Calculating liveness:

1	(program(v w x y z t.1 t.2)	
2	(movq (int 1) (var v))	
3	(movq (int 46) (var w))	
4	(movq (var v) (var x))	
5	(addq (int 7) (var x))	
6	(movq (var x) (var y))	
7	(addq (int 4) (var y))	
8	(movq (var x) (var z))	
9	(addq (var w) (var z))	
10	(movq (var y) (var t.1))	
11	(negq (var t.1))	
12	(movq (var z) (var t.2))	
13	(addq (var t.1) (var t.2))	
14	(movq (var t.2) (reg rax)))	

Algorithm:

At each instruction, we calculate:

- $L_{before}(k) = (L_{after}(k) - W(k)) \cup R(k)$

Where $W(k)$ are the variables **written** by instruction I_k

And $R(k)$ are the variables **read** by instruction I_k

Traverse the instruction sequence back to front (i.e., backwards in execution order).

Let $I_1, \dots, I_n$ be an instruction sequence.

We write:

- $L_{after}(k)$ for the set of live variables after instruction I_k
- $L_{before}(k)$ for the set of live variables before instruction I_k

The live variables after an instruction are always the same as those before the next instruction:

- $L_{after}(k) = L_{before}(k+1)$

Furthermore:

- $L_{after}(n) = \{\}$