Vordiplom Wirtschaftswissenschaften
Allgemeine Informatik I
WS 2001/2002
12. April 2002
Beispiellösung

Bearbeitungszeit: 120 Minuten

Name:
Vorname:
Matrikelnummer:

Aufgabe	Punkte	Bewertung
1	**12**	
a)	2	
b)	2	
c)	2	
d)	2	
e)	4	
2	**10**	
a)	4	
b)	6	
3	**8**	
a)	4	
b)	4	
4	**14**	
a)	6	
b)	8	
5	**14**	
6	**8**	
7	**8**	
8	**6**	
9	**10**	
Summe:	**90**	
	Note:	

Bitte benutzen Sie für die Lösungen den freigelassenen Platz nach der jeweiligen Angabe; sollte dieser nicht ausreichen, verwenden Sie bitte die Rückseite, wobei die Zuordnung zur jeweiligen Aufgabe deutlich erkennbar sein muss!
Viel Erfolg!!!

Aufgabe 1 **(12 Punkte)**

In Ihrem Heimatverzeichnis befinden sich u.a. die drei Verzeichnisse *tmp*, *allgInfo* und *data*. In *allgInfo* existieren zwei Unterverzeichnisse *blatt1* und *blatt2*.

(a) **2 Punkte**

Sie befinden sich in Ihrem Heimatverzeichnis. Für das 3. Übungsblatt haben Sie im Katalog *tmp* einen Katalog *blatt3* angelegt, in dem die Lösungsdateien zu diesem Blatt enthalten sind. Um Ihre Systematik aufrechtzuerhalten, sollte der Katalog *blatt3* in den Katalog *allgInfo*. Geben Sie an, wie Sie dies mit einem einzigen UNIX-Kommando bewerkstelligen!
Lösung:

```
thales$ mv tmp/blatt3 allgInfo
```

(b) **2 Punkte**

Sie haben in Ihrem Heimatverzeichnis – noch immer Ihr Arbeitskatalog – viele Dateien, deren Namen mit einer Ziffer beginnt. Alle diese – und nur diese – sind mit einem einzigen UNIX-Kommando zu löschen:
Lösung:

```
thales$ rm [0-9]*
```

(c) **2 Punkte**

Ihr Arbeitskatalog ist nun der Unterkatalog *tmp* Ihres Heimat-Katalogs. In der Vielzahl von Dateien, die sich hier angesammelt haben, gibt es genau eine, in den die Datumsangabe 12.4.2002 vorkommt. Geben Sie an, wie sie deren Namen herausfinden können!
Lösung:

```
thales$ egrep '12\.4\.2002' *
```

(d) **2 Punkte**

Ihr Arbeitskatalog ist immer noch der Unterkatalog *tmp* Ihres Heimatkatalogs. Sie wollen von hier in den Unterkatalog *data* Ihres Heimatkatalogs wechseln. Geben Sie **zwei verschiedene** Kommandozeilen an, mit denen Sie dies bewerkstelligen können!
Lösung:

```
thales$ cd ../data
```

```
thales$ cd $HOME/data
```

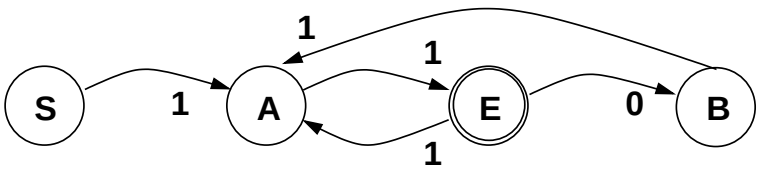
(e) **4 Punkte**

Welches der folgenden UNIX-Kommandos ist **kein** Filter? Bitte in **jeder** Zeile jeweils ankreuzen! Falsche Antworten führen ggf. zu Punktabzug!

Kommando	ist Filter	ist kein Filter
ls		X
sort	X	
ps		X
du		X

Aufgabe 2 **(10 Punkte)**

Eine Sprache über dem Vokabular $V = \{0, 1\}$ sei durch folgenden endlichen (deterministischen) Automaten definiert:



Der Startzustand ist **S**, der Endzustand ist **E**. Nicht angegebene Zustandsübergänge sind als Übergänge in einen hier nicht zu betrachtenden Fehlerzustand zu sehen.

a) **(4 Punkte)**
Kreuzen Sie in folgender Tabelle entweder in Spalte "wahr" oder "falsch" an:

Dieser Satz gehört zur Sprache	wahr	falsch
1110		X
1111011	X	
11001111		X
110110111		X

b) **(6 Punkte)**
Geben Sie für die durch obigen Automaten definierte Sprache eine äquivalente Definition mittels EBNF (Produktionsregeln) an!

Lösung:

$S \to 1A$, $A \to 1E$, $E \to 1A$, $E \to 0B$, $B \to 1A$, $E \to \epsilon$

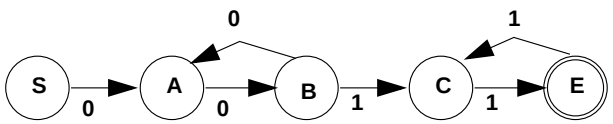
Aufgabe 3 **(8 Punkte)**

Eine formale Sprache über dem Vokabular $V = \{0, 1\}$ sei verbal wie folgt definiert:

> *alle die Folgen von Symbolen aus V, bei denen am Anfang eine gerade Anzahl (mindestens 2) von Nullen steht, auf die eine gerade Anzahl (mindestens 2) von Einsen folgt*

Beispiele: 000011 gehört dazu, 00011 nicht, da die Teilfolge von Nullen am Anfang die Länge 3 hat!

a) **(4 Punkte)**
Geben Sie einen endlichen (deterministischen) Automaten an, der diese Sprache definiert.
Lösung:



b) **(4 Punkte)**
In einem Textsystem wie *MS-WORD* ist unter der Funktion *Suchen* die Einstellung *Erweitern* möglich, die auf die Suche *Mit Mustervergleich* (regulären Ausdrücken) führt. Unter dem Begriff *Sonstiges* werden mögliche Meta-Symbole aufgeführt:

Meta-Symbol	Bedeutung	Meta-Smbol	Bedeutung
<	Wortanfang	>	Wortende
()	Ausdruck	@	Vorkommen 1 oder mehr

Formulieren Sie damit einen regulären Ausdruck, der alle die **Worte** aus einem Text identifiziert, die zu obiger Sprache gehören!
Lösung:

```
<(00)@(11)@>
```

Aufgabe 4 **(14 Punkte)**

a) (6 Punkte)

Schreiben Sie eine Oberon-Prozedur **NumberOfDigits**, die als Parameter eine ganze Zahl (Typ: *INTEGER*) erhält und die Anzahl der Dezimalstellen dieser Zahl zurückliefert!

Lösung:

```
PROCEDURE NumberOfDigits(z: INTEGER):INTEGER;
  VAR i: INTEGER;
BEGIN
   i:=0;
   IF z = 0 THEN RETURN 1 END;
   IF z < 0 THEN z := -z END;
   WHILE z > 0 DO
      z := z DIV 10;
      INC(i);
   END;
   RETURN i
END NumberOfDigits;
```

Fortsetzung nächste Seite

b) (8 Punkte)

Schreiben Sie eine Oberon-Prozedur *Int2Asc*, die als Parameter ein ganze Zahl (Typ: *INTEGER*) erhält und über einen Parameter den String, bestehend aus dem Vorzeichen dieser Zahl gefolgt von den Ziffern liefert. Die Zahl 0 ("Null") soll als +0 geliefert werden.

Beispiel:
Aufruf: `Int2Asc(-123, s); (*s als ARRAY OF CHAR *)`
danach: `Write.String(s);` liefert: -123

Hinweis: Sie können die Prozedur aus Aufgabe a) verwenden (auch wenn Sie diese möglicherweise nicht gelöst haben)!

Lösung:

```
PROCEDURE Int2Asc(z: INTEGER; VAR f: ARRAY OF CHAR);
  VAR i: INTEGER;

BEGIN
  i := 0;
  IF (z >= 0) THEN
    f[i] := "+";
  ELSE
    f[i] := "-"; z := -z;
  END;

  i := NumberOfDigits(z);
  f[i+1] := 0X;
  WHILE i > 0 DO
     f[i] := CHR(ORD("0")+ z MOD 10);
     z := z DIV 10;
     i := i-1;
  END;
END Int2Asc;
```

Aufgabe 5 **(14 Punkte)**

Schreiben Sie ein vollständiges Oberon-Programm, das einen Text von der Standardeingabe einliest und die enthaltenen Zeichen wie folgt an die Standardausgabe ausgibt:

- ein Kleinbuchstabe mit **Sn**, wobei **n** die Position des Buchstaben im Alphabet ist, die Ausgabe von n soll immer mit genau zwei Stellen erfolgen
  Beispiel: “e” wird als S05, “k” als S11 ausgegeben.
- ein Grossbuchstabe mit **Cn**, wobei **n** die gleiche Bedeutung hat wie bei den Kleinbuchstaben
  Beispiel: “D” wird als C04, “M” als C13 ausgegeben.
- eine Ziffer mit **Nn**
  Beispiel: “2” wird als N02, “5” als N05 ausgegeben
- alle anderen Zeichen werden im Format **An** ausgegeben, wobei n die Position des Zeichens in der ASCII-Tabelle ist und mit drei Stellen ausgegeben werden soll.
- Zwischen den einzelnen Ausgaben soll als Trenner stets ein Leerzeichen ausgegeben werden.

Anwendungs-Beispiel:

```
theseus$ cat text
Heute ist der 12.04.2002.
theseus$ NumCode < text
C08 S05 S21 S20 S05 A032 S09 S19 S20 A032 S04 S05 S18
A032 N01 N02 A046 N00 N04 A046 N02 N00 N00 N02 A046 A010
```

Lösung bitte auf die nächste Seite

Lösung von Aufgabe 5:

```
MODULE NumCode;
 IMPORT Read, Streams, Print;
  VAR ch : CHAR;
BEGIN
  Read.Char(ch);
  WHILE ~Streams.stdin.eof DO
  CASE ch OF
    | "a" .. "z" : Print.F1("S%02d ",ORD(ch)-ORD("a")+1);
    | "A" .. "Z" : Print.F1("C%02d ",ORD(ch)-ORD("A")+1);
    | "0" .. "9" : Print.F1("N%02d ",ORD(ch)-ORD("0"));
   ELSE
     Print.F1("A%03d ",ORD(ch));
   END;
  Read.Char(ch);
  END;
  Print.F("\n");
END NumCode.
```

Alternativ: siehe nächste Seite

```
MODULE NumCode;
 IMPORT Read, Streams, Write;
  VAR ch : CHAR; i: INTEGER;
BEGIN
  Read.Char(ch);
  WHILE ~Streams.stdin.eof DO
    CASE ch OF
    | "a" .. "z" :
                  i := ORD(ch)-ORD("a")+1;
                  Write.Char("S");
                  IF i < 10 THEN Write.Char("0") END;
                  Write.Int(i,0);
    | "A" .. "Z" :
                  i := ORD(ch)-ORD("A")+1;
                  Write.Char("C");
                  IF i < 10 THEN Write.Char("0") END;
                  Write.Int(i,0);
   | "0" .. "9" :
                  Write.Char("N"); Write.Char("0"); Write.Char(ch);
   ELSE
                  i := ORD(ch);
                  Write.Char("A");
                  IF i < 10 THEN
                    Write.String("00");
                  ELSIF i < 100 THEN
                    Write.Char("0");
                  ELSE
                  END;
                  Write.Int(i,0);
   END;
   Write.Char(" ");
   Read.Char(ch);
  END;
  Write.Ln;
END NumCode.
```

Aufgabe 6 **(8 Punkte)**

Gegeben ist das folgende Programm:

```
MODULE Obscure;
  IMPORT Write;

  VAR u,v,w: CHAR;

  PROCEDURE Magic(VAR x: CHAR; y,z: CHAR);
  BEGIN
   z:=x; x:=y; y:=z
  END Magic;

BEGIN
  u :="a"; v:="c"; w:="h";
  Magic(u,v,w);
  Write.Char(u); Write.Char(v); Write.Char(w);Write.Ln;
  Magic(v,w,u);
  Write.Char(u); Write.Char(v); Write.Char(w);Write.Ln;
  Magic(w,u,v);
  Write.Char(u); Write.Char(v); Write.Char(w);Write.Ln;
  Magic(u,v,w);
  Write.Char(u); Write.Char(v); Write.Char(w);Write.Ln;
END Obscure.
```

Geben Sie an, was dieses Programm an die Standardausgabe schreibt!
Lösung:

```
cah
cha
ahc
hac
```

Aufgabe 7 **(8 Punkte)**

Gegeben sei folgende Prozedur:

```
PROCEDURE WhichNumber(a,b,c: INTEGER): INTEGER;
BEGIN
   IF (a > 0) OR ((b >= 0) & (c > 0)) THEN
      RETURN 0
   ELSE
      IF (a >= 0) & (c > 0 ) THEN
            RETURN 1
      ELSE
            RETURN 2
      END
   END
END WhichNumber;
```

Wie lautet die Menge aller Paare (a,b,c), für die diese Prozedur den Wert **1** liefert? Bitte inklusive Herleitung angeben!
Lösung:

$\{ (a, b, c) \mid a = 0, b \leq 0, c > 0 \}$

Aufgabe 8 **(6 Punkte)**

In einem Kostenrechnungs-Programm sollen Artikel-Daten, die wie folgt beschrieben sind, verarbeitet werden:

- Artikel-Nummer; Form: `nnnn`, mit `n` Dezimalziffer; Beispiel: `4711`
- Artikel-Bezeichnung, max. 30 Zeichen; Beispiel: `Gewindebohrer M28`
- Verkaufspreis (in €); Beispiele: `23.10`, `20`
- Haltbarkeitsdatum; Form: Tag.Monat.Jahr; Beispiel: `21.2.2003`

Geben Sie hierfür Konstanten- und Typ-Vereinbarungen in Oberon an, so dass Variable vereinbart werden können, die je einen vollständigen Artikel-Datensatz aufnehmen können!

Lösung:

```
CONST ArtikelNamenLaenge = 31;

TYPE
      ArtikelBezeichnung = ARRAY ArtikelNamenLaenge OF CHAR;

      Datum = RECORD
                tag, monat, jahr: INTEGER;
              END;

      ArtikelDaten = RECORD
                       nummer: INTEGER;
                       bezeichnung: ArtikelBezeichnung
                       verkaufsPreis: REAL;
                       haltbarkeitsDatum: Datum;
                     END;
VAR artikel: ArtikelDaten;
```

Aufgabe 9 **(10 Punkte)**

Sie legen einen Betrag von K € zu einem Zinsatz von p % für n Jahre an. Die jährlich anfallenden Zinsen werden jeweils dem Anlagebetrag zugeschlagen. Schreiben Sie eine Oberon-Prozedur, die basierend auf den genannten Größen den Betrag nach n Jahren liefert!

Lösung:

```
PROCEDURE Anlage(K : REAL; p, n : INTEGER) : REAL;
  VAR i : INTEGER;
 BEGIN
 i:=0;
 WHILE (i < n) DO
   K:=K*(1+p/100);
   INC(i);
 END;
 RETURN K;
 END Anlage;
```