1:	C	assumption
2:	map⇒#(t1→t2)→[t1]→[t2]	assumption
3:	f⇒#num→num	assumption
4:	f⇒#t1→t2	assumption
5:	xs⇒#[t1]	assumption
6:	(==):[t1]→[t1]→bool	constant
7:	xs:[t1]	C(x)⇒S; S>T
8:	(==)xs:[t1]→bool	F G : T 6,7
9:	nil:[t1]	constant
10:	xs==nil:bool	F G : T 8,9
11:	nil:[t2]	constant
12:	(::):t2→[t2]→[t2]	constant
13:	f:t1→t2	C(x)⇒S; S>T
14:	hd:[t1]→t1	constant
15:	xs:[t1]	C(x)⇒S; S>T
16:	hd xs:t1	F G : T 14,15
17:	f(hd xs):t2	F G : T 13,16
18:	(::)(f(hd xs)):[t2]→[t2]	F G : T 12,17
19:	map:(t1→t2)→[t1]→[t2]	C(x)⇒S; S>T
20:	f:t1→t2	C(x)⇒S; S>T
21:	map f:[t1]→[t2]	F G : T 19,20
22:	tl:[t1]→[t1]	constant
23:	xs:[t1]	C(x)⇒S; S>T
24:	tl xs:[t1]	F G : T 22,23
25:	map f(tl xs):[t2]	F G : T 21,24
26:	f(hd xs)::map f(tl xs):[t2]	F G : T 18,25
27:	if xs==nil then nil else f(hd xs)::map f(tl xs)fi:[t2]	if E then ET else EF fi : T 10,11,26
28:	λxs.if xs==nil then nil else f(hd xs)::map f(tl xs)fi:[t1]→[t2]	λx.E : T1→T2 5-27
29:	λf.λxs.if xs==nil then nil else f(hd xs)::map f(tl xs)fi:(t1→t2)→[t1]→[t2]	λx.E : T1→T2 4-28
30:	map⇒#(t1→t2)→[t1]→[t2]	assumption
31:	f⇒#num→num	assumption
32:	x⇒#num	assumption
33:	(+):num→num→num	constant
34:	x:num	C(x)⇒S; S>T
35:	(+)x:num→num	F G : T 33,34
36:	x:num	C(x)⇒S; S>T
37:	x+x:num	F G : T 35,36
38:	λx.x+x:num→num	λx.E : T1→T2 32-37
39:	(t1→t2)→[t1]→[t2]≺∀(t1,t2).(t1→t2)→[t1]→[t2]	{T≺S}
40:	num→num≺#num→num	generalise
41:	map⇒∀(t1,t2).(t1→t2)→[t1]→[t2]	assumption
42:	f⇒#num→num	assumption
43:	map:(num→num)→[num]→[num]	C(x)⇒S; S>T
44:	f:num→num	C(x)⇒S; S>T
45:	map f:[num]→[num]	F G : T 43,44
46:	(::):num→[num]→[num]	constant
47:	0:num	n:num
48:	(::)0:[num]→[num]	F G : T 46,47
49:	(::):num→[num]→[num]	constant
50:	1:num	n:num
51:	(::)1:[num]→[num]	F G : T 49,50
52:	(::):num→[num]→[num]	constant
53:	2:num	n:num
54:	(::)2:[num]→[num]	F G : T 52,53
55:	nil:[num]	constant
56:	2::nil:[num]	F G : T 54,55
57:	1::2::nil:[num]	F G : T 51,56
58:	0::1::2::nil:[num]	F G : T 48,57
59:	map f(0::1::2::nil):[num]	F G : T 45,58
60:	letrec map=λf.λxs.if xs==nil then nil else f(hd xs):: map f(tl xs)fi,f=λx.x+x in map f(0::1::2::nil)end:[num]	letrecrules'1 2-29,30-38,39,40,41-59