Fachhochschule Wedel 05.02.2008

Prof. Dr. Sebastian Iwanowski

Aufgaben zur Klausur in

***Grundlagen der Theoretischen Informatik* (WS 2007 / 2008)**

Studiengänge B_Inf, B_TInf, B_MInf, B_WInf

Zeit: 90 Minuten,

erlaubte Hilfsmittel: keine

Bitte tragen Sie Ihre Antworten und fertigen Lösungen ausschließlich an den freien Stellen nach den jeweiligen Aufgaben ein (ggf. auf der jeweiligen Rückseite weiterschreiben).

Diese Klausur besteht einschließlich dieses Deckblatts aus 7 Seiten.

Für die Klausur werden insgesamt 50 Bewertungseinheiten (BE) vergeben. Zum Bestehen benötigen Sie mindestens 25 BE.

Viel Erfolg !

1. Aufgabe (10 BE)

Auf einer Party treffen sich Alex, Bernd, Claudia, Dieter und Erna. Da das Bier alle ist, soll jemand in den Keller gehen und neues holen. Da es dort recht gruselig ist, geht keiner gerne alleine. Manche fürchten sich aber auch voreinander.

Folgende Regeln sind zu beachten:

1. Wenn Bernd geht, geht auch Claudia in den Keller.
2. Wenn Erna geht, dann geht auch Dieter.
3. Wenn Claudia geht, dann will Alex unbedingt mit.
4. Wenn Bernd oben bleibt, dann geht Erna in den Keller.
5. Wenn Alex geht, dann bleibt Dieter oben.
6. Wenn Bernd geht, geht Erna in den Keller.

a) Bezeichnen Sie die Aussage „Alex geht in den Keller“ mit A und die Aussage „Alex bleibt oben“ mit der Negation davon, analog für die anderen 4 Personen. Formulieren Sie die 8 Regeln als eine einzelne aussagenlogische Formel mit den Literalen A, B, C, D, E! (3 BE)

b) Bringen Sie die Formel aus a) in KNF! (2 BE)

c) Errechnen Sie eine erfüllbare Belegung der Formel, indem Sie das Resolutionsprinzip auf die Klauseln anwenden (nur so!). Nummerieren Sie alle alten und die durch Resolution neu entstandenen Klauseln sukzessive durch und geben Sie jeweils an, aus welchen Klauseln Sie eine neue resolvieren! (5 BE)

2. Aufgabe (10 BE)

Seien $x,y \in \mathbb{R}$ für die folgenden 3 Prädikate:

P1(...): $y^2 > x$

P2(...): $\forall x < 0: y^2 > x$

P3(...): $\forall x \leq 0: y^2 < x$

a) Geben Sie für jedes Prädikat an, von welchen Variablen es abhängt. (2 BE)

b) Entscheiden Sie dann für jedes Prädikat, ob es gültig, widerlegbar, erfüllbar oder widersprüchlich ist (jeweils genau 2 Antworten!) und begründen Sie jede Antwort. (5 BE)

c) Bilden Sie die logische Negation jedes der oben genannten Prädikate. (3 BE)

3. Aufgabe (8 BE)

a) Finden Sie zum folgenden Programmausschnitt und der gegebenen Nachbedingung die schwächste Vorbedingung! Geben Sie alle Zwischenschritte Ihrer Beweiskette an! Vereinfachen Sie die Bedingungen, wo es sinnvoll ist (aber nur dann)! (6 BE)

b) Geben Sie eine Variablenbelegung an, die den then-Block durchläuft und die Nachbedingung erfüllt, und eine, die den else-Block durchläuft und die Nachbedingung erfüllt! (2 BE)

```
if (x - y > 0)

   then

      x := x - y

   else

      y := x - y;

{y – x > 0}
```

4. Aufgabe (10 BE)

Betrachten Sie folgendes Programm:

```
Gegeben seien n Zahlen a[1], ..., a[n] (also n ∈ ℕ\{0}).

m := 0;
k := 0;
while (k < n) do
  begin
    k := k + 1;
    if m < a[k]
      then
        m := a[k]
  end
```

a) Geben Sie jeweils eine Nachbedingung für k und m an! (2 BE)

b) Beweisen Sie a):

Benennen Sie dafür den Wert der Variablen k nach i Schleifendurchläufen mit k_i und den der Variablen m mit m_i, beweisen Sie invariante Bedingungen für diese Werte mit vollständiger Induktion und folgern Sie daraus die Terminierung der Schleife und die Aussage von a)! (8 BE)

5. Aufgabe (6 BE)

Gegeben sei folgende Funktion f:

```
function f(x: ℕ): ℕ;
begin
 if (x=0)  then return 5
           else return (1-x)^2 * f(x-1)
end;
```

a) Von welchem Typ ist die Rekursion? Wenn mehrere Antworten möglich sind, dann geben Sie nur den speziellsten Typ an! Begründen Sie Ihre Antwort! (2 BE)

b) Was berechnet die Funktion in Abhängigkeit von x? Geben Sie eine informelle Begründung für Ihre Aussage ab! (4 BE)

6. Aufgabe (6 BE)

Gegeben sei das Array [2 7 3 4]. Dieses soll sortiert werden.

a) Geben Sie die Namen von 2 Algorithmen an, die diese Aufgabe lösen können! Geben Sie deren Laufzeit im schlechtesten Fall in Abhängigkeit von der Größe n des Arrays mit Hilfe von Landau-Symbolen an! (2 BE)

b) Skizzieren Sie durch Angabe von Zwischenschritten, wie der für große n **langsamere** Algorithmus mit den oben genannten Eingabedaten arbeitet! (3 BE)

c) Nennen Sie einen Nachteil des **schnelleren** Algorithmus gegenüber anderen Verfahren! (1 BE)