

Aufgaben zur Klausur Grundlagen der Programmierung im SoSe 03
Zeit: 60 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). Sollten Unklarheiten oder Mehrdeutigkeiten bei der Aufgabenstellung auftreten, so notieren Sie bitte, wie Sie die Aufgabe interpretiert haben.

Viel Erfolg !

Diese Klausur besteht einschließlich dieses Deckblattes aus 6 Seiten.

Aufgabe 1.

Gegeben sei folgende aussagenlogische Formel: $(A \wedge (B \wedge C)) \wedge ((A \wedge B) \wedge C)$.
Zeigen Sie durch Transformation und Vereinfachung, daß die Formel eine Tautologie ist.

Aufgabe 2.

Sei P ein Prädikatensymbol, f ein Symbol für eine einstellige Funktion und a eine Konstante. Bestimmen Sie die Wahrheitswerte der prädikatenlogischen Formeln $(\forall x (P(x, f(x))))$ und $(\exists x (P(f(x), x)))$ bezüglich Struktur $A = (\mathbb{N}, I)$, wobei \mathbb{N} die Menge der natürlichen Zahlen ist und I wie folgt definiert sei:

$$I(a) = 1$$

$$I(f) = n \mapsto n + 1, f \text{ bildet also jedes } n \text{ aus } \mathbb{N} \text{ auf seinen Nachfolger } n+1 \text{ ab.}$$

$$I(P) = \{ (n,m) \mid m > n \}$$

Hinweis: Da das Universum von A unendlich ist, lassen sich nicht alle Fälle hinschreiben. Sie müssen also bei der Ermittlung des Wahrheitswertes über alle Werte argumentieren. Bitte schreiben Sie Ihre Argumente gut lesbar auf.

Aufgabe 3.

Bitte beschreiben und begründen Sie, was der folgende Algorithmus berechnet, wenn für den Parameter a ein aufsteigend sortiertes Array und für q ein Element aus N_0 angegeben wird.

```
tuwas(a : array [1..N] of N0, q : N0): N0
begin
  var x, l, r : N0;
  l, r := 1, N;
  repeat
    x := (l + r) div 2;
    if q < a[x]
      then r := x-1
      else l := x+1
    end if
  until (q = a[x] or (l >= r));
  if q = a[x]
    then x
    else 0
  end if
end
```

Welche Worst-Case-Komplexität hat der Algorithmus? Bitte begründen Sie Ihre Antwort.

Spezifizieren Sie als Vorbedingung für den Algorithmus ein Prädikat, das ausdrückt, daß nur ein aufsteigend sortiertes Array als Wert des ersten Parameters angegeben werden darf.

Aufgabe 4.

Geben Sie für den Algorithmus aus Aufgabe 3 ein rekursives Programm an. Das Programm soll aus einer Funktion f mit den gleichen Parametern wie $tuwas$ aus Aufgabe 3 bestehen. Die zu schreibende rekursive Funktion soll von f aus aufgerufen werden.

Aufgabe 5.

In einer Version eines bekannten Betriebssystems, das nun auf relativ einfachen Mobiltelefonen des Modells „BG“ angeboten werden soll, sind Namen für lokale Dateien auf Zeichen eines speziellen Alphabets beschränkt. Die Zeichen sind Elemente eines speziellen Alphabets $\{b, g, 0, 1, .\}$. Dateinamen bestehen aus zwei Teilen, einem Hauptnamen mit genau fünf Zeichen aus $\{b, g, 0, 1\}$ und einer Extension mit drei Zeichen aus $\{b, g\}$. Hauptname und Extension werden durch das Zeichen $.$ (Punkt) getrennt. Der Hauptname soll nicht mit einem Zeichen aus $\{0, 1\}$ beginnen. Allerdings soll dafür gesorgt werden, daß in der Extension nach zwei b 's mindestens ein g kommt.

Geben Sie einen endlichen Automaten an, die Sprache der oben beschriebenen Dateinamen akzeptiert. Geben Sie bitte die Menge der Zustände, die Menge der Startzustände (ggf. kann es nur einen Startzustand geben) sowie auch die Menge der Endzustände an. Die Übergangsrelation können Sie graphisch angeben.