

Input: Function *func* and its control flow blocks

Output: Live and dead variables at each block in function *func*

1: **Variables**

Code *c* at block *blk* in function *func*

def(c): a set of variables defined at code *c*

use(c): a set of variables used at code *c*

in(c): a set of live variables before code *c*

out(c): a set of live variables after code *c*

VARs(blk): a set of all variables used in block *blk*

IN(blk): a set of live variables before *blk*

OUT(blk): a set of live variables after *blk*

LIVE_VARS(blk): a set of live variables after *blk*

DEAD_VARS(blk): a set of dead variables after *blk*

2: **end Variables**

3: *//* Compute live variables at each block of function *func*

4: **procedure** COMPUTE_LIVE_VARS(*func*)

5: **for each** block *blk* other than RETURN in function *func* **do**

6: $OUT(blk) = \emptyset$ *//* Initialise *OUT* set in all blocks

7: **end for**

8: $IN(RETURN) = \{return\ variable\}$

9: $OUT(RETURN) = \{return\ variable\}$

10: **while** Changes to any *IN(blk)* set **do** *//* Repeat until fixed-point

11: **for each** block *blk* other than RETURN in backward order **do**

12: *// OUT* at block *blk* as the union of *IN* in all its child blocks

13: $OUT(blk) = \bigcup_{s \in succ[blk]} IN(s)$

14: *//* Compute live variables from last to first code

15: **for each** $c_i \in \{c_n \dots c_0\}$ at block *blk* **do**

16: **if** $c_i == c_n$ **then**

17: *// OUT(blk)* set is *out* set at last code of block *blk*

18: $out(c_n) = OUT(blk)$

19: **else**

20: *// out* set is *in* set of previous code

21: $out(c_i) = in(c_{i+1})$

22: **end if**

23: *//* Compute liveness transfer equation

24: $in(c_i) = use(c_i) \cup (out(c_i) - def(c_i))$

25: **end for**

26: $IN(blk) = in(c_0)$ *// IN(blk)* is *in* set at first code of block *blk*

27: **end for**

28: **end while**

29: *//* Compute live and dead variables at each block

30: **for each** block *blk* other than RETURN **do**

31: $LIVE_VARs(blk) = OUT(blk)$

32: $DEAD_VARs(blk) = VARs(blk) - OUT(blk)$

33: **end for**

34: **end procedure**