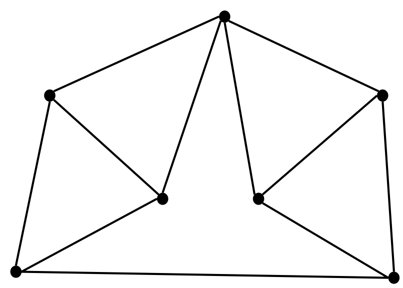
In-Class Problems Week 6, Mon.

Problem 1. Let G be the graph below[1]. Carefully explain why $\chi(G) = 4$.



Problem 2. 6.042 is often taught using recitations. Suppose it happened that 8 recitations were needed, with two or three staff members running each recitation. The assignment of staff to recitation sections is as follows:

- R1: Tina, Jay, Jessica
- R2: Tina, Tom, Albert
- R3: Jay, Jeff
- R4: Chiyoun, Tom, Jessica
- R5: Chiyoun, Srini, Albert
- R6: Srini, Jeff
- R7: Srini, Tom
- R8: Jay, Jeff, Albert

[1] From *Discrete Mathematics*, Lovász, Pelikan, and Vesztergombi. Springer, 2003. Exercise 13.3.1

Two recitations can not be held in the same 90-minute time slot if some staff member is assigned to both recitations. The problem is to determine the minimum number of time slots required to complete all the recitations.

(a) Recast this problem as a question about coloring the vertices of a particular graph. Draw the graph and explain what the vertices, edges, and colors represent.

(b) Show a coloring of this graph using the fewest possible colors. What schedule of recitations does this imply?

Problem 3. A portion of a computer program consists of a sequence of calculations where the results are stored in variables, like this:

	Inputs:		a, b
Step 1.	c	$=$	$a + b$
2.	d	$=$	$a * c$
3.	e	$=$	$c + 3$
4.	f	$=$	$c - e$
5.	g	$=$	$a + f$
6.	h	$=$	$f + 1$
	Outputs:		d, g, h

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. Computers usually have few registers, however, so they must be used wisely and reused often. The problem of assigning each variable in a program to a register is called *register allocation*.

In the example above, variables a and b 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 second step and we'd get the wrong answer in the third step. On the other hand, variables b and d may use the same register; after the first step, we no longer need b and can overwrite the register that holds its value. Also, f and h may use the same register; once $f + 1$ is evaluated in the last step, the register holding the value of f can be overwritten.

(Assume that the computer carries out each step in the order listed and that each step is completed before the next is begun.)

(a) Recast the register allocation problem as a question about graph coloring. What do the vertices correspond to? Under what conditions should there be an edge between two vertices? Construct the graph corresponding to the example above.

(b) Color your graph using as few colors as you can. Call the computer's registers $R1$, $R2$, etc. Describe the assignment of variables to registers implied by your coloring. How many registers do you need?

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

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

How might you cope with this complication?