p 134 11
AD

Paper 13 Introduction to Functional Programming 2000

1.
```
fun elems Empty           = []
  | elems (Leaf x)        = [x]
  | elems (Branch(t1,t2)) = elems(t1)@elems(t2);
```

2.
```
fun elemsi (Empty, l)         = l
  | elemsi (Leaf x, l)        = x::l
  | elemsi (Branch(t1,t2),l) = elemsi(t1,elemsi(t2,l));
```

3.
```
fun appendq Nil s          = s
  | appendq Cons(h, tf) s = Cons(h, fn () => appendq(tf(), s));

fun elemsq Empty           = Nil
  | elemsq (Leaf x)        = Cons(x,fn () => Nil)
  | elemsq (Branch(t1,t2)) = appendq(elemsq(t1),elemsq(t2));
```