Loop terminology

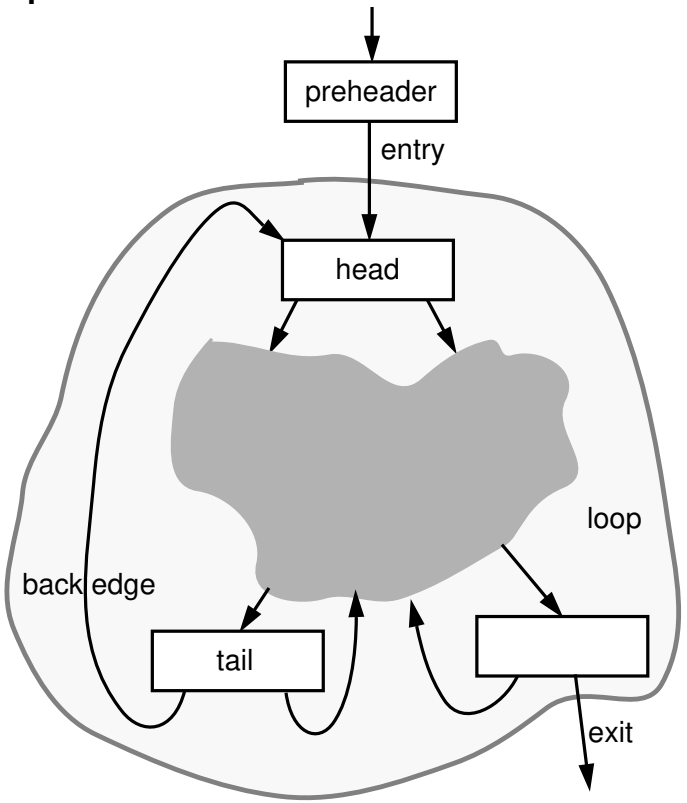
loop: strongly-connected component in CFG

loop entry edge: source not in loop, target in loop
loop exit edge: the reverse
back edge: target is loop head node

loop head node: target of loop entry edge
loop tail node: source of back edge
loop preheader node:
single node that's source of loop entry edge

nested loop: SCC that's a subset of another SCC
reducible loop: loop with only 1 loop head

Example



Pragmatics of iterative analysis

To speed iterative analysis, want to test guess ASAP

- ideal: proceed along shortest path from loop head to loop tail
- practical: don't follow loop exit until fixpoint reached

Another example: constant propagation

Goal: data flow analysis that implements constant propagation

What info computed for each program point?

Initial info?

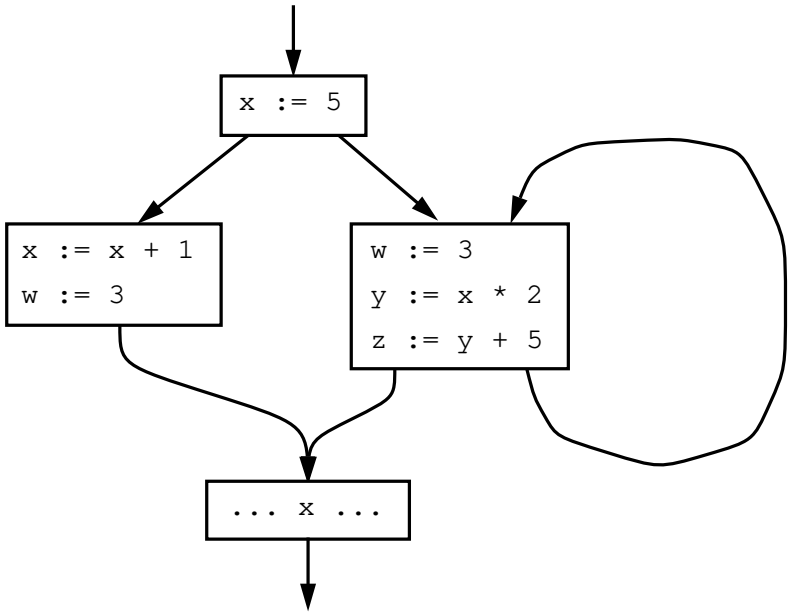
Effect of `x := y + z` (GEN and KILL)?

Effect of `*p := *q + *r`?

Merge function?

Initial guess for back edge to start iterating?

Example



May vs. must info

Some kinds of info imply possibilities: **may** info

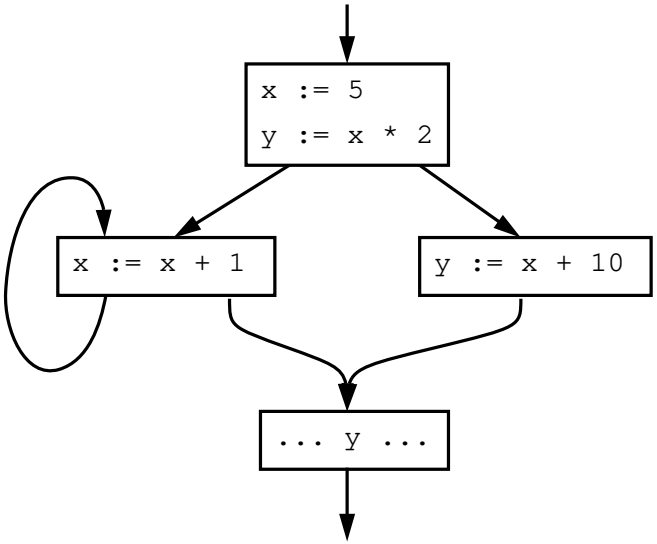
Other kinds of info imply guarantees: **must** info

	May	Must
desired info	small set	big set
GEN	add everything that might be true	add only if guaranteed true
KILL	remove only if guaranteed wrong	remove everything possibly wrong
MERGE	$\cup$	$\cap$
guess for back edge (optimistic)	$\varnothing$	universal set

Another example: live variables

Want the set of variables that are **live** at each pt. in program

- live: might be used later in the program

Supports dead assignment elimination, register allocation

Example