Project Blog: Networks and Spanning Trees

David Ruch
Metropolitan State
Project used in MTH 3170 Discrete Mathematics for Computer Science

Background. My class had 12 students, all CS majors who had taken at least two semesters of programming and two semesters of calculus. We met twice a week with 110 minutes sessions. We had spent two weeks previously on basic graph theory (Grimaldi text), e.g. Euler circuits, but nothing on trees.

Summary. We spent two weeks (four class days) on the project, skipping the Cayley section entirely. Basically we did a "highlights of Prufer and Boruvka" project. Two weeks was not enough time for us to work through all the exercises thoroughly, although I think the students enjoyed the highlights and got a lot out of it. An informal poll afterward suggested that the students liked Boruvka much more than Prufer. I think a stand-alone version of the Boruvka project could be developed and would make a nice two week project.

Implementation.

Day 1. We started right after Thanksgiving, with no advance reading. Since we skipped Cayley, as motivation for Prufer and his symbols,we first spent an hour of class on the definition of a labeled tree and attempting to count all such trees for $n = 2, 3, 4, 5$ vertices. We did $n = 2, 3$ as a class to get started. The students then worked in pairs or alone on finding all trees for $n = 4$. After about 20 minutes, a number of people had 12 trees. As a group we found all 16 trees and students made some interesting observations: at least 2 leaves in every tree, and exactly 3 edges in every tree.

Then they worked on the $n = 5$ problem for 20 minutes or so, with no one finding more than 55 of the 125 trees (I told them in advance how many to search for).

We updated our observations: at least 2 leaves in every tree, and exactly $n - 1$ edges in every tree.

I outlined a way to organize the counting to get all 125 trees, but we agreed that the method didn't generalize very well to higher n.

We all agreed that it would be tough to come up with an elegant way to do the counting. I think this exercise really helped motivate Prufer and his clever symbols!

After the break, we read the introduction together, skimmed over Cayley, and started in on Prufer. We read the first part carefully and I let them work in pairs or alone on Exercises 3.2 and 3.3, creating some symbols from trees and vice versa. By the way, we also did Prufer's own example $\{2, 4, 6, 4\}$ – very helpful! We discussed the results together and the students seemed clear on what was going on at the end of the second hour.

I asked them do Exercises 3.3(h) and 3.3(c) for Day 2, submission by E-mail an hour before class.

Day 2. I checked their 3.3(h) and 3.3(c) before class - the results were good, with a few glitches.

In class we quickly went over 3.3(h) and 3.3(c) together, then they worked in pairs or alone on Exercises 3.4 , 3.5, 3.6. They progressed, but slowly. After an hour working and then break, we talked through these exercises and did Exercises 3.7 and 3.9 as a group to finish Day 2 and our "highlights" of Prufer.

At this point I felt they had a strong understanding of Prufer's symbol construction and his counting argument, except perhaps a rigorous understanding of the one–to-one correspondence between symbols and trees.

I would have liked more time on the latter exercises, but chose to move on to Boruvka instead.

I asked them to do 3.11 and 3.16 for homework.

Comments.

1. I was not too happy with my student solutions to 3.11, they felt it was totally obvious and had trouble articulating a clean argument. One idea would be to add a problem early explicitly having them prove that "A connected graph on n vertices with $n-1$ edges must be a tree".

2. An additional nice exercise would be to prove Prufer's claim that the degree of a vertex k in a tree is m iff k appears $m-1$ times in the corresponding symbol.

To begin Boruvka, I asked them to read pages 20-24 at home and do Exercises 4.1, 4.2, 4.3, submission by e-mail an hour before class on Day 3.

Day 3. Checked their Exercises 4.1, 4.2, 4.3 before class, good results with very few glitches.

Note: It would be nice to have Exercise 4.2 a bit more complex, e.g. illustrating that the nearest neighbor relation is not symmetric.

In class we talked some as a group about Boruvka's notation $[k_{\alpha_1,\beta_1}\ k_{\alpha_2,\beta_2}]$ at the bottom of page 24 for the distance between pairs of H_λ. They then worked alone or in small groups on Exercises 4.5, 4.6, 4.7(a). This took awhile but they got it fine. We discussed a bit as a group and introduced 4.7 (b) at the end of class. I assigned the rest of 4.7 (b) as homework.

Day 4. We reviewed as a class where we were with Boruvka, and outlined what was left to prove the algorithm really works. Since this was the final day for the project (and the course!), we only hit some highlights.

As a class we looked at a numerical example of a hypothetical cycle in $G_0(V)$ to see why the connected components can't have a cycle. Then I let them work alone to build a proof by contradiction. We then discussed it as a class using David P's argument rather than the Exercise 4.8,4.9 approach.

Next we talked through a sketch of the induction proof for $G_m(V)$ as a class and went on to talk about why $G_c(V)$ has minimal total edge length. I let them work for while on Exercise 4.13 – why the edges of $G_0(V)$ must be a subset of T_0. A nice exercise!

We then talked through an outline of the induction proof for $G_m(V)$ and why in fact $G_c(V) = T_0$ (Exercises 4.14 and 4.15) because we were out of time.

We did not have time for the last two exercises on algorithm speed and programming, but they looked interesting and accessible for my students.

I created an extra credit problem (see other attachments) where I gave them a distance matrix M but no diagram with the points shown and asked them to run through Boruvka's algorithm. A number of them did this successfully and seemed to enjoy it.