

3-4

assign-homes: $X^* \rightarrow X^0$

```
(program (vs ... ) (x86-inst ... )
  let k = |vs ... |
```

 $vs = (x \ y \ z)$ $k = 3$

count = k if even and k+1 if odd

count = 4

```
[ pushq %rbp    movq rsi,rbp
  subq count,rsi ] setup
```

 $\sigma_{rbp} = [x \mapsto 0$ $\sigma' \dots = \text{map } \text{rename}(\sigma) \text{ is}$ $\sigma \quad y \mapsto 1$ $z \mapsto z]$

```
[ addq count,rsi
  popq rbp ] restore
```

ret

rename (σ , addq x, y) $= \text{addq } -\bar{x} * 8 (\%rbp)$ $\sigma(x) = \bar{x}$ $- \bar{y} * 8 (\%rbp)$ $\sigma(y) = \bar{y}$ patch: $X^0 \rightarrow X$ movq x, y
 \Rightarrow

movq -8(%rbp), -16(%rbp)

 \Rightarrow

movq -8(%rbp), %rax

movq %rax, -16(%rbp)

man: inst $\rightarrow \text{list}(\text{inst})$ mut: inst ($\text{inst} \rightarrow \text{void}$) $\rightarrow \text{void}$

4-1

(program

(program (v w x y z t,1 t,2) | L_{after}

| | | | |
|--------------------|---|-----------------------------|----------|
| (let ([v 1]) | → | (movq (int 1) (var v)) | v |
| (let ([w 46]) | → | (movq (int 46) (var w)) | v, w |
| (let ([x (+ v 7)]) | → | (movq (var v) (var x)) | w, x |
| (let ([y (+ 4 x)]) | → | (addq (int 4) (var x)) | w, x |
| (let ([z (+ x w)]) | → | (movq (var x) (var y)) | w, x, y |
| (+ z (- y)))))) | → | (addq (int 4) (var y)) | w, x, y |
| | → | (movq (var x) (var z)) | w, y, z |
| | → | (addq (var w) (var z)) | y, z |
| | → | (movq (var y) (var t,1)) | t,1, z |
| | → | (negq (var t,1)) | t,1, z |
| | → | (movq (var z) (var t,2)) | t,1, t,2 |
| | → | (addq (var t,1) (var t,2)) | t,2 |
| | → | (movq (var t,2) (reg rax))) | {} |

liveness → when a variable is needed

interference → rel between var $\hat{=}$ the two vars are live at same time

spillage → when a var lives on the stack

2 3

mov 5, a = I₁ Are a and b live at once?

mov 30, b
2

asm = I₁ ... I_n

mov a, c
3

L_{after}(k) = live after I_k

L_{before}(k) = live before I_k

mov 10, b
4

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

add b, c = I₅

L_{after}(n) = \emptyset = L_{before}(1)

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

W = things written

R = things read

u interferes with v = "live a tonce"

Graph $\mathcal{I} = (V, E)$ $V = \text{variables}$

S. a' go in rlx

read c

If I_k is (movg $\leq d$), then for $v \in \text{Latter}(k)$

If I_k is like $(addg \ s \ d)$, then for $v \in \text{Latter}(k)$

add (d, v) to E unless $v = d$

$$\begin{aligned} \{\omega, v\} \\ & (x, w) \\ & (x, w) \\ & (y, w) \\ & (y, w), (y, x) \\ & (z, y), (z, w) \\ & (z, y) \\ & (t_1, z) \\ & (t_1, z) \\ & (t_1 z, t_1) \\ \end{aligned}$$

You must rotate

Figure



