assign-homes : X (vars) ⇒ X (no vars)
   - the stack is <u>vast</u>, but it is slow
   - registers are <u>few</u> and fast
             ↪ (< 16)

```
(let v := 1 in
(let w := 46 in
(let x := v + 7 in
(let y := 4 + x in
(let z := x + w in
  (+ z (-y)))))))
```

7-2

movq	$1, !v	// let v := 1	{v} ↗ ∅
movq	$46, !w	// let w := 46	{v,w}
movq	!v, !x		{x,w}
addq	$7, !x	// let x = v+7	{x,w}
movq	!x, !y		{x,w,y}
addq	$4, !y	// let y= 4+x	{x,w,y} ⎫
movq	!x, !z		{w,y,z}
addq	!w, !z	// let z = x+w	{y,z}
movq	!y, !t		{t,z}
negq	!t	// (- y)	{t,z}
movq	!z, %rax		{z}
addq	!t, %rax	// (+ z t)	∅ ← live-after

live set= active vars

5-3)

Instrs i_0 .... i_n or $\{\% \, rax\}$

$$\text{LiveAfter}(k) = \emptyset \quad \text{if} \quad k = n$$

$$\text{LiveBefore}(k+1) \quad o.w.$$

$$\text{LiveBefore}(k) = (\text{LiveAfter}(k) \setminus W(k)) \cup R(k)$$

$W : \text{inst} \Rightarrow \text{set (vars and reg) we write}$

$R : \text{inst} \Rightarrow \text{set (vars and reg) we } \underline{\text{read}}$

$m : \text{arg} \Rightarrow \text{set (var and reg)}$

$W(\text{popq } a) = M(a)$ $m \, \%r = \{\%r\}$

$R(\text{popq } a) = \emptyset$ $m \, v = \{v\}$

$W(\text{pushq } a) = \emptyset$

$R(\text{pushq } a) = M(a)$

$W(negq\ a) = M(a)$ $R(negq\ a) = M(a)$

$W(addq\ src, dst) = M(dst)$ $R(addq\ src, dst) = M(src) \cup M(dst)$

$W(movq\ src, dst) = M(dst)$ $R(movq\ src, dst) = M(src)$

$movq\ !x, !z$ Initial $= \{w, y, z\}$

$W = \{z\}$ $R = \{x\}$

$$(\{w, y, z\} \setminus \{z\}) \cup \{x\}$$
$$= \{x, y, w\}$$

interference : when we need two vars at
the same time

V = {vars and regs}   edges = I

x I y   iff   ∃k. LiveAfter(k) ⊇ {x,y}

Move-biasing = M

I)

M)
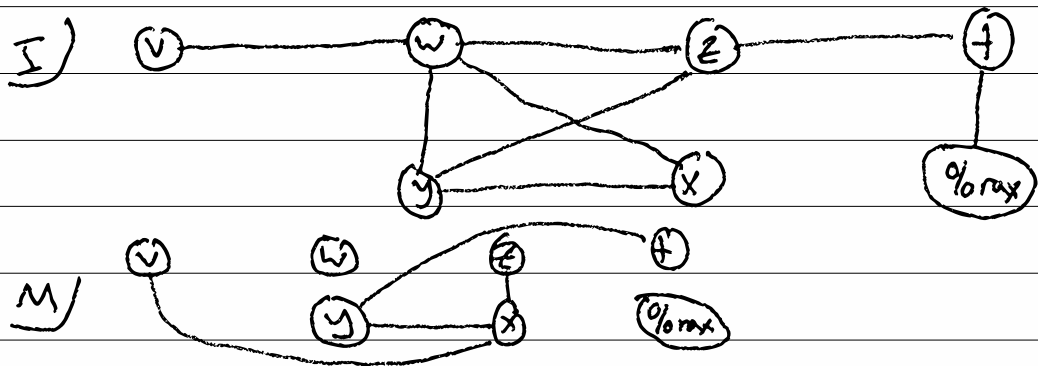
5-6)

Suppose inst k is ....

    (addq s d)

      $\forall v \notin$ LiveAfter(k), we'll add (d, v) to I

                 except for $v = d$

    (movq s d)

      $\forall v \in$ LiveAfter(k), we'll add (d, v) to I

                except for $v = d$ or $v = s$

If inst k treats a register r ~~keeps~~ "special" then add

     (r, v) to I for all $v \in$ LiveAfter(k)

cally it treats caller-saved regs special and %rax

       b> rax, rdx, rcx, rsi, rdi, r8-r11