
Aufgaben zur Klausur in
Grundlagen der Theoretischen Informatik (WS 2008 / 2009)
Studiengänge B_Inf, B_TInf, B_MInf, B_WInf

Zeit: 90 Minuten,
erlaubte Hilfsmittel: keine

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

Diese Klausur besteht einschließlich dieses Deckblatts aus 6 Seiten.

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

Viel Erfolg !

1. Aufgabe (10 BE)

- i) Bringen Sie die folgenden Formeln in konjunktive Normalform! Vereinfachen Sie die Formeln so weit wie möglich!
- ii) Geben Sie jeweils an, aus wie vielen Klauseln Ihr Ergebnis besteht!
- iii) Geben Sie außerdem an, ob es sich um eine Tautologie handelt oder um eine Formel, die zugleich erfüllbar und widerlegbar ist oder um einen Widerspruch. Begründen Sie Ihre Antwort!

a) $p \rightarrow (\neg p \vee q)$ (3 BE)

b) $\neg(p \vee (\neg p \wedge q) \vee \neg q)$ (3,5 BE)

c) $\neg(p \rightarrow q) \rightarrow p$ (3,5 BE)

3. Aufgabe (8 BE)

Finden Sie zum folgenden Programmausschnitt und der gegebenen Nachbedingung die schwächste Vorbedingung! Geben Sie alle Zwischenschritte Ihrer Beweiskette an!

```
if (x = 1)
  then
    begin

      y := - x;

      x := x / y;

    end
  else
    begin

      x := x / y;

      y := -x;

    end

end

{x = 1}
```

4. Aufgabe (9 BE)

Gegeben sei das folgende Programm:

```
procedure Euklid (x,y: integer)
{ y > 0 }                Vorbedingung
  q := 0;
  r := x;
  while r ≥ y do
    begin
      r := r - y;
      q := q + 1
    end
{ r = x - q * y < y }    Nachbedingung
```

Beweisen Sie, dass aus der angegebenen Vorbedingung die angegebene Nachbedingung folgt, indem Sie folgendermaßen vorgehen:

- i) Stellen Sie eine Invariante auf, die von den Werten der Variablen nach i Schleifendurchläufen abhängt (Tipp: Orientieren Sie sich an der Nachbedingung!).
- ii) Beweisen Sie die Invariante durch vollständige Induktion nach i .
- iii) Beweisen Sie die Terminierung und folgern Sie die geforderte Nachbedingung.

5. Aufgabe (10 BE)

Gegeben sei die folgende Funktion:

```
procedure f (n, m: integer): integer
begin
  if (n = m)
    then return m * (m - n)
    else return n + f (n, m - 1)
end {f}
```

- a) Von welchem Rekursionstyp ist die Rekursion? Geben Sie eine Begründung an! (2 BE)
- b) Nennen Sie eine Vorbedingung, die erfüllt sein muss, damit f überhaupt einen Wert berechnet! Geben Sie an, wie sich die Implementierung verhält, wenn diese Vorbedingung nicht erfüllt ist! (2 BE)
- c) Was berechnet diese Funktion, wenn die Vorbedingung erfüllt ist, in Abhängigkeit von n und m? (2 BE)
- d) Beweisen Sie c) durch vollständige Induktion nach einem der Parameter! (4 BE)
(Hinweis: Die Verankerung hängt vom anderen Parameter ab!)