

Übungen zur Vorlesung  
**Modellierung**  
WS 2003/2004  
Blatt 7 Musterlösungen

**KORREKTURAUFGABE 44 :**

Bilden Sie zu den folgenden prädikatenlogischen Formeln eine zugehörige pränex Normalform.

- a)  $\alpha = \exists x(P(x) \vee \forall yZ(y, x)) \wedge \exists x(H(x) \vee \neg\exists zF(z, y, x))$   
 b)  $\beta = \exists x(A(x) \wedge (\forall xB(x) \vee \forall yC(x))) \rightarrow (\exists yD(y) \wedge \neg\forall yE(y))$   
 c)  $\gamma = \exists w(\exists y(P(x) \rightarrow \forall v(\neg\exists x(H(x, y) \vee \exists z(F(z, y, x)))))) \rightarrow D(v)$   
 d)  $\delta = \exists x(\neg A(x) \rightarrow (\forall xB(x) \wedge \forall xC(x))) \wedge \neg(\forall y\neg D(x) \vee \neg E(y))$

**Lösung:**

Es gibt verschiedene Wege, eine pränex Normalform herzustellen. Erfahrungsgemäß passieren die wenigsten Fehler, wenn zunächst eine Negationsnormalform hergestellt wird, d.h. Implikations- und Äquivalenzzeichen durch die entsprechenden Formeln mit Junktoren  $\wedge, \vee, \neg$  ersetzt werden und die Negationen dann vor die Prädikate gezogen werden. Anschließend bringt man die Quantoren nach vorn (gegebenenfalls gleiche quantifizierte Variable umbenennen) und transformiert den Kern in KNF.

Pränexe Normalform von  $\alpha$ :

$$\begin{aligned} & \exists x(P(x) \vee \forall yZ(y, x)) \wedge \exists x(H(x) \vee \neg\exists zF(z, y, x)) \\ \approx & \exists x(P(x) \vee \forall yZ(y, x)) \wedge \exists x(H(x) \vee \forall z\neg F(z, y, x)) \\ \approx & \exists v(P(v) \vee \forall wZ(w, v)) \wedge \exists x(H(x) \vee \forall z\neg F(z, y, x)) \\ \approx & \exists v\forall w\exists x\forall z((P(v) \vee Z(w, v)) \wedge (H(x) \vee \neg F(z, y, x))) \end{aligned}$$

Pränexe Normalform von  $\beta$ :

$$\begin{aligned} & \exists x(A(x) \wedge (\forall xB(x) \vee \forall yC(x))) \rightarrow (\exists yD(y) \wedge \neg\forall yE(y)) \\ \approx & \exists x(A(x) \wedge (\forall uB(u) \vee \forall yC(x))) \rightarrow (\exists vD(v) \wedge \neg\forall wE(w)) \\ \approx & \neg(\exists x(A(x) \wedge (\forall uB(u) \vee \forall yC(x)))) \vee (\exists vD(v) \wedge \neg\forall wE(w)) \\ \approx & \forall x\neg(A(x) \wedge (\forall uB(u) \vee \forall yC(x))) \vee (\exists vD(v) \wedge \exists w\neg E(w)) \\ \approx & \forall x(\neg A(x) \vee (\exists u\neg B(u) \wedge \exists y\neg C(x))) \vee (\exists vD(v) \wedge \exists w\neg E(w)) \\ \approx & \forall x(\neg A(x) \vee (\exists u\neg B(u) \wedge \neg C(x))) \vee (\exists vD(v) \wedge \exists w\neg E(w)) \\ \approx & \forall x\exists u\exists v\exists w((\neg A(x) \vee (\neg B(u) \wedge \neg C(x))) \vee (D(v) \wedge \neg E(w))) \\ \approx & \forall x\exists u\exists v\exists w(((\neg A(x) \vee \neg B(u)) \wedge (\neg A(x) \vee \neg C(x)) \vee (D(u) \wedge \neg E(w)))) \\ \approx & \forall x\exists u\exists v\exists w((\neg A(x) \vee \neg B(u) \vee D(u)) \wedge (\neg A(x) \vee \neg C(x) \vee D(u)) \wedge \\ & (\neg A(x) \vee \neg B(u) \vee \neg E(w)) \wedge (\neg A(x) \vee \neg C(x) \vee \neg E(w))) \end{aligned}$$

Pränexe Normalform von  $\gamma$ :

$$\begin{aligned}
 & \exists w(\exists y(P(x) \rightarrow \forall v(\neg\exists x(H(x, y) \vee \exists zF(z, y, x)))) \rightarrow D(v)) \\
 \approx & \exists w(\neg\exists y(\neg P(x) \vee \forall v(\neg\exists x(H(x, y) \vee \exists zF(z, y, x)))) \vee D(v)) \\
 \approx & \exists w(\forall y\neg(\neg P(x) \vee \forall v(\neg\exists x(H(x, y) \vee \exists zF(z, y, x)))) \vee D(v)) \\
 \approx & \forall w(\forall y(P(x) \wedge \exists v\neg(\neg\exists x(H(x, y) \vee \exists zF(z, y, x)))) \vee D(v)) \\
 \approx & \forall w(\forall y(P(x) \wedge \exists v(\exists x(H(x, y) \wedge \forall z\neg F(z, y, x)))) \vee D(v)) \\
 \approx & \forall w(\forall y(P(x) \wedge (\exists u(H(u, y) \wedge \forall z\neg F(z, y, u)))) \vee D(v)) \\
 \approx & \forall w\forall y\exists u\forall z((P(x) \wedge (H(u, y) \wedge \neg F(z, y, x))) \vee D(v)) \\
 \approx & \forall w\forall y\exists u\forall z((P(x) \vee D(v)) \wedge (H(u, y) \vee D(v)) \wedge (\neg F(z, y, x) \vee D(v)))
 \end{aligned}$$

Pränexe Normalform von  $\delta$ :

$$\begin{aligned}
 & \exists x(\neg A(x) \rightarrow (\forall xB(x) \wedge \forall xC(x))) \wedge \neg(\forall y\neg D(x) \vee \neg E(y)) \\
 \approx & \exists x(A(x) \vee (\forall zB(z) \wedge \forall vC(v))) \wedge (D(x) \wedge E(y)) \\
 \approx & \exists u(A(u) \vee (\forall zB(z) \wedge \forall vC(v))) \wedge (D(x) \wedge E(y)) \\
 \approx & \exists u\forall z\forall v((A(u) \vee (B(z) \wedge C(v))) \wedge (D(x) \wedge E(y))) \\
 \approx & \exists u\forall z\forall v((A(u) \vee B(z)) \wedge (A(u) \vee C(v)) \wedge D(x) \wedge E(y))
 \end{aligned}$$

### AUFGABE 45 :

Zeigen Sie, dass folgende Formelpaare jeweils nicht logisch äquivalent sind. Wählen Sie für  $P$  und  $Q$  möglichst einfache Interpretationen und möglichst einfache Grundbereiche.

- $\forall x\exists yP(x, y)$  und  $\exists y\forall xP(x, y)$
- $\forall x(P(x) \vee \neg Q(x))$  und  $(\forall xP(x)) \vee (\forall x\neg Q(x))$
- $\exists xP(x) \leftrightarrow \exists xQ(x)$  und  $\exists x(P(x) \leftrightarrow Q(x))$

### Lösung:

- Gegenbeispiel für logische Äquivalenz:

$$\begin{aligned}
 \text{Grundbereich } U &= \{1, 2\} \\
 \mathfrak{S}(P) &= \{(1, 2), (2, 1)\}
 \end{aligned}$$

Es gilt dann  $\mathfrak{S}(\forall x\exists yP(x, y)) = 1$ , aber  $\mathfrak{S}(\exists x\forall yP(x, y)) = 0$ .

- Gegenbeispiel für logische Äquivalenz:

$$\begin{aligned}
 \text{Grundbereich } U &= \{1, 2\} \\
 \mathfrak{S}(P) &= \{1\} \\
 \mathfrak{S}(Q) &= \{1\}
 \end{aligned}$$

Es gilt dann  $\mathfrak{S}(\forall x(P(x) \vee \neg Q(x))) = 1$ , aber  $\mathfrak{S}(\forall x(P(x)) \vee (\forall x\neg Q(x))) = 0$ .

c) Gegenbeispiel für logische Äquivalenz:

Grundbereich  $U = \{1, 2\}$

$\mathfrak{S}(P) = \{1\}$

$\mathfrak{S}(Q) = \{2\}$

Es gilt dann  $\mathfrak{S}(\exists x P(x) \leftrightarrow \exists x Q(x)) = 1$ , aber  $\mathfrak{S}(\exists x (P(x) \leftrightarrow Q(x))) = 0$ .

#### AUFGABE 46 :

Überprüfen Sie die Korrektheit der folgenden Äquivalenzen mit Hilfe der Quantorenregeln. Benennen Sie jeweils ALLE nötigen Regeln.

- a)  $\forall x P(x) \wedge \exists x Q(x) \approx \forall x \exists x_1 (P(x) \wedge Q(x_1))$
- b)  $\exists x Q(x) \wedge \forall y P(z) \wedge \forall y P(y) \approx \exists x \forall y (Q(x) \wedge P(z) \wedge P(y))$
- c)  $\exists x \forall z (\neg P(x) \vee \exists y Q(y)) \approx \exists x \neg \forall y (P(x) \wedge \neg Q(y))$
- d)  $\forall x (P(x) \wedge \neg (\exists y Q(y) \wedge P(y))) \approx \forall x \forall z (P(x) \wedge \neg (Q(z) \wedge P(y)))$

#### Lösung:

- a)  $\forall x P(x) \wedge \exists x Q(x)$   
 $\approx \forall x P(x) \wedge \exists x_1 Q(x_1)$  (Umbenennung)  
 $\approx \forall x \exists x_1 (P(x) \wedge Q(x_1))$  (Quantifizierung)
- b)  $\exists x Q(x) \wedge \forall y P(z) \wedge \forall y P(y)$   
 $\approx \exists x Q(x) \wedge P(z) \wedge \forall y P(y)$  (Quantorelimination)  
 $\approx \exists x \forall y (Q(x) \wedge P(z) \wedge P(y))$  (Quantifizierung)
- c)  $\exists x \forall z (\neg P(x) \vee \exists y Q(y))$   
 $\approx \exists x (\neg P(x) \vee \exists y Q(y))$  (Quantorelimination)  
 $\approx \exists x \exists y (\neg P(x) \vee Q(y))$  (Quantifizierung)  
 $\approx \exists x \neg \forall y (P(x) \wedge \neg Q(y))$  (Quantorwechsel)
- d)  $\forall x (P(x) \wedge \neg (\exists y Q(y) \wedge P(y)))$   
 $\approx \forall x (P(x) \wedge (\forall y \neg Q(y) \vee \neg P(y)))$  (Quantorwechsel)  
 $\approx \forall x (P(x) \wedge (\forall z \neg Q(z) \vee \neg P(y)))$  (Umbenennung)  
 $\approx \forall x \forall z (P(x) \wedge (\neg Q(z) \vee \neg P(y)))$  (Quantifizierung)  
 $\approx \forall x \forall z (P(x) \wedge \neg (Q(z) \wedge P(y)))$  (De Morgan)

**AUFGABE 47 :**

Entwerfen Sie mit wenigen Befehlen einen Algorithmus, der bei Eingabe von zwei ganzen Zahlen  $a$  und  $b$  den Rest der ganzzahligen Division berechnet. Benutzen Sie hierzu die in der Vorlesung vorgestellte einfache Programmiersprache.

Vorbedingung :  $\{a, b \in \mathbb{N} \text{ und } a \geq 0 \text{ und } b > 0\}$

Nachbedingung :  $\{c = a \bmod b\}$

**Lösung:**

Als vollständigen Algorithmus erhalten wir:

```

{a, b ∈ ℕ und a ≥ 0 und b > 0}
begin
  c := a;
  while (c ≥ b) do
    c := c - b;
  end
{c = a mod b}

```

**AUFGABE 48 :**

Wenden Sie bei den folgenden Verifikationen die Zuweisungsregel und die Abschwächungsregeln an und bestimmen Sie so die Nach- bzw. Vorbedingungen.

- a)  $\{2 < b\}$   
 $a := b;$   
 $\{\dots\}$
- b)  $\{a = c + 2\} \rightarrow \{\dots\}$   
 $b := a - 2;$   
 $\{\dots\}$
- c)  $\{\dots\}$   
 $a := a - 2;$   
 $\{2 > a - 2\}$
- d)  $\{a < b\} \rightarrow \{\dots\}$   
 $c := b - 2;$   
 $\{\dots\}$
- e)  $\{a = b^2\} \rightarrow \{\dots\}$   
 $a := a + 2b + 1;$   
 $\{a = b^2 + 2b + 1\}$   
 $b := b + 1;$   
 $\{\dots\}$

- h)  $\{\dots\}$   
 $a := a - b;$   
 $\{a \geq 2\}$
- g)  $\left\{a = \sum_{i=1}^b i\right\} \rightarrow \{\dots\}$   
 $a := a + b + 1;$   
 $\{\dots\}$

**Lösung:**

Lösung in blau ergänzt.

- a)  $\{2 < b\} \rightarrow \{2 < b \text{ und } b = b\}$   
 $a := b;$   
 $\{2 < b \text{ und } a = b\}$
- b)  $\{a = c + 2\} \rightarrow \{a = c + 2 \text{ und } a - 2 = a - 2\}$   
 $b := a - 2;$   
 $\{a = c + 2 \text{ und } b = a - 2\}$
- c)  $\{2 > a - 4\}$   
 $a := a - 2;$   
 $\{2 > a - 2\}$
- d)  $\{a < b\} \rightarrow \{a < b \text{ und } b - 2 = b - 2\}$   
 $c := b - 2;$   
 $\{a < c \text{ und } c = b - 2\}$
- e)  $\{a = b^2\} \rightarrow \{a + 2b + 1 = b^2 + 2b + 1\}$   
 $a := a + 2b + 1;$   
 $\{a = b^2 + 2b + 1\} \rightarrow \{a = (b + 1)^2\}$   
 $b := b + 1;$   
 $\{a = b^2\}$
- f)  $\{a - b \geq 2\}$   
 $a := a - b;$   
 $\{a \geq 2\}$
- g)  $\left\{a = \sum_{i=1}^b i\right\} \rightarrow \left\{a + b + 1 = \sum_{i=1}^{b+1} i\right\}$   
 $a := a + b + 1;$   
 $\left\{a = \sum_{i=1}^{b+1} i\right\}$