CS 206 Homework 7

Fall 2020

1. Let D be a set of size $n > 0$. Explain why there are exactly 2^n binary relations on D that are both symmetric and antisymmetric.

2. Let S be a sequence of n different numbers. A subsequence of S is a sequence that can be obtained by deleting elements of S.

   For example, if S is $(6, 4, 7, 9, 1, 2, 5, 3, 8)$, then $(6, 4, 7)$ and $(7, 2, 5, 3)$ are both subsequences of S.

   An increasing subsequence of S is a subsequence of whose successive elements get larger. For example, $(1, 2, 3, 8)$ is an increasing subsequence of S. Decreasing subsequences are defined similarly; $(6, 4, 1)$ is a decreasing subsequence of S.

   And let A be the set of numbers in S. (So A is $[1, 9]$ for the example above.) There are two straightforward linear orders for A. The first is numerical order where A is ordered by the $<$ relation. The second is to order the elements by which comes first in S; call this order $<_S$. So for the example above, we would have

   $$6 <_S 4 <_S 7 <_S 9 <_S 1 <_S 2 <_S 5 <_S 3 <_S 8.$$

   Let $\prec$ be the product relation of the linear orders $<_S$ and $<$. That is, $\prec$ is defined by the rule

   $$a \prec a' ::= a < a' \text{ and } a <_S a'.$$

   So $\prec$ is a partial order on A.

   (a) List all the maximum-length increasing subsequences of S, and all the maximum-length decreasing subsequences.

   (b) Draw a diagram of the partial order $\prec$ on A. What are the maximal and minimal elements?

   (c) Explain the connection between increasing and decreasing subsequences of S, and chains and anti-chains under $\prec$.

(d) Prove that every sequence S of length n has an increasing subsequence of length greater than $\sqrt{n}$ or a decreasing subsequence of length at least $\sqrt{n}$.

3. A simple graph is called regular when every vertex has the same degree. Call a graph balanced when it is regular and is also a bipartite graph with the same number of left and right vertices.

   Prove that if G is a balanced graph, then the edges of G can be partitioned into blocks such that each block is a perfect matching.

   For example, if G is a balanced graph with $2k$ vertices each of degree j, then the edges of G can be partitioned into j blocks, where each block consists of k edges, each of which is a perfect matching.

4. A portion of a computer program consists of a sequence of calculations where the results are stored in variables, like this (with inputs a, b and outputs d, g, h):

Step	Calculation
1	`c = a + b`
2	`d = a * c`
3	`e = c + 3`
4	`f = c - e`
5	`g = a + f`
6	`h = f + 1`

   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. 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 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 wed 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.

   (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 computers 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:

```
...
t = r + s
u = t * 3
t = m - k
v = t + u
...
```

How might you cope with this complication?

5. Define the distance $d(x, y)$ of two vertices x, y in a graph G to be the minimum length of an x-y path in G (if no path exists, $d(x, y) = \infty$).

   The greatest distance between any two vertices of G is the *diameter* of G, written $\mathrm{diam}(G)$.

   The *radius* of G is defined by finding a *central* vertex, namely the vertex whose greatest distance from any other vertex is as small as possible. Then $\mathrm{rad}(G) = \min_{x \in V} \max_{y \in V} d(x, y)$.

   Show that $\mathrm{rad}(G) \leq \mathrm{diam}(G) \leq 2 \cdot \mathrm{rad}(G)$.