

**Aufgaben zur Übergangsprüfung in
*Grundlagen der Programmierung (SS 2005)***

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 Deckblattes aus 6 Seiten.

Für die Prüfung werden insgesamt 40 Bewertungseinheiten (BE) vergeben. Zum Bestehen benötigen Sie mindestens 20 BE.

Viel Erfolg !

1. Aufgabe (6 BE)

Bringen Sie die folgende Formel in konjunktive Normalform!

Entscheiden Sie, ob es sich um eine Tautologie handelt.

Ist die Formel erfüllbar?

Geben Sie jeweils eine Begründung an!

$$(\neg p \wedge q) \vee (\neg q \wedge z) \vee (\neg z \wedge p)$$

2. Aufgabe (12 BE)

Gegeben seien die folgenden Prädikate mit den zugehörigen Bedeutungen:

$L(x,y)$	x liebt y
$H(x,y)$	x hasst y
$F(x)$	x ist weiblich
$M(x)$	x ist männlich
$K(x,y)$	x ist ein Kind von y
$V(x,y)$	x ist mit y verheiratet

Drücken Sie folgende Sachverhalte durch eine Verknüpfung der oben stehenden Prädikate aus (Sie dürfen nicht voraussetzen, dass der Vorname automatisch das Geschlecht festlegt):

a) Karin ist eine Tochter von Klaus.

b) Susi ist eine Schwester von Anne.

c) Max hat nur Töchter.

d) Jede Mutter liebt ihr Kind.

e) Jede Person ist mit maximal einer Person verheiratet.

f) Jeder Mann hasst alle anderen Liebhaber seiner Frau.

3. Aufgabe (7 BE)

Finden Sie zum folgendem Programmausschnitt und der gegebenen Nachbedingung die schwächste Vorbedingung und vereinfachen Sie diese so weit wie möglich! Geben Sie alle Zwischenschritte Ihrer Beweiskette an!

```
if x > 0
```

```
  then
```

```
    x := -x
```

```
  else
```

```
    x := (x+1)2;
```

```
y := x + 5; {gehört nicht mehr zur Verzweigung}
```

```
{y < 6}
```

4. Aufgabe (9 BE)

Gegeben sei die folgende Prozedur:

```
procedure interessanteSchleife (n): integer
begin
  i := 0; s := 0;

  while i ≤ n do
  begin
    i := i + 1;
    s := i + s;
  end {while}

  return result;
end {interessanteSchleife}
```

4.1 Was berechnet die Prozedur? Geben Sie die genaue Abhängigkeit von n an!

4.2 Beweisen Sie Ihre Aussage mit einer Ihnen geläufigen Verifikationstechnik!

5. Aufgabe (6 BE)

Gegeben sei die folgende Prozedur:

```
procedure rekursiv (m,n): integer
begin

    if n ≤ 0
        then return 0
        else return m + n + rekursiv (m, n-1)

end {rekursiv}
```

5.1 Geben Sie die Werte von `rekursiv(2,2)`, `rekursiv(2,1)` und `rekursiv(1,1)` an!

5.2 Was berechnet diese Prozedur im Allgemeinen? Geben Sie eine exakte Antwort in Abhängigkeit von m und n an!