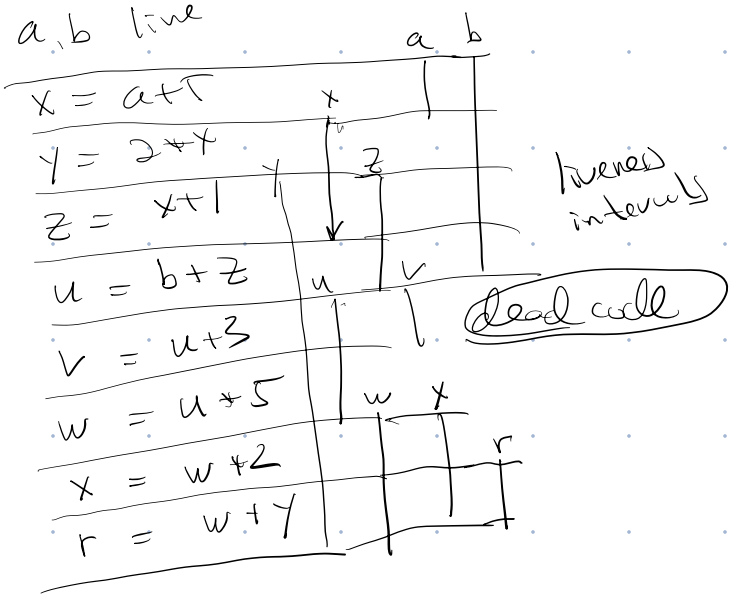
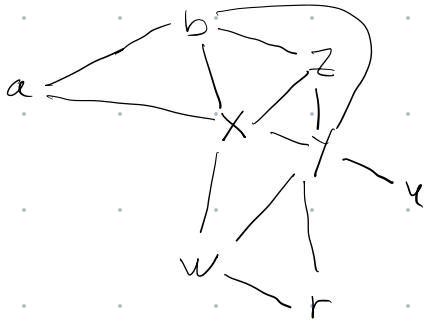
a, b live



r, x live



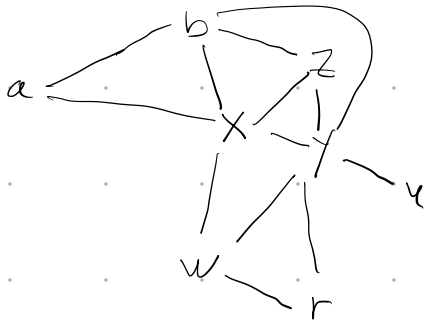
Algorithm to color with $k = 3$ colors

SIMPLIFY

(1) remove each node with $< k$ neighbors and push on a stack

if we color the remaining graph then we can color that node

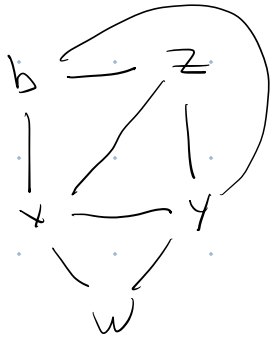
For us, a, u, r have < 3 neighbors



interference graph

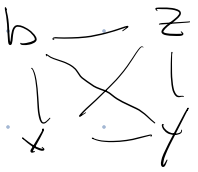
Assume we have 3 registers. Can we "color" the nodes of the graph with 3 colors R1, R2, R3 so no edge connects nodes of the same color?

Stack a, u, r



Now w has < 3 edges so push on stack

Stack a, u, r, w



(2) Spill each node has ≥k edges

So pick one to put in memory

say b

Memory b

Stack a u r e, (b)

---

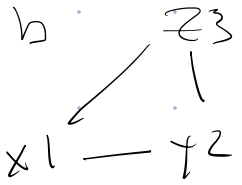
(3) Select Colors

b has no register

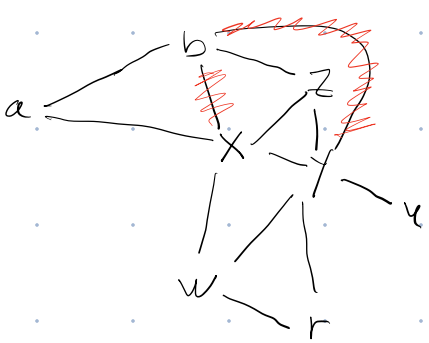
So rewrite code to

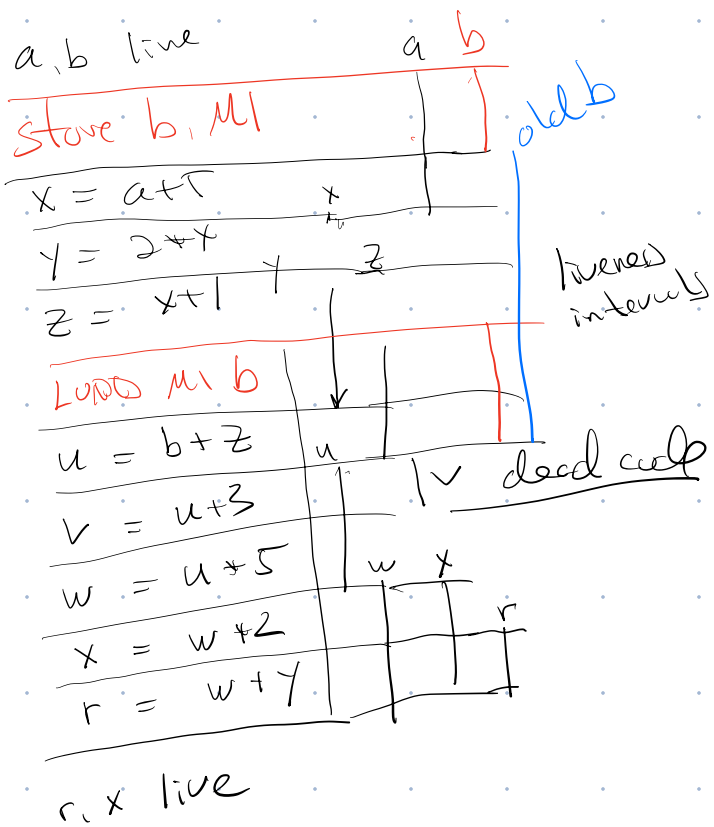
b stored in memory

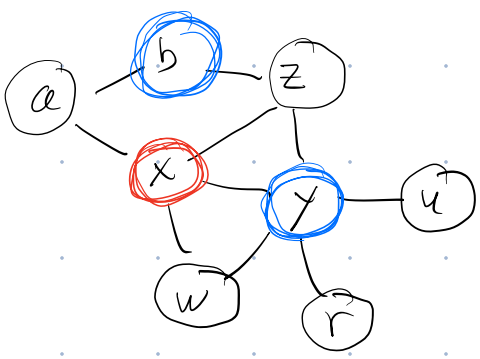
and Start Over



simpler graph







Stack

Simplify a, b, w, r, u, t, y, z

Rebuild graph from stack and

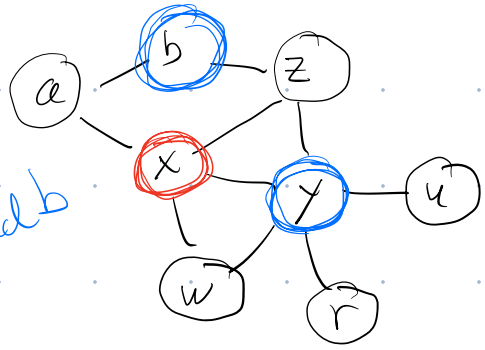
Select colors

R1 black

R2 blue

R3 red

Rewrite code with registers



a in R1
b in R2

	a, b live	a	b
Store R2, M1	Store b, M1		
R3 = R1 + 5	x = a + 5	x	
R2 = 2 * R3	y = 2 * x	y	z
R1 = R3 + 1	z = x + 1		
Load M1, R2	Load M1 b		
R1 = R2 + R1	u = b + z	u	
dead code, delete	v = u + 3		
R1 = R1 + 5	w = u + 5	w	x
R3 = R1 + 2	x = w + 2		r
R1 = R1 + R2	r = w + y		

old b

liveness intervals

r, x live

r in R1
x in R3