

## Selbsttest 1

### 1. Wissensfragen – Algorithmus und Programm

- 1.1 Ein Algorithmus ist ein schrittweises, präzises Verfahren zur Lösung eines Problems.
- 1.2 Ein Algorithmus muss weiters endlich und ausführbar sein.
- 1.3 Ein Programm ist die Beschreibung eines Algorithmus in einer Programmiersprache.
- 1.4 Ein Programm besteht aus Daten und Befehlen.

### 2. Wissensfragen – Variablen und Anweisungen

- 2.1 Variablen sind Behälter für Werte.
- 2.2 Variablen haben einen Namen und einen Datentyp.
- 2.3 Der Datentyp beschränkt die Menge erlaubter Werte.
- 2.4 Die drei grundlegenden Befehlsarten sind Sequenz, Selektion und Iteration.

### 3. Wissensfragen – Beschreibung von Programmiersprachen

- 3.1 Die Syntax gibt Regeln, nach denen Sätze gebaut werden dürfen.
- 3.2 Die Semantik definiert die Bedeutung der Sätze.
- 3.3 Eine Grammatik ist eine Menge von Syntaxregeln.
- 3.4 EBNF bedeutet Erweiterte Backus-Naur-Form.

### 4. Wissensfragen – Metazeichen der EBNF

- 4.1 Das „=-“-Zeichen trennt Regelseiten.
- 4.2 Das „|“-Zeichen trennt Alternativen.
- 4.3 Die eckigen Klammern umgeben wahlweise vorkommende Teile.
- 4.4 Die geschwungenen Klammern umgeben 0..n-malig vorkommende Teile.

### 5. Beispiel zu EBNF: Datumsformate

Um verschiedene Datumsformate darzustellen, wurde folgende Beschreibung in Prosa gewählt:  
Es gibt zwei Arten wie ein Datum angezeigt werden kann: als Kurzfassung oder als ausführliche Version:

- Die Kurzfassung enthält zuerst den Tag in numerischer Form, dann den Monat (ebenfalls numerisch) und zuletzt das Jahr, wobei beim Jahr nur die beiden letzten Stellen angezeigt werden. Diese Daten werden von einem Zeichen getrennt, das entweder ein Punkt oder ein Schrägstrich sein kann. Gemischte Verwendung von Trennzeichen ist nicht erlaubt.

- Die ausführliche Darstellung des Datums enthält zuerst die deutsche Bezeichnung des Wochentags (gefolgt von einem Beistrich) und des Monats gefolgt vom Tag (numerisch). Durch einen Beistrich getrennt folgt das Jahr.

Bei beiden Darstellungen können beliebig viele Leerzeichen ( ' ') zwischen den einzelnen Werten stehen.

Beispiel:

- kurz: 15. 4.97 oder 3/2/78
- lang: Montag, Januar 10, 1997

Geben Sie die EBNF für solche Datumsdarstellungen an.

```

Date = ShortDate | LongDate.
ShortDate = space (number space '.' space MonthNumber space '.') |
            (number space '/' space MonthNumber space '/') space digit digit.
LongDate = space Day space ',' space Month space number space ',' space
            space number.
MonthNumber = "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9" | "10" |
             "11" | "12".
Day = "Montag" | "Dienstag" | "Mittwoch" | "Donnerstag" | "Freitag" |
      "Samstag" | "Sonntag".
Month = "Januar" | "Februar" | "März" | "April" | "Mai" | "Juni" | "Juli" |
        "August" | "September" | "Oktober" | "November" | "Dezember".
space = { " " }.

```

## 6. Beispiel zu Algorithmen in Prosa

Schreiben Sie einen Algorithmus in Prosa, welcher als Eingabe vier ganze positive Zahlen  $n_1$ ,  $n_2$ ,  $n_3$ ,  $n_4$  erwartet und diese aufsteigend sortiert wieder ausgibt. Beachten Sie, dass auch Zahlen mit gleichem Wert vorkommen können. Demonstrieren Sie Ihren Algorithmus anhand von zwei Schreibtischtests.

```

Lies 4 Zahlen n1, n2, n3, n4 ein.
Wenn n1 größer als n2 ist, dann vertausche die beiden Zahlen.
Wenn n3 größer als n4 ist, dann vertausche die beiden Zahlen.
Wenn n2 größer als n3 ist, dann vertausche die beiden Zahlen.
Wiederhole die vorherigen 3 Schritte.
Gib die Zahlen der Reihe nach aus.

```

## 7. Grafische Darstellung von Algorithmen: Maximale Spanne

Schreiben Sie einen Algorithmus zur Berechnung der maximalen Spanne (Differenz zwischen Minimum und Maximum) einer Reihe von eingelesenen Zahlen. Zahlen werden mit `read(number)` eingelesen. '-1' gibt an, daß die Reihe abgeschlossen ist und soll nicht zur Berechnung verwendet werden. Falls die Zahlenreihe leer ist, so soll als maximale Spanne 0 ausgegeben werden. Geben Sie den Algorithmus sowohl als Ablaufdiagramm als auch als Struktogramm an.

