Haskell Unit 4: ZF-expressions

Antoni Diller
26 July 2011

Introduction

A ZF-expression has the following form:

\[
[ \ x * x \mid x \leftarrow [1, 2, 7, 12], \text{even } x \ ]
\]

This is a very convenient way of defining a list. The above ZF-expression is the list of all those things of the form \( x * x \) such that \( x \) is drawn from the list \([1, 2, 7, 12]\) and \( x \) is even. The above ZF-expression evaluates to \([4, 144]\).

The general format of a ZF-expression is:

\[
[ \ \text{EXP} \mid \text{QUAL, \ldots, QUAL} \ ]
\]

where QUAL is either a Boolean-valued expression or a generator. A generator is something of the form:

\[
\text{VARIABLE} \leftarrow \text{LIST}
\]
\[
\text{PATTERN} \leftarrow \text{LIST}
\]

Reduction rules for ZF-expressions

(ZF1) \( [ \ \text{e} \mid \text{v} \leftarrow [] , \text{q} \ ] \) reduces to \([ \ ]\), where \( q \) is zero or more qualifiers.

(ZF2) \( [ \ \text{e} \mid \text{v} \leftarrow f:fs , \text{q} \ ] \) reduces to \([ \ \text{e} \mid \text{q} \ ] [ \ v := f \ ] ++ [ \ \text{e} \mid \text{v} \leftarrow fs , \text{q} \ ] \), where \( h [ \ v := f \ ] \) represents \( h \) with all occurrences of \( v \) in it replaced by \( f \).

(ZF3) \( [ \ \text{e} \mid \text{False} , \text{q} \ ] \) reduces to \([ \ ]\).

(ZF4) \( [ \ \text{e} \mid \text{True} , \text{q} \ ] \) reduces to \([ \ \text{e} \mid \text{q} \ ]\).

(ZF5) \( [ \ \text{e} \ ] \) reduces to \([ \ \text{e} \ ]\).

1
Quicksort

ZF-expressions allow a very concise definition of Hoare’s Quicksort algorithm:

\[
\begin{align*}
\text{quick}~[\ ] &= [] \\
\text{quick}~[x] &= [x] \\
\text{quick}~(x:xs) &= \text{quick}~[ u \mid u \leftarrow xs, u < x ] ++ [ x ] ++ \\
&\quad \text{quick}~[ u \mid u \leftarrow xs, u \geq x ]
\end{align*}
\]