Fachhochschule Wedel
Prof. Dr. Uwe Schmidt

26. Januar 1996

Aufgaben zur Klausur **Grundlagen der Programmierung**, **Software Engineering** und **Logische Programmierung** im WS 95/96 (WI03)

Zeit: 120 Minuten

erlaubte Hilfsmittel: keine

Bitte tragen Sie Ihre Antworten und fertigen Lösungen ausschließlich an den gekennzeichneten Stellen in das Aufgabenblatt ein. Ist ihre Lösung wesentlich umfangreicher, so überprüfen Sie bitte nochmals Ihren Lösungsweg.

Viel Erfolg !

Diese Klausur besteht einschließlich dieses Deckblattes aus 10 Seiten

Aufgabe 1:

Beweisen Sie durch Umformung und Zurückführen auf die Operationen $\vee$,$\wedge$ und $\neg$, daß $(x \oplus x) \Leftrightarrow \mathsf{false}$ ein Satz der Aussagenlogik ist.

$$(x \oplus x) \Leftrightarrow \mathsf{false}$$

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

Zeigen Sie ebenfalls durch Umformung, daß **false** ein neutrales Element bezüglich der Operation $\oplus$ ist.

false ist neutrales Element bezüglich $\oplus$

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

Aufgabe 2:

Gegeben sei eine Variable f für ein Feld

var

f : array $[0..n-1]$ of Z

und die folgenden prädikatenlogischen Formeln

1. $\exists\, 0 \leq i, j < n \bullet f[i] < 0 \Leftrightarrow f[j] > 0$
2. $\exists\, 0 \leq i, j < n \bullet f[i] < 0 \wedge f[j] \geq 0$
3. $\exists\, 0 \leq i, j < n \bullet f[i] \geq 0 \oplus f[j] \geq 0$
4. $\exists\, 0 \leq i, j < n \bullet i \neq 0 \wedge f[i] < 0 \wedge f[j] \geq 0$
5. $\exists\, 0 \leq i < n \bullet f[i-1] < 0 \wedge f[i] \geq 0$
6. $\exists\, 0 \leq i, j < n \bullet f[i] < 0 \wedge f[j] \geq 0 \wedge (j = i-1 \vee j = i+1)$
7. $\exists\, 0 \leq i < n \bullet \exists\, i < j < n \bullet f[i] < 0 \Leftrightarrow f[j] \geq 0$
8. $\exists\, 0 \leq i < n \bullet \exists\, i < j < n \bullet f[i] < 0 \wedge f[j] \geq 0$
9. $\forall\, 0 \leq i < n \bullet i \bmod 2 = 0 \Rightarrow f[i] \geq 0$
10. $\forall\, 0 \leq i < n-2 \bullet i \bmod 2 = 0 \Rightarrow (f[i] \geq 0 \Leftrightarrow f[i+2] \geq 0)$
11. $\forall\, 0 \leq i, j < n \bullet (i-j) \bmod 2 = 0 \Rightarrow (f[i] \geq 0 \Leftrightarrow f[j] \geq 0)$
12. $\forall\, 0 \leq i, j < n \bullet (i-j) \bmod 2 = 0 \Leftrightarrow (f[i] \geq 0 \Rightarrow f[j] \geq 0)$
13. $\forall\, 0 \leq i, j < n \bullet (i-j) \bmod 2 = 0 \Leftrightarrow (f[i] \geq 0 \Leftrightarrow f[j] \geq 0)$

Geben sie für die folgenden Aussagen die Nummer(n) von **gleichwertigen** Formeln an, Mehrfachnennungen sind möglich, gibt es keine Formel tragen Sie 0 an die vorgesehene Stelle ein.

1. Nicht alle Elemente haben das gleiche Vorzeichen.

   ....

2. Alle benachbarten Elemente haben das gleiche Vorzeichen.

   ....

3. Nicht alle benachbarten Elemente haben das gleiche Vorzeichen.

   ....

4. Alle Elemente mit geradem Index sind positiv.

   ....

5. Alle Elemente mit ungeradem Index sind negativ.

   ....

6. Alle Elemente mit geradem Index haben gleiches Vorzeichen.

   ....

7. Jedes Element hat das gleiche Vorzeichen wie sein übernächstes.

   ....

8. Jedes Element hat ein anderes Vorzeichen als seine benachbarten Elemente.

   ....

9. false

   ....

10. Welche Formeln können nicht ausgewertet werden, sind also fehlerhaft aufgebaut.

    ....

Aufgabe 3:

Gegeben sei ein Feld f mit reellen Zahlen. Es soll der Mittelwert der in f gespeicherten Werte in der Variablen $mittel$ berechnet werden.

var f : array $[0..n-1]$ of R;
var i : N_0;
var $mittel$: R;

Die Nachbedingung P lautet also:

$$mittel = \frac{1}{n} * \sum_{j=0}^{n-1} f[j]$$

Die Berechnung soll mit einer Schleife erfolgen, und zwar so, daß zu jedem Zeitpunkt der Berechnung in der Variablen $mittel$ der Mittelwert eines Anfangsstücks von f enthalten ist.

Hierzu ist die Nachbedingung durch eine Einführung einer neuen (Lauf-) Variablen i für die obere Grenze in die Form $I \wedge \neg B$ umzuformen. Es ergibt sich also:

$$mittel = \frac{1}{i} * \sum_{j=0}^{i-1} f[j] \wedge i = mittel$$

Hieraus ergibt sich das folgende Programmschema

```
                          { V }
i, mittel := E0, E1;
                          { I }
while B do
                          { I∧B }
    i, mittel := E2, E3
                          { I }
end while
                          { I∧¬B }
```

Wie sieht die Initialisierung von i aus?

E_0 : ....

Wie wird i in der Schleife verändert?

E_2 : ....

Berechnen Sie die Initialisierung von E_1 aus der Invarianten und der Initialisierung von i

$$I[i \leftarrow E_0, mittel \leftarrow E_1]$$

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

E_1 kann nicht immer ausgewertet werden. Welche Vorbedingung V stellt die korrekte Auswertung von E_1 sicher?

V : ....

Berechnen Sie den Ausdruck E_3 für die Veränderung der Variablen *mittel* aus der Invarianten und dem Ausruck E_2

$$I[i \leftarrow E_2, mittel \leftarrow E_3]$$

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

Geben sie einen Ausdruck für die Variante t an, der die Terminierung sicherstellt.

t : ....

Setzen sie alle Einzelteile zu dem vollständigen Progamm zusammen (ohne Kopie der Deklarationen aber mit Zusicherungen für die Vorbedingung, die Invariante und die Nachbedingung)

....

....

....

....

....

....

....

Aufgabe 4:

Gegeben sei die folgende Funktion:

$q(i : \mathsf{N}_0; j : \mathsf{N}_0) : \mathsf{N}_0$
  $\mathsf{if}\ i < j$
  $\mathsf{then}\ i$
  $\mathsf{else}\ (i\ \mathsf{div}\ j) + q(i\ \mathsf{mod}\ j, j)$

Transformieren Sie diese Funktion in ein gleichwertiges Prolog Prädikat. Der div Operator wird durch den Infix Operator // dargestellt, der mod Operator durch den Infix Operator mod

. . . .

. . . .

. . . .

. . . .

. . . .

. . . .

. . . .

. . . .

. . . .

. . . .

Fachhochschule Wedel
Prof. Dr. Uwe Schmidt

26. Januar 1996

Aufgaben zur Klausur **Grundlagen der Programmierung** und **Software Engineering** im WS 95/96 (II13)

Zeit: 120 Minuten

erlaubte Hilfsmittel: keine

Bitte tragen Sie Ihre Antworten und fertigen Lösungen ausschließlich an den gekennzeichneten Stellen in das Aufgabenblatt ein. Ist ihre Lösung wesentlich umfangreicher, so überprüfen Sie bitte nochmals Ihren Lösungsweg.

Viel Erfolg !

Diese Klausur besteht einschließlich dieses Deckblattes aus 10 Seiten

---

Aufgabe 1:

Beweisen Sie durch Umformung und Zurückführen auf die Operationen $\vee$,$\wedge$ und $\neg$, daß $(x \oplus x) \Leftrightarrow \mathsf{false}$ ein Satz der Aussagenlogik ist.

$$(x \oplus x) \Leftrightarrow \mathsf{false}$$

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

Zeigen Sie ebenfalls durch Umformung, daß false ein neutrales Element bezüglich der Operation $\oplus$ ist.

false ist neutrales Element bezüglich $\oplus$

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

Aufgabe 2:

Gegeben sei eine Variable f für ein Feld

var

f : array $[0..n-1]$ of Z

und die folgenden prädikatenlogischen Formeln

1. $\exists\, 0 \le i,j < n \bullet f[i] < 0 \Leftrightarrow f[j] > 0$
2. $\exists\, 0 \le i,j < n \bullet f[i] < 0 \wedge f[j] \ge 0$
3. $\exists\, 0 \le i,j < n \bullet f[i] \ge 0 \oplus f[j] \ge 0$
4. $\exists\, 0 \le i,j < n \bullet i \ne 0 \wedge f[i] < 0 \wedge f[j] \ge 0$
5. $\exists\, 0 \le i < n \bullet f[i-1] < 0 \wedge f[i] \ge 0$
6. $\exists\, 0 \le i,j < n \bullet f[i] < 0 \wedge f[j] \ge 0 \wedge (j = i-1 \vee j = i+1)$
7. $\exists\, 0 \le i < n \bullet \exists\, i < j < n \bullet f[i] < 0 \Leftrightarrow f[j] \ge 0$
8. $\exists\, 0 \le i < n \bullet \exists\, i < j < n \bullet f[i] < 0 \wedge f[j] \ge 0$
9. $\forall\, 0 \le i < n \bullet i \bmod 2 = 0 \Rightarrow f[i] \ge 0$
10. $\forall\, 0 \le i < n-2 \bullet i \bmod 2 = 0 \Rightarrow (f[i] \ge 0 \Leftrightarrow f[i+2] \ge 0)$
11. $\forall\, 0 \le i,j < n \bullet (i-j) \bmod 2 = 0 \Rightarrow (f[i] \ge 0 \Leftrightarrow f[j] \ge 0)$
12. $\forall\, 0 \le i,j < n \bullet (i-j) \bmod 2 = 0 \Leftrightarrow (f[i] \ge 0 \Rightarrow f[j] \ge 0)$
13. $\forall\, 0 \le i,j < n \bullet (i-j) \bmod 2 = 0 \Leftrightarrow (f[i] \ge 0 \Leftrightarrow f[j] \ge 0)$

Geben sie für die folgenden Aussagen die Nummer(n) von **gleichwertigen** Formeln an, Mehrfachnennungen sind möglich, gibt es keine Formel tragen Sie 0 an die vorgesehene Stelle ein.

1. Nicht alle Elemente haben das gleiche Vorzeichen.

   ....

2. Alle benachbarten Elemente haben das gleiche Vorzeichen.

   ....

3. Nicht alle benachbarten Elemente haben das gleiche Vorzeichen.

   ....

4. Alle Elemente mit geradem Index sind positiv.

   ....

5. Alle Elemente mit ungeradem Index sind negativ.

   ....

6. Alle Elemente mit geradem Index haben gleiches Vorzeichen.

   ....

7. Jedes Element hat das gleiche Vorzeichen wie sein übernächstes.

   ....

8. Jedes Element hat ein anderes Vorzeichen als seine benachbarten Elemente.

   ....

9. false

   ....

10. Welche Formeln können nicht ausgewertet werden, sind also fehlerhaft aufgebaut.

    ....

Aufgabe 3:

Gegeben sei ein Feld f mit reellen Zahlen. Es soll der Mittelwert der in f gespeicherten Werte in der Variablen $mittel$ berechnet werden.

var f : array $[0..n-1]$ of R;
var i : N_0;
var $mittel$: R;

Die Nachbedingung P lautet also:

$$mittel = \frac{1}{n} * \sum_{j=0}^{n-1} f[j]$$

Die Berechnung soll mit einer Schleife erfolgen, und zwar so, daß zu jedem Zeitpunkt der Berechnung in der Variablen $mittel$ der Mittelwert eines Anfangsstücks von f enthalten ist.

Hierzu ist die Nachbedingung durch eine Einführung einer neuen (Lauf-) Variablen i für die obere Grenze in die Form $I \wedge \neg B$ umzuformen. Es ergibt sich also:

$$mittel = \frac{1}{i} * \sum_{j=0}^{i-1} f[j] \wedge i = mittel$$

Hieraus ergibt sich das folgende Programmschema

$\{ V \}$
$i, mittel := E_0, E_1;$
$\{ I \}$
while B do
$\{ I \wedge B \}$
$\quad i, mittel := E_2, E_3$
$\{ I \}$
end while
$\{ I \wedge \neg B \}$

Wie sieht die Initialisierung von i aus?

E_0 : ....

Wie wird i in der Schleife verändert?

E_2 : ....

Berechnen Sie die Initialisierung von E_1 aus der Invarianten und der Initialisierung von i

$$I[i \leftarrow E_0, mittel \leftarrow E_1]$$

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

E_1 kann nicht immer ausgewertet werden. Welche Vorbedingung V stellt die korrekte Auswertung von E_1 sicher?

V : ....

Berechnen Sie den Ausdruck E_3 für die Veränderung der Variablen *mittel* aus der Invarianten und dem Ausruck E_2

$$I[i \leftarrow E_2, mittel \leftarrow E_3]$$

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

$\Leftrightarrow$ *Begründung :* ....

....

Geben sie einen Ausdruck für die Variante t an, der die Terminierung sicherstellt.

t : ....

Setzen sie alle Einzelteile zu dem vollständigen Progamm zusammen (ohne Kopie der Deklarationen aber mit Zusicherungen für die Vorbedingung, die Invariante und die Nachbedingung)

....

....

....

....

....

....

....

Aufgabe 4:

Gegeben seinen folgende Variablen

var i, m, p, q, r, x, y : N_0
var s : B
var am : R
var f : array $[0..n-1]$ of N_0

Berechnen Sie zu den folgenden Vor- und Nachbedingungen einen geeigneten Ausdruck E, so daß die Zuweisung korrekt arbeitet.

1. $\{$ true $\}$ $i, s := 1, E$ $\{ s \Leftrightarrow (\forall\, 0 < j < i \bullet f[j-1] < f[j] \}$

   E : ....

2. $\{$ true $\}$ $q, r := 0, E$ $\{ x = q * y + r \}$

   E : ....

3. $\{$ true $\}$ $i, s := 0, E$ $\{ s \Leftrightarrow (\exists\, 0 \leq j < i \bullet f[j] \leq m) \}$

   E : ....

4. $\{$ true $\}$ $i, s := 0, E$ $\{ s \Leftrightarrow (\forall\, 0 \leq j \leq i \bullet f[j] \leq m) \}$

   E : ....
