

# Hausaufgabe 4 (I) (6 Punkte)

Abgabe: 1. Juli 2012

## Aufgabe

Schreiben Sie ein Programm, das

1. einen ungeklammerten mathematischen Ausdruck unter **Beachtung der Vorrangregeln** in einen **vollständigen dyadischen Ausdruck** umwandelt und das Ergebnis auf dem Bildschirm ausgibt!
2. aus dem vollständig geklammerten dyadischen Ausdruck von 1. einen **Ausdrucksbaum** generiert
3. den Ausdrucksbaum in **Postorder** traversiert und das Ergebnis auf dem Bildschirm ausgibt!
4. den Baum durch Ändern und Löschen von Knoten weitestgehend vereinfacht, indem z.B. rein numerische Operationen berechnet oder triviale Operationen ersetzt werden. Beschreiben Sie die Art aller von Ihnen implementierten Ersetzungen in einem PDF-Dokument, z.B.  $a*1 \rightarrow a$
5. den (vereinfachten) Baum in **Präorder** traversiert und das Ergebnis auf dem Bildschirm ausgibt!
6. den (vereinfachten) Baum als **vollständig geklammerten dyadischen Ausdruck** ausgibt!

# Hausaufgabe 4 (2) (6 Punkte)

## Abgabe: 1. Juli 2012

### Hinweise

1. Der mathematische Ausdruck ist als **Kommandozeilenparameter** zu übergeben. Sie können davon ausgehen, dass der mathematische Ausdruck syntaktisch korrekt ist.
2. Operanden sind **ganzzahlige Zahlen** im Intervall  $[-999, 999]$  oder **Kleinbuchstaben**.
3. Als Operatoren sind nur Addition(+), Subtraktion(−) und Multiplikation(\*) zu implementieren.
4. Ein vollständig geklammerter dyadischer Ausdruck (DA) sei wie folgt definiert:

**DA** ::= ( Operand Operator Operand )

**Operator** ::= + | − | \*

**Operand** ::= DA | Zahl | Kleinbuchstabe

5. Gleichrangige Operatoren sind von rechts nach links zu klammern, z.B.  $-1+b-1+3 \rightarrow (-1+(b-(1+3)))$
6. Sie dürfen **ausschließlich für die Bildschirmausgabe Bibliotheksfunktionalität** verwenden!
7. Fügen Sie bei der Prä- bzw. Postorderausgabe zwischen Operanden und Operatoren immer ein Leerzeichen ein, damit Vorzeichen von Operatoren von Operanden unterschieden werden können!
8. Ihr Programm muss sich mit **GCC** übersetzen lassen!

# Hausaufgabe 4 (3) (6 Punkte)

Abgabe: 1. Juli 2012

## Beispiel

Programmname: expression

Programmaufruf: expression "2\*3-a+6-b\*0"

Ausgabe des vollständig geklammerten dyadischen Ausdrucks:

$((2*3)-(a+(6-(b*0))))$

Ausgabe des Ausdrucksbaums in Postorder:

2 3 \* a 6 b 0 \* - + -

Ausgabe des optimierten Ausdrucksbaums in Präorder:

- 6 + a 6

Ausgabe des optimierten Ausdrucksbaums geklammert:

$(6-(a+6))$

Ausdrucksbaum der Eingabe

