

Übung01: Algorithmen, EBNF

Abgabetermin: 15.03.19, 15:30 Uhr

Name: _____ Matrikelnummer: _____

| Aufgabe | schriftlich abzugeben | elektronisch abzugeben | gelöst |
|-----------|--------------------------------------|------------------------|--------------------------|
| Aufgabe01 | Ablaufdiagramm, Schreibtischtests | | <input type="checkbox"/> |
| Aufgabe02 | Ablaufdiagramm, Schreibtischtests | | <input type="checkbox"/> |
| Aufgabe03 | Ablaufdiagramm, Schreibtischtests | | <input type="checkbox"/> |
| Aufgabe04 | EBNF Grammatikregeln | | <input type="checkbox"/> |
| Aufgabe05 | EBNF Grammatikregeln | | <input type="checkbox"/> |

Achtung!

Bitte auf diesem Deckblatt:

- **Name** und **Matrikelnummer** ausfüllen,
- gelöste Aufgaben **ankreuzen**

und dann der schriftlichen Abgabe als erste Seite anheften.

Algorithmen (12 Punkte)

In diesen Aufgaben sollen Algorithmen für beliebig lange Folgen von Zahlen entworfen, als Ablaufdiagramme spezifiziert und mit Schreibtischtests überprüft werden.

Aufgabe01: Anzahl der Vorkommisse (4 Punkte)

Der Algorithmus `zähleVorkommen(↓zahlen, ↓z, ↑anzahl)` bestimmt für eine gegebene Zahl `z`, wie oft sie in der Folge vorkommt. Ist die Zahlenfolge leer, soll eine Anzahl von 0 zurückgegeben werden.

Aufgabe02: Durchschnitt Berechnen (4 Punkte)

Der Algorithmus `berechneDurchschnitt(↓zahlen, ↑durchschnitt)` berechnet den Durchschnitt der gegebenen Zahlenfolge. Ist die Zahlenfolge leer, soll ein Durchschnitt von 0 zurückgegeben werden. Beispiel: 3, 6, 1, 3, 7, 4 → Durchschnitt: 3,57.

Aufgabe03: Paar Finden (4 Punkte)

Der Algorithmus `findePaar(↓zahlen, ↓z1, ↓z2, ↑kommtVor)` prüft, ob in der Zahlenfolge die gegebenen Zahlen `z1` und `z2` als Paar vorkommen, d.h. ob auf die Zahl `z1` direkt `z2` folgt. Je nach Fall soll der Wert der Variable `kommtVor` auf `wahr` bzw. `falsch` gesetzt werden. Ist die Zahlenfolge leer, soll der Wert ebenfalls `falsch` sein. Beispiel: 3, 6, 1, 3, 7, 4 → (`z1: 1, z2: 3, kommtVor: wahr`), (`z1: 3, z2: 4, kommtVor: falsch`).

Hinweis

Achten Sie darauf, dass die Algorithmen auch bei oben genannten Grenz- und Sonderfällen funktionieren. Überprüfen Sie diese Fälle ebenfalls mit Schreibtischtests. Verwenden Sie zur Formulierung der Algorithmen umgangssprachliche Ausdrücke wie:

- „`x` ← erste Zahl“, „`x` ← nächste Zahl“
- „solange Folge nicht zu Ende“, „solange eine nächste Zahl existiert“
- „Länge der Folge“

Sie können alle allgemein bekannten mathematischen Operatoren und Funktionen wie arithmetische Operationen, Vergleichsoperationen oder Mengenoperationen verwenden.

Abzugeben:

- Ablaufdiagramme (Handzeichnungen genügen)
- Schreibtischtests (2-3 pro Algorithmus)

EBNF (12 Punkte)

In dieser Aufgabe sollen Grammatiken für die Verarbeitung von gegebenen Datenstrukturen mithilfe von EBNF Regeln spezifiziert werden.

Aufgabe04: Datum und Uhrzeit (6 Punkte)

Ein Datum beginnt mit einem vierstelligen Jahr. Danach kommt ein Monatskürzel aus drei Buchstaben, die in einer fixen Aufzählung definiert sind (Jan, Feb, ... Nov, Dec). Weiters folgt der zweistellige Tag des Monats. Jahr, Monat und Tag sind jeweils von einem Leerzeichen getrennt. Nach dem Datum kommen Beistrich und Leerzeichen, gefolgt von der zweistelligen Stunde. Nach einem Doppelpunkt folgt die zweistellige Minute. Optional kann auf die Minute, durch einen Doppelpunkt eingeleitet, die zweistellige Sekunde folgen. Nach der Minute oder Sekunde kann optional, eingeleitet durch einen Beistrich und ein Leerzeichen, entweder am oder pm als Bezeichnung folgen. *Ihre Grammatik muss nicht prüfen, ob z.B. der Tag oder die Uhrzeit korrekte Werte aufweisen.*

Beispiele für gültige Eingaben:

- 2016 Oct 16, 16:17
- 2012 Dec 31, 23:59:59
- 1999 Feb 28, 06:20, pm
- 2041 Jun 07, 08:06:04, am

Aufgabe05: E-Mail Adressen (6 Punkte)

Eine E-Mail Adresse beginnt mit einem Namen. Auf diesen Namen kann optional, getrennt durch einen Punkt, ein weiterer Name folgen. Danach kommt ein @-Symbol. Abschließend folgt eine komplexe Domainbezeichnung, die am Anfang aus einem Namen bestehen muss. In der Mitte kann die Bezeichnung aus beliebig vielen Namen bestehen, die jeweils durch Punkte voneinander getrennt sind. Die Bezeichnung schließt ab mit einem Punkt, gefolgt von der Endung at, com oder net.

Beispiele für gültige Eingaben:

- webmaster@mydomain.net
- max.mustermann@example.com
- noreply@ssw.unilinz.ac.at

Hinweis

Strukturieren Sie Ihre EBNF Grammatik mithilfe mehrerer Regeln. Verwenden Sie für Zahlen und Namen die Regeln aus den Vorlesungsfolien.

Abzugeben:

- EBNF Grammatikregeln