**Problem 1. [16 points]** A portion of a computer program consists of a sequence of calculations where the results are stored in variables, like this:

$$\begin{array}{rrcl}
 & \text{Inputs:} & & x, y \\
\text{Step 1.} & a & = & x - 24 \\
2. & b & = & x * a \\
3. & c & = & 3 \\
4. & d & = & y - c \\
5. & e & = & y * *c \\
6. & f & = & e + 1 \\
 & \text{Outputs:} & & b, d, e
\end{array}$$

A computer can perform such calculations most quickly if the value of each variable is stored in a *register*, a chunk of very fast memory inside the microprocessor. Programming language compilers face the problem of assigning each variable in a program to a register. Computers usually have few registers, however, so they must be used wisely and reused often. This is called the *register allocation* problem.

In the example above, variables $x$ and $y$ must be assigned different registers, because they hold distinct input values. Furthermore, $c$ and $d$ must be assigned different registers; if they used the same one, then the value of $c$ would be overwritten in the fourth step and we'd get the wrong answer in the fifth step. On the other hand, variables $b$ and $d$ may use the same register; we no longer need $b$ and can overwrite the register that holds its value. Assume that the computer carries out each step in the order listed and that each step is completed before the next is begun.

**(a)** [6 pts] Recast the register allocation problem as a question about graph coloring. What do the vertices correspond to? Construct the graph corresponding to the example above.

**(b)** [5 pts] How many registers do you need?

**(c)** [5 pts] Suppose that a variable is assigned a value more than once, as in the code snippet below:

$$\begin{array}{c}
\ldots \\
t = r + s \\
u = t * 3 \\
t = m - k \\
v = t + u \\
\ldots
\end{array}$$

How might you cope with this complication?