

**Einführung in die Programmierung**  
WS 2009/10

**Übungsblatt 2: Zahlendarstellung, Syntaxdefinition**

Besprechung: 09./11./12.11.2009

Ende der Abgabefrist: Montag, 09.11.2009 10:00 Uhr.

**Hinweise zur Abgabe:**

Geben Sie bitte Ihre gesammelten Lösungen zu diesem Übungsblatt in einer Datei `loesung02.zip` unter <http://www.pst.ifi.lmu.de/uniworx/> ab.

**Wichtig:** Achten Sie bitte darauf, dass Ihre Lösungsdateien die korrekten, d. h. die in der Angabe geforderten Namen haben, sonst kann Ihre Lösung nicht der richtigen Aufgabe zugeordnet werden.

**Aufgabe 2-1**     *Darstellung negativer Zahlen und Mengen von Zahlen*

**10 Punkte**

In der Vorlesung haben wir die  $p$ -adische Darstellung von natürlichen Zahlen kennengelernt. Wenn man eine bestimmte Anzahl  $n$  von Stellen annimmt, lässt sich eine schematische Darstellung von ganzen Zahlen (die Sie formeller erst später kennenlernen) im Binärsystem (d.h.  $p = 2$ ) wie folgt veranschaulichen:

$$-2^{n-1} \cdot a_{n-1} + 2^{n-2} \cdot a_{n-2} + \dots + 2^1 \cdot a_1 + 2^0 \cdot a_0$$

wobei  $a_{n-1}a_{n-2} \dots a_0$  eine Zeichenreihe aus dem Alphabet  $\{0, 1\}$  ist. Hier wird also der erste Summand negiert. Dadurch sind mit einer Zeichenkette der Länge  $n$  nicht die Zahlen von 0 bis  $p^n$  darstellbar, sondern die Zahlen von  $-p^{n-1}$  bis  $p^{n-1} - 1$ . Für  $n = 8$  wäre somit die kleinste darstellbare Zahl  $-128$  mit der Binärdarstellung `10000000`, die größte Zahl  $127$  mit der Binärdarstellung `01111111`.

Betrachten wir nun folgende Mengen  $B \subseteq A \subseteq \mathbb{Z}$  von Zahlen:

$$A = \{-8, -7, -6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6, 7\}$$

$$B = \{-4, -3, -2, -1, 0, 1, 2, 3\}$$

Sei eine Funktion  $f : A \rightarrow B$  definiert wie folgt:

$f(-8) = 0$	$f(0) = 0$
$f(-7) = 1$	$f(1) = 1$
$f(-6) = 2$	$f(2) = 2$
$f(-5) = 3$	$f(3) = 3$
$f(-4) = -4$	$f(4) = -4$
$f(-3) = -3$	$f(5) = -3$
$f(-2) = -2$	$f(6) = -2$
$f(-1) = -1$	$f(7) = -1$

Welcher tiefere Sinn steckt wohl hinter dieser Funktion? Erklären Sie in einer Datei `darstellung.txt` die Bedeutung der Funktion.

**Aufgabe 2-2**      *Syntaxdefinition: BNF-Satzform für arithmetische Ausdrücke*

**10 Punkte**

Ein arithmetischer Ausdruck entsteht durch Verknüpfung von Zahlen mit arithmetischen Operatoren. Formal können arithmetische Ausdrücke folgendermaßen definiert werden:

- Jede Zahl ist ein arithmetischer Ausdruck.
  - Wenn  $A$  ein arithmetischer Ausdruck ist, dann ist auch  $(A)$  ein arithmetischer Ausdruck, d.h. ein korrekt geklammerter arithmetischer Ausdruck ist wiederum ein arithmetischer Ausdruck.
  - Wenn  $A_1$  und  $A_2$  arithmetische Ausdrücke sind, dann sind auch  $A_1 + A_2$ ,  $A_1 - A_2$ ,  $A_1 * A_2$  sowie  $A_1 \div A_2$  arithmetische Ausdrücke, d.h. auch die Summe, die Differenz, das Produkt und der Quotient von arithmetischen Ausdrücken sind arithmetische Ausdrücke.
- (a) Modellieren Sie in BNF-Satzform die syntaktische Variable  $\langle \text{Ausdruck} \rangle$ , welche alle Arten von arithmetischen Ausdrücken akzeptiert. Diese Variable ist also das sogenannte *Startsymbol*. Gerne dürfen Sie weitere Hilfs-Variablen einführen.
- (b) Geben Sie für den arithmetischen Ausdruck  $((3 + 5) * 8) + 12$  die Ableitung an. Halten Sie sich dabei exakt an die von Ihnen definierten Syntaxregeln. Wenden Sie in jedem Ableitungsschritt nur genau eine Regel an.

Geben Sie die Lösung beider Teilaufgaben in einer Datei `ausdruck.txt` ab.

**Aufgabe 2-3**     *Semantik der BNF***10 Punkte**

Folgende BNF-Satzform definiert die Syntax von Klausuren für die Vorlesung *Einführung in die Programmierung*. Dabei steht STRING für eine beliebige Zeichenkette, die hier nicht weiter definiert werden soll. Sie können diese Datenstruktur als gegeben betrachten.

$\langle EIP\_Klausur \rangle ::= (\langle Aufgabe \rangle)^+$   
 $\langle Aufgabe \rangle ::= \langle Nummer \rangle \langle Thema \rangle \langle Punkte \rangle \langle Inhalt \rangle$   
 $\langle Nummer \rangle ::= \langle Nichtnullziffer \rangle$   
 $\langle Thema \rangle ::= \text{Daten und Algorithmen} \mid \text{Mathematische Grundlagen} \mid$   
                   $\text{Imperative Programmierung} \mid \text{Objektorientierte Programmierung} \mid$   
                   $\text{Objektorientierter Entwurf} \mid \text{Datenstrukturen und Algorithmen}$   
 $\langle Punkte \rangle ::= \langle Nichtnullziffer \rangle (\langle Ziffer \rangle)^*$   
 $\langle Inhalt \rangle ::= \langle String \rangle$   
 $\langle Ziffer \rangle ::= 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$   
 $\langle Nichtnullziffer \rangle ::= 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$

(a) Entscheiden Sie für die folgenden drei Klausuren, ob es sich um gültige Klausuren im Sinne der oben definierten Syntax handelt. Falls es sich um eine korrekte Klausur handelt, skizzieren Sie deren Ableitung (d.h. Sie dürfen hier auch einzelne Schritte zusammenfassen. Achten Sie aber darauf, dass der logische Aufbau noch erkennbar ist). Falls es sich um keine korrekte Klausur handelt, begründen Sie Ihre Antwort.

- 1 Objektorientierter Entwurf 12 Inhalt dieser Aufgabe  
  2 Daten und Algorithmen     7 Inhalt dieser Aufgabe  
  3 Daten und Algorithmen     9 Inhalt dieser Aufgabe
- 4 Datenstrukturen und Algorithmen 17 Inhalt dieser Aufgabe  
  9 Daten und Algorithmen            0 Inhalt dieser Aufgabe
- 8 Inhalt dieser Aufgabe 8 Mathematische Grundlagen

(b) Geben Sie nun selbst zwei (zusätzliche) Klausuren an, die aus der oben definierten Satzform ableitbar sind. Die Ableitungen dieser Klausuren sind *nicht* verlangt.

Geben Sie die Lösung beider Teilaufgaben in einer Datei `semantik.txt` ab.