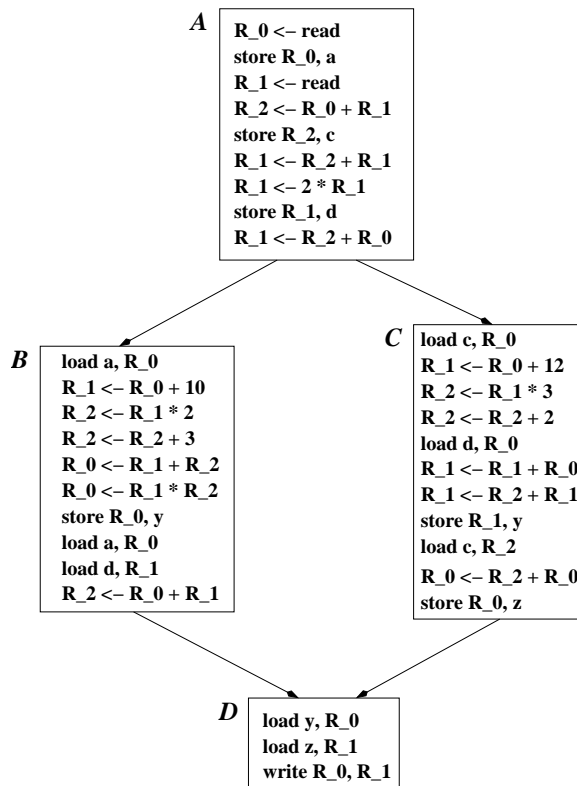


CS423 - 01 Homework #3 – SOLUTIONS

DUE: Wednesday, April 26, 2006, *in class*

If you are a graduate student, you must do all of the following 6 problems; if you are an undergraduate student, you must do problems 1- 5. In any case, your solutions are to be handed in at the beginning of class on 3/16. Show your work to receive partial credit!

1. Do problem 1, parts *b* and *c* (we did part *a* in class) from section 13.4 on page 775 of the text book.
part (*b*)



part (*c*) One could simply leave the *LIVEOUT*(*a*) values in registers and then re-adjust the register assignments in blocks *B* and *C* so that they do not use any of the registers mapped to values found in *LIVEOUT*(*a*). This can avoid some loads (like loading the value *c* in block *C*).

2. Do problem 1, part *a* from section 13.5 on pages 775-776 of the text book. You do not have to do parts *b* or *c*!

We first need to find a node with out degree of less than 3. Only node r_3 meets this constraint. Removal of node r_3 reduces the out degree of node r_1 to be 2 (but no others have out degree less than 3), and thus we can remove r_1 . Unfortunately, none of the remaining nodes has an out degree of less than 3 at this point in time, so we must spill one - a good choice might be the one with highest out degree remaining (as it conflicts with the most other values). Node r_2 matches this concern, and is removed but *NOT* placed on the stack of removed nodes! Now, all remaining nodes can be removed and placed on the stack. This leads to:

STACK: (top) $r_7, r_6, r_5, r_4, r_1, r_3$ (bottom)

and *SPILLED* = r_2 .

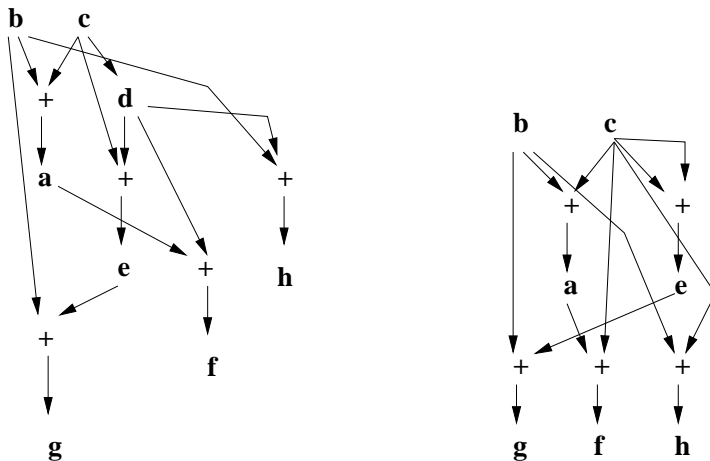
Now we just pop nodes off the stack, giving them a unique color that does not violate the 3-color-ability of the example. In this case, the assignments might be (assuming the colors are 1, 2, and 3):

$$r_7 \rightarrow 1, r_6 \rightarrow 1, r_5 \rightarrow 2, r_4 \rightarrow 2, r_1 \rightarrow 1, r_3 \rightarrow 3$$

Interestingly, one could now color r_2 with 3 successfully.

3. Do problem 2 (all parts) from section 8.3 on page 759 of the text book.

part (a)



part (b)

$ \begin{aligned} a^2 &\leftarrow b^0 + c^1 \\ d^1 &\leftarrow c^1 \\ e^3 &\leftarrow c^1 + d^1 \\ f^4 &\leftarrow a^2 + d^1 \\ g^5 &\leftarrow b^0 + e^3 \\ h^2 &\leftarrow b^0 + d^1 \end{aligned} $	$ \begin{aligned} a^2 &\leftarrow b^0 + c^1 \\ e^3 &\leftarrow c^1 + c^1 \\ f^4 &\leftarrow a^2 + c^1 \\ g^5 &\leftarrow b^0 + e^3 \\ h^2 &\leftarrow b^0 + c^1 \end{aligned} $
---	---

part (c) Building a DAG looks at names, so it cannot see that c and d have the same value after the assignment $d \leftarrow c$. Thus the DAG method cannot find the redundancy in the left hand block.

4. Do problem 1, parts a and b (not part c) from section 8.4 on page 760 of the text book.

part (a) Ignoring the EBB's of size 1, we get EBB's

$$\{\{1, 2\}, \{3, 5\}, \{3, 6\}, \{3, 5, 6\}, \{4, 7\}\}$$

part (b)

$$\{\{4, 7\}, \{3, 5, 6, 8\}\}$$

5. Do problem 1 (all parts) from section 8.5 on page 761 of the text book.

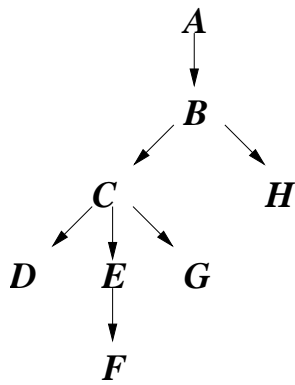
part (a)

$$\{A\}, \{B, C, D, E, F\}, \{G\}, \text{ and } \{H\}$$

part (b)

NODE	A	B	C	D	E	F	G	H
DOM	{A}	{A,B}	{A,B,C}	{A,B,C,D}	{A,B,C,E}	{A,B,C,E,F}	{A,B,C,G}	{A,B}

part (c)



6. Do problem 2 (all parts) from section 8.6 on page 762 of the text book.

The first thing to do is (in part to simplify this written answer) to give each expression a numeric identifier (i.e. expression number):

expr	$2*c$	$a+d$	$c+c$	$d+e$	$b+f$	$g+f$	$k+21$	$a+c$	$c+d$	$a+b$	$k+3$	$f+c$
value #	1	2	3	4	5	6	7	8	9	10	11	12

Now, performing dataflow analysis gives:

Block	DEExpr	VARkill	EXPRkill	Avail
A	1,2	a,b	2,5, 8,10	-
B	4	c,e	1,3,4,8,9,12	1,2
C	5	c,e,f	1,3,4,5,6,8,9,12	1,2
D	5,10	g,k	6,7,11	2,5
E	5	d,k	2,4,7,9,11	2,4
F	11,12	a,d,j	2,4,8,9,10	2,5
G	5,6,7	d,h,i	2,4,9	5

Resulting in re-written code:

