Aufgaben-Blatt: Symbolisches Differenzieren

Das Problem des *symbolischen Differenzierens* besteht darin, einen gegebenen arithmetischen Ausdruck E *symbolisch* nach einer Variable x abzuleiten. *Symbolisch* heißt in diesem Zusammenhang, dass das Ergebnis dieser Operation keine Zahl ist, sondern wieder ein arithmetischer Ausdruck. Hat beispielsweise E den Wert $x \cdot \exp(x)$, so gilt nach der Produkt-Regel der Differential-Rechnung

$$\frac{d}{dx}\Big(x \cdot \exp(x)\Big) = 1 \cdot \exp(x) + x \cdot \exp(x).$$

Also ist das Ergebnis, das wir erhalten, wenn wir den arithmetischen Ausdruck $x \cdot exp(x)$ nach x differenzieren, der Ausdruck $1 \cdot \exp(x) + x \cdot \exp(x)$.

Um einen Algorithmus entwickeln zu können, der arithmetische Ausdrücke symbolisch differenziert, definieren wir zunächst induktiv die Menge $\mathcal{E}$ der arithmetischen Ausdrücke.

1. Die Variablen x, y und z sind arithmetische Ausdrücke.
2. Alle Zahlen sind arithmetische Ausdrücke.
3. Gilt $s, t \in \mathcal{E}$, so gilt auch:
   (a) $s + t \in \mathcal{E}$,
   (b) $s - t \in \mathcal{E}$,
   (c) $s \cdot t \in \mathcal{E}$,
   (d) $s \;/\; t \in \mathcal{E}$.
4. Ist $s \in \mathcal{E}$ und ist $n \in \mathbb{Z}$ so gilt auch $s^n \in \mathcal{E}$.

Als nächstes müssen wir überlegen, wie arithmetische Ausdrücke in *SetlX* repräsentiert werden können. Wir definieren dazu eine Repräsentations-Funktion

$\texttt{rep} : \mathcal{E} \to \textit{SetlX}$

die als Eingabe einen arithmetischen Ausdruck nimmt und diesen in eine *SetlX*-Datenstruktur transformiert:

1. Variablen werden als Strings dargestellt. Daher gilt
   $\texttt{rep}(v) = v \quad$ für alle Variablen v.
2. Genauso ist die Repräsentation von Zahlen trivial:
   $\texttt{rep}(x) = x \quad$ für alle $x \in \mathbb{N}$.
3. $\texttt{rep}(s + t) := [\texttt{rep}(s), \text{“}+\text{”}, \texttt{rep}(t)]$.
4. $\texttt{rep}(s - t) := [\texttt{rep}(s), \text{“}-\text{”}, \texttt{rep}(t)]$.
5. $\texttt{rep}(s \cdot t) := [\texttt{rep}(s), \text{“}*\text{”}, \texttt{rep}(t)]$.
6. $\texttt{rep}(s \;/\; t) := [\texttt{rep}(s), \text{“}/\text{”}, \texttt{rep}(t)]$.
7. $\texttt{rep}(s^n) := [\texttt{rep}(s), \text{“}**\text{”}, n]$.

Aufgabe 1: Schreiben Sie eine *SetlX*-Prozedur `diff`, so dass der Aufruf $\texttt{diff}(E, x)$ den arithmetischen Ausdruck E symbolisch nach der Variablen x differenziert.

Hinweis: Unter

`www.dhbw-stuttgart.de/~stroetma/Logic/SetlX-Programs/derivative-frame.stlx`

finden Sie ein Programm-Gerüst, in dem Sie nur noch die Prozedur `diff()` implementieren müssen. Das Gerüst enthält bereits einen Parser, einen Pretty-Printer und diverse Testfälle. Die Ableitungs-Regeln sind wir folgt:

1. Summen-Regel: $\frac{d}{dx}\big(g+h\big) = \frac{dg}{dx} + \frac{dh}{dx}$
2. Differenzen-Regel: $\frac{d}{dx}\big(g-h\big) = \frac{dg}{dx} - \frac{dh}{dx}$
3. Produkt-Regel: $\frac{d}{dx}\big(g\cdot h\big) = \frac{dg}{dx}\cdot h + g\cdot\frac{dh}{dx}$
4. Quotienten-Regel: $\frac{d}{dx}\Big(\frac{g}{h}\Big) = \frac{\frac{dg}{dx}\cdot h - g\cdot\frac{dh}{dx}}{h\cdot h}$
5. Potenz-Regel: $\frac{d}{dx}g^n = n\cdot g^{n-1}\cdot\frac{dg}{dx}$ für alle $n\in\mathbb{Z}$

Hinweis: Es gibt eine vordefinierte Prozedur `isInteger`() die genau dann "`true`" liefert, wenn ihr Argument eine ganze Zahl ist.

Aufgabe 2: Erweitern Sie das Programm so, dass auch arithmetische Ausdrücke, die Funktionen wie exp(), `ln`(), sqrt(), sin(), cos(), tan(), oder arctan() enthalten, differenziert werden können.

Hinweis: Zur Ableitung dieser Funktionen gelten die folgenden Regeln:

$f(x)$	$\frac{d}{dx}f$
$\texttt{exp}(x)$	$\texttt{exp}(x)$
$\texttt{ln}(x)$	$\frac{1}{x}$
$\texttt{sin}(x)$	$\texttt{cos}(x)$
$\texttt{cos}(x)$	$-\texttt{sin}(x)$

$f(x)$	$\frac{d}{dx}f$
$\texttt{tan}(x)$	$\frac{1}{\texttt{cos}^2(x)}$
$\texttt{arctan}(x)$	$\frac{1}{1+x^2}$
$\sqrt{x}$	$\frac{1}{2\sqrt{x}}$

Hinweis: Berücksichtigen Sie die Kettenregel. Ist die Funktion $h(x)$ als

$$h(x) = g\big(f(x)\big)$$

definiert, so läßt sich die Ableitung von $h(x)$ nach der Formel

$$\frac{d}{dx}h(x) = g'\big(f(x)\big)\cdot f'(x) \quad \text{mit } g'(x) = \frac{d}{dx}g(x) \text{ und } f'(x) = \frac{d}{dx}f(x)$$

berechnen.

Aufgabe 3: Schreiben Sie eine Prozedur `simplify`, die einen gegebenen arithmetischen Ausdruck unter Berücksichtigung der Regeln

$$1\cdot x = x\cdot 1 = x, \quad 0\cdot x = x\cdot 0 = 0, \quad 0+x = x+0 = x$$

vereinfacht.

Hinweis: Je rekursiver Sie Ihr Programm schreiben, desto einfacher wird es!

Aufgabe 4*: Erweitern Sie das Programm so, dass auch arithmetische Ausdrücke der Form $s ** t$ für beliebige arithmetische Ausdrücke s und t differenziert werden können. Testen Sie die Implementierung, indem Sie

$$\frac{d}{dx}\big(x^x\big)$$

berechnen.