Program Transformation and Analysis

Assignment 3

Benjamin Brandt Ohrt, zpn492

May 8, 2019

1 Introduction

This the third of five weekly assignment in the course Program Transformation and Analysis (PAT) at Copenhagen University. The course professor is Robert Glück. The course is held in block 4, 2019.

In this assignment the focus is on the elimination of intermediate trees and the deforestation algorithm of Philip Wadler.

2 Assignment

Exercise 1

Show the transformation of function composition append (append xs ys) zs into the treeless program in Fig. 3 using the Deforestation Algorithm defined in Fig. 4. Present your transformation in the style of Fig. 5. Explain the steps and discuss shortly in what sense the composition is optimized. [1]

Let append be a function described in section 3

Rule	T[[append (append xs ys) zs]] where f xs ys zs	
(6)	f Nil ys zs	= T[[append (append Nil ys) zs]]
(5)		= T[[append ys zs]]
		= append ys zs
(6)	f (Cons x xs) ys zs	= T[[append (append (Cons x xs) ys) zs]]
(5)		= T[[append ((Cons x) (append xs ys)) zs]]
(5)		= T[[append (append xs ys) (Cons x zs)]]
		= f xs ys (Cons x zs)
		append (append xs ys) zs transforms to:
	f xs ys zs	= case xs of
		Nil : append ys zs
		Cons x xs : Cons x (f xs ys zs)

Exercise 2

Apply the algorithm described in exercise 1 to append xs (append ys zs). What is the difference to the result in exercise 1?

For the transformation of append xs (append ys zs) i use the Deforestation algorithm described in [1].

Rule			
			T[[append xs (append ys zs)]]
(3)	T[[case xs of		
		Nil	= (append ys zs)
		Cons x xs	= cons x (append xs (append ys zs))]]
(4)	case xs of		
		Nil	= T[[append ys zs]]
		Cons x xs	= T[[Cons x (append xs (append ys zs))]]
(2)	case xs of		
		Nil	= T[[append ys zs]]
		Cons x xs	= Cons x T[[(append xs (append ys zs))]]
(dec)		h1 ys zs	= T[[append ys zs]]
(dec)		h0 xs ys zs	= T[[append xs (h1 ys zs)]]
(sub)	case xs of		
		Nil	= h1 ys zs
		Cons x xs	= Cons x (h0 xs ys zs)
			append xs (append ys zs) transforms to:
	h0 xs ys zs =		
	case xs of		
		Nil	= (h1 ys zs)
		Cons x xs	= Cons x (h0 xs ys zs)
	h1 ys zs =		
		case ys of	
		Nil	= zs
		Cons y ys	= Cons y (h1 ys zs)

3 Append

```
Append          : list a -> list a -> list a
Append xs ys =
  case xs of
    Nil         : ys
    Cons x xs   : Cons x (append xs ys)
```

References

[1] Wadler P., *Deforestation: transforming programs to eliminate trees. Theoretical Computer Science, 73(2): 231-248, 1990.* https://doi.org/10.1016/0304-3975(90)90147-A