CO 330, LECTURE 35 SUMMARY

FALL 2017

Summary

First we finished up our discussion of the Prüfer encoding. We went through an example of the inverse Prüfer algorithm, then we looked at the two promised applications:

Proposition 1. *The number of labelled (unrooted) trees on n vertices is n^{n-2}.*

Proof. The Prüfer encoding gives a bijection between the set of labelled trees on n vertices and lists of length $n-2$ in letters $\{1, 2, \ldots, n\}$. (We know this is a bijection because we checked the maps are mutually inverse, and on any such list inverse Prüfer will give a tree because it cannot create cycles since for every edge added one vertex is always a leaf and so does not appear again later in the algorithm, and we know the number of edges and vertices is correct for a tree). There are n^{n-2} such lists. □

We can also lexicographically order trees by their lists under the Prüfer correspondence. This is easy because we can just view the list as a base n representation of a number (after shifting the elements down to $\{0, \ldots, n-1\}$ instead of $\{1, \ldots, n\}$). Specifically this gives the following

```
def rankTree
  input E, n (E edge set of a labelled tree on n vertices)
  L = Pruefer(E, n)
  r = 0
  p = 1
  for i from n-2 to 1
    r = r + p*(L(i)-1)
    p = p*n
  return r
def unrankTree
  input r, n
  for i from n-2 to 1
    L(i) = (r mod n) + 1
    r = floor((r-L(i)+1)/n)
  return inversePruefer(L,n)
```

Next I gave a summary of what we did in this class after the midterm. This left only time for one problem for me to solve from you. The question you asked was how to recursively generate $\mathcal{A} = \mathcal{B} \times \mathcal{C} \times \mathcal{D}$. The short answer is to let $\mathcal{F} = \mathcal{C} \times \mathcal{D}$ because then you just generate $\mathcal{A} = \mathcal{B} \times \mathcal{F}$ and as a separate function generate $\mathcal{F} = \mathcal{C} \times \mathcal{D}$ both of which you know how to do. You can do it all in one function, which I put on the board but the indexing gets a little hairy.

References

The references on the end of the Prüfer stuff are the same as for last class.

If you want more problems to practice on, you have the course notes and also look at the assignments and practice problems from

http://people.math.sfu.ca/~kyeats/teaching/math343.html

Your exam should appear in odyssey now. Good luck and see you there.