Vordiplom Wirtschaftswissenschaften (WH)

Allgemeine Informatik I

SS 1999/2000

4. Mai 2000

Name ______________________________

Vorname ______________________________

Matr.Nr. ______________________________

Aufgabe		Punkte	Bewertung
1	a)	4	
	b)	6	
	c)	4	
2	a)	2	
	b)	2	
	c)	4	
	d)	4	
	e)	2	
3	a)	4	
	b)	2	
	c)	2	
	d)	2	
4		10	
5		6	
6		12	
7		14	
Summe		**80**	

☞ Bearbeitungszeit: 120 Min.

☞ Sollte der vorgesehene Platz zur Berabeitung einer Aufgabe nicht ausreichen, so verwenden Sie bitte die Rückseite (mit erkennbarer Zuordnung zur Aufgabenstellung!

Viel Erfolg!

Aufgabe 1 (14 Punkte)

Die formale Sprache **L** über dem Vokabular **V = {0,1}** sei verbal wie folgt definiert:

L = { alle mit ‘01’ beginnenden Zeichenfolgen aus V^*,
bei denen die Anzahl der unmittelbar nebeneinander stehenden
Einsen ungerade ist, Folgen von Einsen durch genau zwei
Nullen getrennt sind und die Gesamtfolge mit zwei Nullen endet }

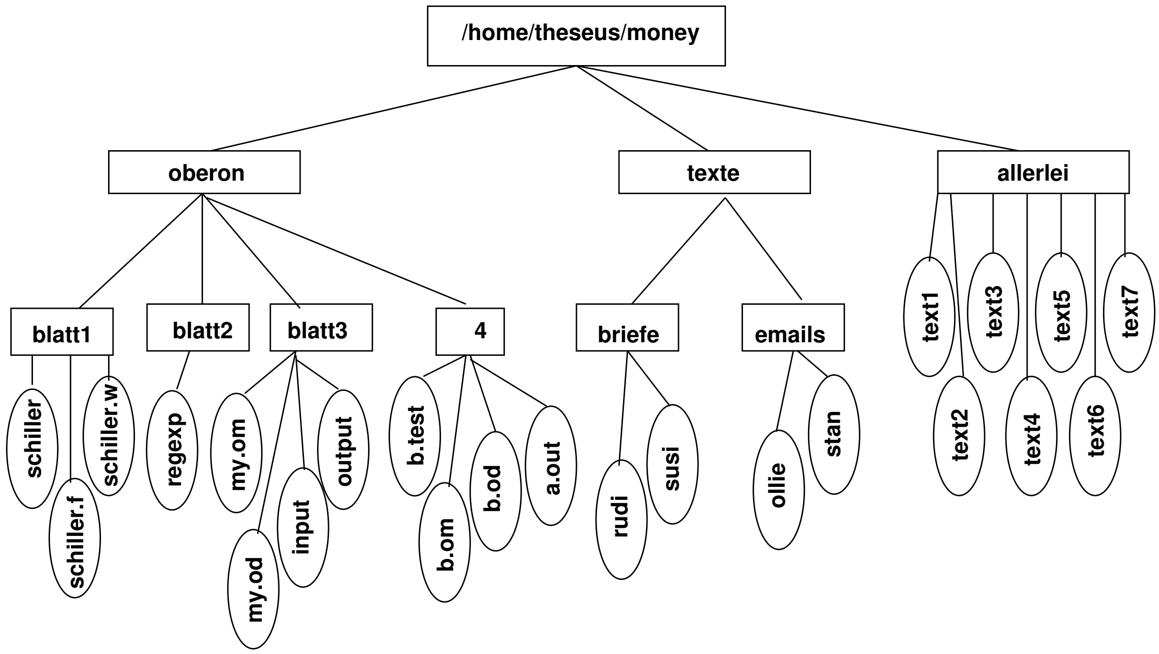
(*a*) 4 Punkte
Beschreiben Sie L mittels einer regulären Grammatik!

(*b*) 6 Punkte
Beschreiben Sie L mittels eines regulären Ausdrucks!

(*c*) 4 Punkte
Beschreiben Sie L mittels eines endlichen deterministischen Automaten (in grafischer Darstellung)!

Aufgabe 2 (14 Punkte)

Gegeben sei in grafischer Darstellung ein Auszug aus der Verzeichnisstruktur des Hörers der Vorlesung "Allgemeine Informatik I" mit dem Login-Namen *money* (Die Rechtecke stellen Katalogdateien, die Ovale reguläre Dateien dar):



(*a*) 2 Punkte
money hat sich angemeldet und ist in den Katalog *texte* gewechselt. Da er im Katalog *allerlei* nur Texte hält, will er den gesamten Katalog *allerlei* als Unterkatalog gleichen Namens in *texte* verschieben. Geben Sie das notwendige Kommando an!

`theseus$` ____

(*b*) 2 Punkte
Er möchte nun in seinem Heimatkalatog einen Unterkatalog namens *bin* anlegen, ohne seinen aktuellen Arbeitskatalog *texte* zu wechseln. Geben Sie das notwendige Kommando an!

`theseus$` ____

(*c*) 4 Punkte

Das im Rahmen der 3. Übungsaufgabe (*blatt3*) erstellte Programm möchte er noch einmal zu Demonstrationszwecken übersetzen. Nach der (erfolgreichen) Übersetzung sollen alle vom Compiler generierten Dateien mit Ausnahme des ausführbaren Programmes entfernt werden. Zum Test des (ausführbaren) Programmes sollen die Eingaben via Pipeline aus der Datei *input* kommen, die Ausgaben sollen via Ausgabe-Umlenkung in die Datei *output* gehen. Geben Sie die notwendigen Kommandos an. Sie können davon ausgehen, dass Ihr Arbeitskatalog bereits *blatt3* ist. Die Zahl der Kommandozeilen, die im folgenden angegeben ist, hat nichts mit der Zahl der notwendigen Kommandos zu tun!

```
theseus$ ________________________________________________

theseus$ ________________________________________________

theseus$ ________________________________________________

theseus$ ________________________________________________

theseus$ ________________________________________________

theseus$ ________________________________________________

theseus$ ________________________________________________
```

(*d*) 4 Punkte

Sie stehen im Katalog *blatt1*. Geben Sie an, was nach Ausführung der folgenden beiden Kommandos im Katalog *blatt1* enthalten ist:

I) **rm *.***

II) **rm schiller***

(*e*) 2 Punkte

Sie stehen im Katalog *allerlei*. Was bewirkt folgendes Kommando?

rm *[1-3]

Aufgabe 3 (10 Punkte)

(*a*) 4 Punkte
Was leistet folgende Kommandozeile?

```
theseus$ cat file | egrep '^.$' | wc -l
```

(*b*) 2 Punkte
Sie haben drei Textdateien *einleitung*, *hauptteil* und *schluss*, die in dieser Reihenfolgen zu einer Datei *gesamt* zusammengefügt werden sollen. Geben Sie die dazu notwendige Kommandozeile an.

`theseus$` ____

(*c*) 2 Punkte
Sie sitzen spätabends alleine im Workstation-Pool und wollen mit dem Kommando **a2ps** einen Programmtext in einer bestimmten Formatierung ausdrucken. Da Sie die Optionen von *a2ps* nicht auswendig wissen und momentan niemanden fragen können, sollten Sie den Weg nennen, wie Sie sofort und nur mit Hilfe des Rechners die entsprechende Option von *a2ps* herausfinden können.

`theseus$` ____

(*d*) 2 Punkte
Sie haben erfolgreich ein Oberon-Programm geschrieben, das als ausfühbare Datei namens **endlich** vorliegt. Sie starten dieses Programm und warten. Es passiert aber nichts, auch die Shell liefert ihren Prompt nicht. Nach 5 Minuten sind sie sicher, daß Ihnen wohl ein "Dauerrenner" geglückt ist. Wie bringen Sie diesen zur Beendigung?

`theseus$  endlich`

____

Aufgabe 4 (10 Punkte)

Gegeben sei die folgende mathematische Funktion:

$$f(x) = \begin{cases} -2x + 1.5, & f\ddot{u}r\ \ x < 0 \\ x^3 + 1.5, & f\ddot{u}r\ \ 0 \le x \le 2 \\ 9.5 + x, & f\ddot{u}r\ \ x > 2 \end{cases}$$

Realisieren Sie diese Funktion als Oberon-Prozedur, so daß Sie der mathematischen Semantik entsprechend benutzt werden kann (also etwa in der Form $f(x) + 3.14 * f(y)$)!

Aufgabe 5 (6 Punkte)

Gegeben ist folgendes Oberon-Programm (das “od”-File erübrigt sich hier):

```
MODULE Do;
   IMPORT Write;
   VAR a,b,c: INTEGER;
   PROCEDURE Magic(VAR x: INTEGER; y: INTEGER);
   BEGIN
      a := x;
      x := y;
      a := x;
   END Magic;

BEGIN
   a := 3; b := 7; c:= 11;
   Magic(b,c);
   Write.String("a = "); Write.Int(a,0); Write.Ln;
   Write.String("b = "); Write.Int(b,0); Write.Ln;
   Write.String("c = "); Write.Int(c,0); Write.Ln;
END Do.
```

Geben Sie an, welche Ausgabe dieses Programm liefert!

Aufgabe 6 (12 Punkte)

Umwandlung von Hexadezimal- in Dezimalzahlen

Hexadezimalzahlen können bekanntlich als Folgen von Zeichen aus der Menge {0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F} betrachtet werden, wobei die “Ziffer” A den Dezimalwert 10 hat, B den Dezimalwert 11, usw.

Formulieren Sie eine Oberon-Funktionsprozedur *MakeDec*, die als ersten Parameter eine solche Zeichenfolge bekommt und über einen zweiten Parameter den entsprechenden Dezimalwert zurückliefert. Mit dem Funktionsrückgabewert TRUE soll signalisiert werden, daß der Parameter für die Hexadezimal-Zahl korrekt war und der Dezimalwert bestimmt werden konnte. Mit dem Rückgabewert FALSE soll signalisiert werden, daß der Parameter für die Hexadezimalzahl nicht korrekt war (also ungültige Zeichen enthielt) und somit der über den zweiten Parameter gelieferte Dezimalwert keinen sinnvollen Wert enthält.

Die Prozedur soll in einem Programm wie folgt verwendet werden können:

```
hexNumber := "1F";
IF ~ MakeDec(hexNumber,dec) THEN
     Write.String("ERROR!");
ELSE
     Write.Int(dec,0);
END;
```

Lösung:

Aufgabe 7 (14 Punkte)

Schreiben Sie eine Oberon-Funktionsprozedur **CountLength**, die in einem als Parameter übergebenen String feststellt, welches Zeichen am häufigsten hintereinanderstehend vorkommt. Sind mehrere Zeichen gleich häufig vertreten, soll das am weitesten vorne stehende Zeichen ausgegeben werden.

Das Programm soll wie folgt benutzt werden können:

```
MODULE Count;
  IMPORT ASCII, Write;
  CONST maxlen = 100;
  VAR     f: ARRAY maxlen OF CHAR;
     n: INTEGER;
     ch: CHAR;
   (* Hier kommen weitere Vereinbarung*)
   (* und hier Ihre Prozedur *)
BEGIN
  f := "SStuttgart";
  n := CountLength(f,ch);
  IF n > 0 THEN
    Write.Char(ch); Write.Int(n,4);
  END;
  Write.Ln;
END Count.
```

Ausgabe:

```
S    2
```

Lösung: