

# Math172Ex10

Trustin Nguyen

November 7, 2023

**Exercise 4:** Fix  $n \geq 3$  and let  $A$  be the graph obtained from  $K_n$  by removing one edge. Find the number of spanning trees of  $A$  (we count them as labeled trees, so different trees will be counted separately even if they are isomorphic).

*Proof.* We know that the number of spanning trees on a complete graph is  $n^{n-2}$ , because that is just the number of trees on  $[n]$  which was proved in class. Now the number of spanning trees is a subset of these  $n^{n-2}$  strings. Here is the method for the code:

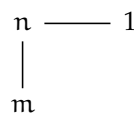
- Find the leaf with the minimal label
- Remove this leaf, and add to the code what the edge led to.
- Repeat this procedure for  $n - 2$  times.
- Stop when there is a tree on two vertices.

Without loss of generality, remove the edge connecting  $1, n$  in  $K_n$ . Then there are two cases:

- Converting a prufer code to a tree ends in adding the edge  $\{1, n\}$  at the end. Because there is a bijection between prufer codes and trees, consider the process of constructing a prufer code.

First show that one of the remaining vertices is always  $n$ . Every tree has at least 2 leaves. Then since  $n$  is the largest number in our graph, there will always be a smaller leaf  $n'$  that we will remove instead of  $n$ . So  $n$  will be one of the remaining vertices at the end.

Suppose the last two vertices are  $1, n$ . We know that whatever leaf we removed before must be connected to either 1 or  $n$ . Suppose that it is connected to  $n$ , and had the label  $m$ . But that is a contradiction, because we must have the graph:



But then since 1 is the smallest leaf, we will have removed  $\{1, n\}$  as an edge, so  $1, n$  are not the last vertices remaining. Then we know that graphs that contain the edge  $\{1, n\}$  contribute to prufer codes that end in 1.

- The edge  $\{1, n\}$  is removed before the last step. Since the vertex  $n$  is never a leaf that is removed until the last step, we know that it is the leaf 1 that is removed that is connected to the vertex  $n$ . Then that means that in our prufer code, this corresponds to adding  $n$  to our code. We must also have no 1's appearing after.

If  $\{1, n\}$  was removed in the first step, then the prufer code contains no 1's and starts with vertex  $n$ . Otherwise, 1 was not a leaf. Then we must have removed

several edges connecting to 1, which would turn 1 into a leaf, which is the smallest one. So then an instance of 1,  $n$  in the prufer code, followed by 0 instances of 1 indicates that we have removed  $\{1, n\}$  as an edge.

Notice that the set of prufer codes that account for these cases are disjoint. For the first case, we require that 1 appears as the last element of the code. So this is just  $n^{n-3}$ , because we set the last element in our  $n - 2$  element string to be 1 and have  $n$  options for the rest of  $n - 3$  places in our string.

In the next case, suppose we had a string of length  $n - 3$ . Then we can account for an instance of 1, 5 in our code with no 1's after by taking the last instance of 1 in our code and replacing it with 1, 5. If there are no 1's, then we append a 5 to the start of the prufer code. There are  $n^{n-3}$  ways to have such strings in this case.

So now we take the number of prufer codes total corresponding to the number of spanning trees on  $K_n$

$$n^{n-2}$$

and remove the number of prufer codes that use the edge  $\{1, n\}$ :

$$n^{n-3} + n^{n-3}$$

to get the number of spanning trees on  $K_n$  with one edge removed:

$$n^{n-2} - 2n^{n-3}$$

this is not defined for  $n = 1, n = 2$ , so the number of spanning trees for those is 1 which completes the proof.  $\square$