

Klausur zur Informatik I

Institut für Informatik
Justus-Liebig-Universität Gießen
Wintersemester 2002/03

Name:	18.02.2003
Vorname:	
Geburtsdatum:	
Geburtsort:	
Studiengang Computerlinguistik:	<input type="checkbox"/> JA <input type="checkbox"/> NEIN

Punktzahl: 1 2 3 4 5 6 Σ

Note:

Aufgabe 1 (5 Punkte)

Eine Gruppe Studenten im 1. Semester hat das folgende Pascal-Programm zur Prüfung von Palindromen geschrieben. Leider haben sich dabei verschiedene Fehler eingeschlichen; manche auf den ersten Blick fehlerhafte Anweisungen sind jedoch korrekt und werden vom Compiler verstanden.

Finden, korrigieren und begründen Sie die Fehler in dem Programm.

Vorsicht: Das Korrigieren von korrekten oder lediglich überflüssigen Anweisungen führt zu Punktabzug!

Achten Sie sowohl auf syntaktische als auch inhaltliche Fehler!

UNIT aufgabe 1;
 VAR from, to : STRING;
 i : REAL;

FUNCTION reverse(s : STRING;
 VAR resultat : STRING
) : BOOLEAN;
 VAR lenght : INTEGER;

BEGIN;
 resultat := '';
 FOR i := length(s)-1 DOWN TO 0 DO
 resultat := resultat+s[length(s)-i];
 reverse := s = resultat
 END;

FUNCTION palin (s : STRING) : BOOLEAN;
 BEGIN
 palin := Reverse(s,s)
 END;

BEGIN
 READLN(from);
 IF reverse(from,to) THEN WRITELN('TRUE');
 WRITELN(to); READLN(from);
 BEGIN reverse(from,from); END;
 WRITELN(palin(from));
 END;

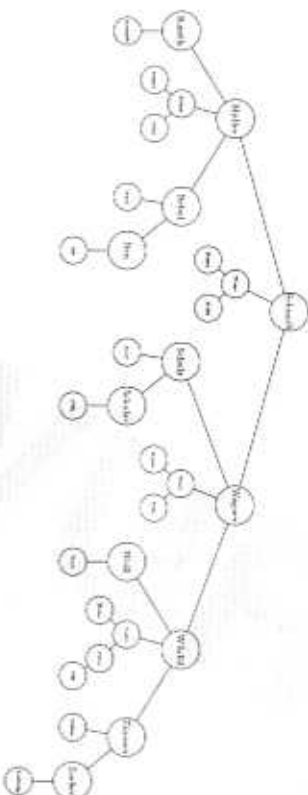
Markieren Sie jeden Fehler mit einer Nummer und geben Sie auf einem gesonderten Blatt die Begründung an.

Aufgabe 2 (5 Punkte)

Schreiben Sie Funktionen/Prozeduren zum Einfügen und Ausgeben von Personendaten eines wie folgt aufgebauten sortierten Baums.

Die Sortierung soll nach Nachnamen erfolgen; bei gleichen Nachnamen erstellen Sie in dem betreffenden Knoten jeweils einen nach Vornamen sortierten zusätzlichen Binärbaum.

Beispiel:



Geben Sie die verwendeten Datenstrukturen und die Funktionen/Prozeduren an.

Aufgabe 3 (5 Punkte)

Welche der folgenden Sprachen sind kontextfrei? Geben Sie eine kontextfreie Grammatik an oder begründen Sie, warum die Sprache nicht kontextfrei ist.

- (a) $L_1 = \{a^n b^m c^n \mid m, n \in \mathbb{N}_0\}$
- (b) $L_2 = \{a^n c^n \mid m, n \in \mathbb{N}_0\}$
- (c) $L_1 \cup L_2$
- (d) $L_1 \cap L_2$

Ist eine der Sprachen L_1 bzw. L_2 in der jeweils anderen enthalten?

Aufgabe 4 (5 Punkte)

Schreiben Sie ein Pascal-Programm für folgendes Spiel: Ein Spieler gibt zunächst einen kurzen Satz ein. Die anderen Spieler sollen durch Raten von einzelnen Buchstaben den eingegebenen Satz bestimmen.

Dazu zeigt das Programm für jeden noch unbekannten Buchstaben das Zeichen '_' an und zählt die Fehleingaben. Nach 10 Fehlern haben die ratenden Spieler verloren.

Beispiel:

```
-----e_e_e_!
Fehler: 3
```

Nächstes Zeichen: a

```
A___a_e_e_e_!
```

Fehler: 3

Nächstes Zeichen: y

```
A___a_e_e_e_!
```

Fehler: 4

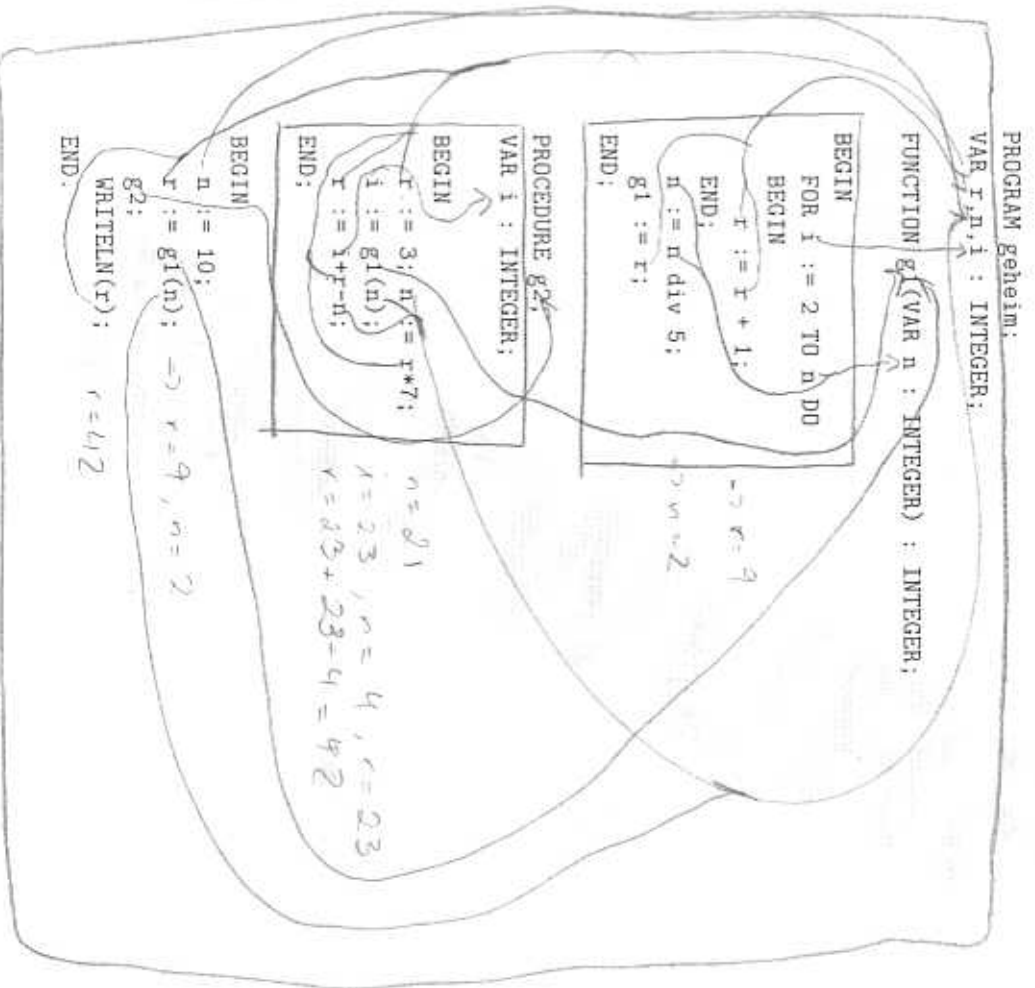
Es gelten folgende Vereinbarungen

- Der eingegebene Satz enthält keine Umlaute.
- Satzzeichen werden grundsätzlich angezeigt.
- Das Raten eines bereits angezeigten Buchstabens ist kein Fehler.

Tipp: Die Funktion `uppercase` liefert zu einem übergebenen Zeichen vom Typ `char` den zugehörigen Großbuchstaben, d.h. `uppercase('a')` ergibt z.B. `'A'` und `uppercase('A')` ergibt ebenfalls `'A'`.

Aufgabe 5 (5 Punkte)

Welche Ausgabe bewirkt das nachfolgende Programm? Begründen Sie Ihre Antwort und geben sie die Block- und Prozedurschachtelung sowie die Bindungsbeziehungen an.



Aufgabe 6 (5 Punkte)

Seien P und Q logische Formeln, und sei S eine Anweisung. Dann heißt S *partiell korrekt bzgl. P und Q* (in Zeichen $\{P\} S \{Q\}$), falls folgendes gilt:

P gilt im Zustand vor Ausführung von S und S terminiert regulär
 $\implies Q$ gilt im Zustand nach Ausführung von S .

Wir sagen dann auch, dass die Hoaresche Zusicherung $\{P\} S \{Q\}$ im Sinne *partieller Korrektheit* gilt oder *gültig* ist. Zur Erinnerung: Die für die folgenden Aufgaben relevanten Regeln zur Herleitung gültiger Hoarescher Zusicherungen, das Axiom für die Zuweisung, die Regel für die while-Schleife sowie die Konsequenzregel lauten:

$$\frac{}{\{Q[x/a]\} x := a \{Q\}} [\text{ass}] \qquad \frac{P \Rightarrow P_1, \{P_1\} S \{Q_1\}, Q_1 \Rightarrow Q}{\{P\} S \{Q\}} [\text{impl}]$$

$$\frac{\{P \wedge b\} S \{P\}}{\{P\} \text{ while } b \text{ do } S \{P \wedge \neg b\}} [\text{while}]$$

- (a) Welche Eigenschaft muss die Anweisung S haben, damit die folgende Zusicherung im Sinne partieller Korrektheit gültig ist? Geben Sie auch ein Beispiel für S an.

$\{\text{true}\} S \{\text{false}\}$

- (b) Welche der folgenden Hoareschen Zusicherungen gelten, welche nicht? (Begründen Sie Ihre Antworten!)

- (i) $\{x = a\} x := x + 1 \{x = a + 1\}$
- (ii) $\{x = x\} x := x + 1 \{x = x + 1\}$

- (c) Leiten Sie die folgende Hoaresche Zusicherung mit den o.a. Regeln her:

$$\{x \geq 0\} \text{ while } x < 0 \text{ do } x := x - 1 \{x = 0\}$$

Beachten Sie, dass $x - 1 \geq 0$ gilt, wenn x größer 0 ist. Welches ist die Schleifeninvariante?

- (d) Die Vorbedingung $\{x \geq 0\}$ im Aufgabenteil (c) charakterisiert die Menge derjenigen Zustände, in denen x einen Wert ≥ 0 hat. Geben Sie eine (schwächere) Vorbedingung P an, die in möglichst vielen Zuständen gilt und für die immer noch

$$\{P\} \text{ while } x < 0 \text{ do } x := x - 1 \{x = 0\}$$

im Sinne partieller Korrektheit gültig ist. Begründen Sie Ihre Antwort.

Viel Erfolg!