1$;
$j \leftarrow -1$;
for $i = 0$ to $|w| - 1$ do
$\langle$Schleifeninvariante gilt$\rangle$
    $j \leftarrow i$;
    $k \leftarrow k * f(w(j), w(|w| - 1 - j))$;
$\langle$Schleifeninvariante gilt$\rangle$
od

SI: $k = 1 \Rightarrow \forall m \leq j : w(m) = w(|w| - 1 - m)$.

Lösung: Kopieren des Zählers in neue Variable, die sich in Rumpf ändert.

Algorithmen

Definiere max(a, b) als größere der beiden Zahlen a, b, min(a, b) als kleinere der beiden Zahlen a, b.

$n \leftarrow a$;
$m \leftarrow b$;
for $i = 0$ to $\lceil 2 * ld(\max(a, b)) \rceil$ do
  $k \leftarrow \min(n, m)$;
  $n \leftarrow \max(n, m) - \min(n, m)$;
  $m \leftarrow k$;
od

.

Algorithmen

Definiere max(a, b) als größere der beiden Zahlen a, b, min(a, b) als kleinere der beiden Zahlen a, b.

$n \leftarrow a$;
$m \leftarrow b$;
for $i = 0$ to $\lceil 2 * ld(\max(a, b)) \rceil$ do
  $k \leftarrow \min(n, m)$;
  $n \leftarrow \max(n, m) - \min(n, m)$;
  $m \leftarrow k$;
od

Beispiel: $(14, 8) \rightarrow (6, 8) \rightarrow (2, 6) \rightarrow (4, 2) \rightarrow (2, 2) \rightarrow (0, 2) \rightarrow (2, 0)$

.

Algorithmen

Definiere $\max(a, b)$ als größere der beiden Zahlen a, b, $\min(a, b)$ als kleinere der beiden Zahlen a, b.

$n \leftarrow a$;
$m \leftarrow b$;
for $i = 0$ to $\lceil 2 * ld(\max(a, b))\rceil$ do
  $k \leftarrow \min(n, m)$;
  $n \leftarrow \max(n, m) - \min(n, m)$;
  $m \leftarrow k$;
od

Beispiel: $(15, 9) \rightarrow (6, 9) \rightarrow (3, 6) \rightarrow (3, 3) \rightarrow (0, 3) \rightarrow (3, 0)$

.

Definiere max(a, b) als größere der beiden Zahlen a, b, min(a, b) als kleinere der beiden Zahlen a, b.

$n \leftarrow a$;
$m \leftarrow b$;
for $i = 0$ to $\lceil 2 * ld(\max(a, b)) \rceil$ do
   $k \leftarrow \min(n, m)$;
   $n \leftarrow \max(n, m) - \min(n, m)$;
   $m \leftarrow k$;
od

Vermutung: Am Ende ist $m = ggt(a, b)$

.

Definiere max(a, b) als größere der beiden Zahlen a, b, min(a, b) als kleinere der beiden Zahlen a, b.

$n \leftarrow a$;
$m \leftarrow b$;
for $i = 0$ to $\lceil 2 * ld(\max(a, b)) \rceil$ do
  $k \leftarrow \min(n, m)$;
  $n \leftarrow \max(n, m) - \min(n, m)$;
  $m \leftarrow k$;
od

Vermutung: Es gilt immer: $ggt(a, b) = ggt(n, m)$.

.

Algorithmen

Vermutung: Es gilt immer: $ggt(a, b) = ggt(n, m)$.

Anfang: $ggt(n, m) = ggt(a, b)$ nach Initialisierung.

.

Algorithmen

Vermutung: Es gilt immer: $ggt(a, b) = ggt(n, m)$.

Invariante: $ggt(n, m) = ggt(a, b)$ gilt am Anfang der Schleife.

Am Ende der Schleife: $ggt(\mathsf{max}(n, m) - \mathsf{min}(n, m), \mathsf{min}(n, m))$

.

Vermutung: Es gilt immer: $ggt(a, b) = ggt(n, m)$.

Invariante: $ggt(n, m) = ggt(a, b)$ gilt am Anfang der Schleife.

Am Ende der Schleife: $ggt(\max(n, m) - \min(n, m), \min(n, m))$

$ggt(n, m)$ teilt $\max(n, m), \min(n, m)$ und damit
$\max(n, m) - \min(n, m)$
$\Rightarrow ggt(\max(n, m) - \min(n, m), \min(n, m)) \geq ggt(n, m)$

.

Vermutung: Es gilt immer: $ggt(a, b) = ggt(n, m)$.

Invariante: $ggt(n, m) = ggt(a, b)$ gilt am Anfang der Schleife.

Am Ende der Schleife: $ggt(\text{max}(n, m) - \text{min}(n, m), \text{min}(n, m))$

$ggt(\text{max}(n, m) - \text{min}(n, m), \text{min}(n, m))$ teilt $\text{min}(n, m)$, $\text{max}(n, m) - \text{min}(n, m)$ und damit auch $\text{max}(n, m)$
$\Rightarrow ggt(\text{max}(n, m) - \text{min}(n, m), \text{min}(n, m))$ teilt n und $m \Rightarrow$
$ggt(\text{max}(n, m) - \text{min}(n, m), \text{min}(n, m)) \leq ggt(n, m)$

.

Vermutung: Es gilt immer: $ggt(a, b) = ggt(n, m)$.

Invariante: $ggt(n, m) = ggt(a, b)$ gilt am Anfang der Schleife.

Am Ende der Schleife: $ggt(\text{max}(n, m) - \text{min}(n, m), \text{min}(n, m))$

$ggt(\text{max}(n, m) - \text{min}(n, m), \text{min}(n, m))$ teilt $\text{min}(n, m)$, $\text{max}(n, m) - \text{min}(n, m)$ und damit auch $\text{max}(n, m)$
$\Rightarrow ggt(\text{max}(n, m) - \text{min}(n, m), \text{min}(n, m))$ teilt n und m $\Rightarrow$
$ggt(\text{max}(n, m) - \text{min}(n, m), \text{min}(n, m)) \leq ggt(n, m)$

$\Rightarrow ggt(n, m) = ggt(\text{max}(n, m) - \text{min}(n, m), \text{min}(n, m))$

.

Für "saubere" Induktion besser geeignet:

$n_0 \leftarrow a$;
$m_0 \leftarrow b$;
for $i = 0$ to $\lceil 2 * ld(\max(a, b)) \rceil$ do
  $n_{i+1} \leftarrow \max(n_i, m_i) - \min(n_i, m_i)$;
  $m_{i+1} \leftarrow \min(n_i, m_i)$;
od

Mehr Speicherplatz!

.

Für “saubere” Induktion besser geeignet:

$n_0 \leftarrow a$;
$m_0 \leftarrow b$;
for $i = 0$ to $\lceil 2 * ld(\max(a,b)) \rceil$ do
  $n_{i+1} \leftarrow \max(n_i, m_i) - \min(n_i, m_i)$;
  $m_{i+1} \leftarrow \min(n_i, m_i)$;
od

Wieso keine Kopie des Zählers?

.

Für “saubere” Induktion besser geeignet:

$n_0 \leftarrow a$;
$m_0 \leftarrow b$;
for $i = 0$ to $\lceil 2 * ld(\max(a, b)) \rceil$ do
  $n_{i+1} \leftarrow \max(n_i, m_i) - \min(n_i, m_i)$;
  $m_{i+1} \leftarrow \min(n_i, m_i)$;
od

Wieso keine Kopie des Zählers? $\rightarrow$ In Schleifeninvariante kommt Zähler nicht direkt vor, nur als Index.

.