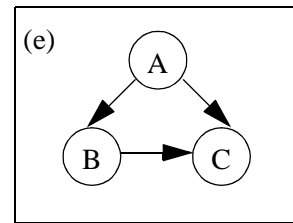
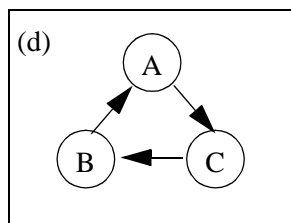
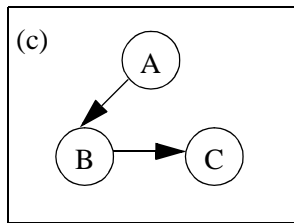
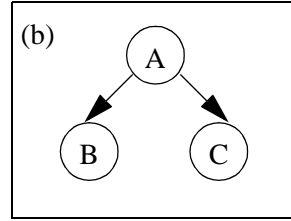
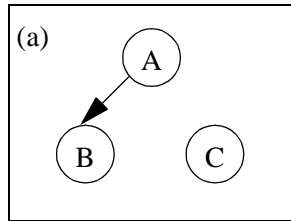


# CS 243

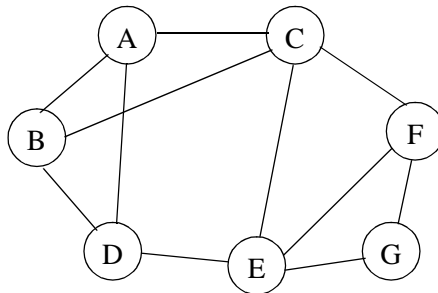
## A Sample Midterm Examination

1. (10 points) Which of the following are semi-lattice diagrams?

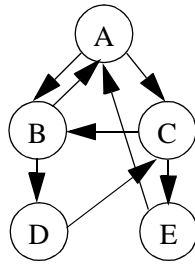


Answer: \_\_\_\_\_

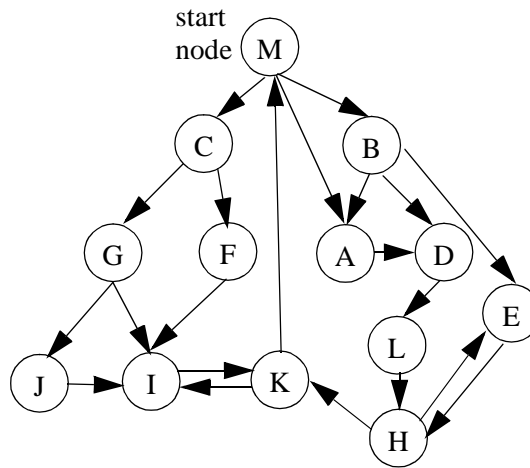
2. (5 points) Suppose you have a machine with 3 registers. Can you find a register allocation for the following interference graph such that no spilling is necessary? Your answer may be one of: yes, no, "I don't know". If the answer is yes, specify a register assignment. If the answer is no, explain why.



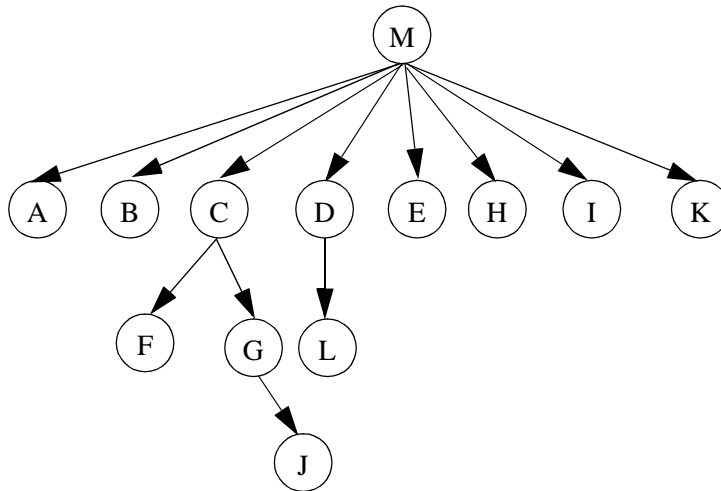
3. (5 points) Draw the dominator tree for the following graph:



4. (10 points) Consider the following control flow graph:

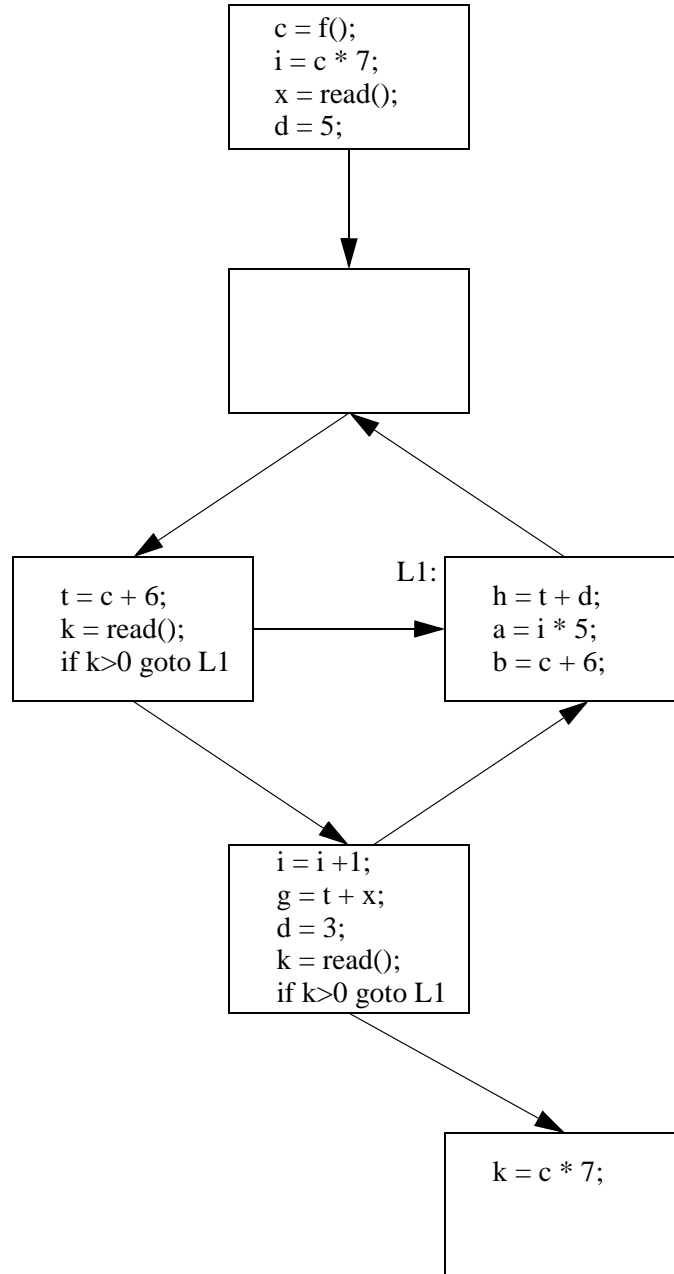


And here is the dominator tree for the above control flow graph.



- What are the natural loops in this flow graph? (Show intermediate steps for partial credit.)
- Is the flow graph reducible? Why or why not?

5. (20 points) Apply all the machine-independent optimizations discussed in class to the following code fragment. You should assume all the variables in the program to be live on exit. Clearly describe the sequence of optimizations you apply and show the final result.



6. (25 points) Design an iterative data flow analysis algorithm to find the signs of variables in a program. Your algorithm must determine, for each basic block, if each variable in the program is positive, non-positive, equal to zero, negative, non-negative, or unknown. The instructions in the program are: assignments, additions, subtractions, squares ( $\text{square}(x)=x*x$ ), conditional and unconditional branches. You do not need to optimize the branch operations.
- What is the direction of your data flow analysis?
  - Draw a diagram of the semi-lattice for one variable, identifying the top and bottom elements clearly.
  - How do you initialize the iterative algorithm?
  - Define the transfer function of a basic block.
  - Is your transfer function monotone?
  - Is your transfer function distributive?
  - Will your algorithm converge? If so, why?