Notes from reading Cameron

- Relation of defect k parking functions to x-parking functions from Stanley and Pitman. For a tuple of integers $(x_1, \cdots, x_n)$, an x-parking function is a sequence $(a_1, \cdots, a_n)$ whose order permutation $(b_1, \cdots b_n)$ satisfies $b_i \leq x_1 + \cdots + x_i$. The notation feels clunky/disconnected from the meaning in this context, but the idea is cool. Somewhat lets you refine the idea of defect - indicates informaiton about the maximum number of people who can't park and where empty spaces are allowed to be

- m cars, n spots - you can still use the circular lot argument, you just get $(n + 1 - m)$ possibilities per diagonal coset :)

- Recursive formula leverages the way in which the last spot has to have $k + 1$ cars that eventually try to park there

- **Question(s) :** On page 5, Cameron says "The previous equation suggests to represent $a(r, s, k)$ by a generating function which is ordinary in u and exponential in a combination of v and t". How? What? Why? More specifically, what is ordinary? Exponential? How would you know when to use each of them?

- Interesting limiting behavior kinds of investigation

- Mention of application to queuing. I kind of wonder what would happen if you let cars/tasks have a preference of *after another car*. Not enough to really think about it, but enough to write down the question

- I think that it would be doable to write a dynamic programming version of the recursive equation for counting things to basically expand the table provided in the paper to a broader set of values of n and m. (also could use the table in the paper to double check code)

Notes from reading Douvropoulus

Good descriptions of what a non-crossing partition is

- This paper is so sassy :)

- What's a uniform proof?

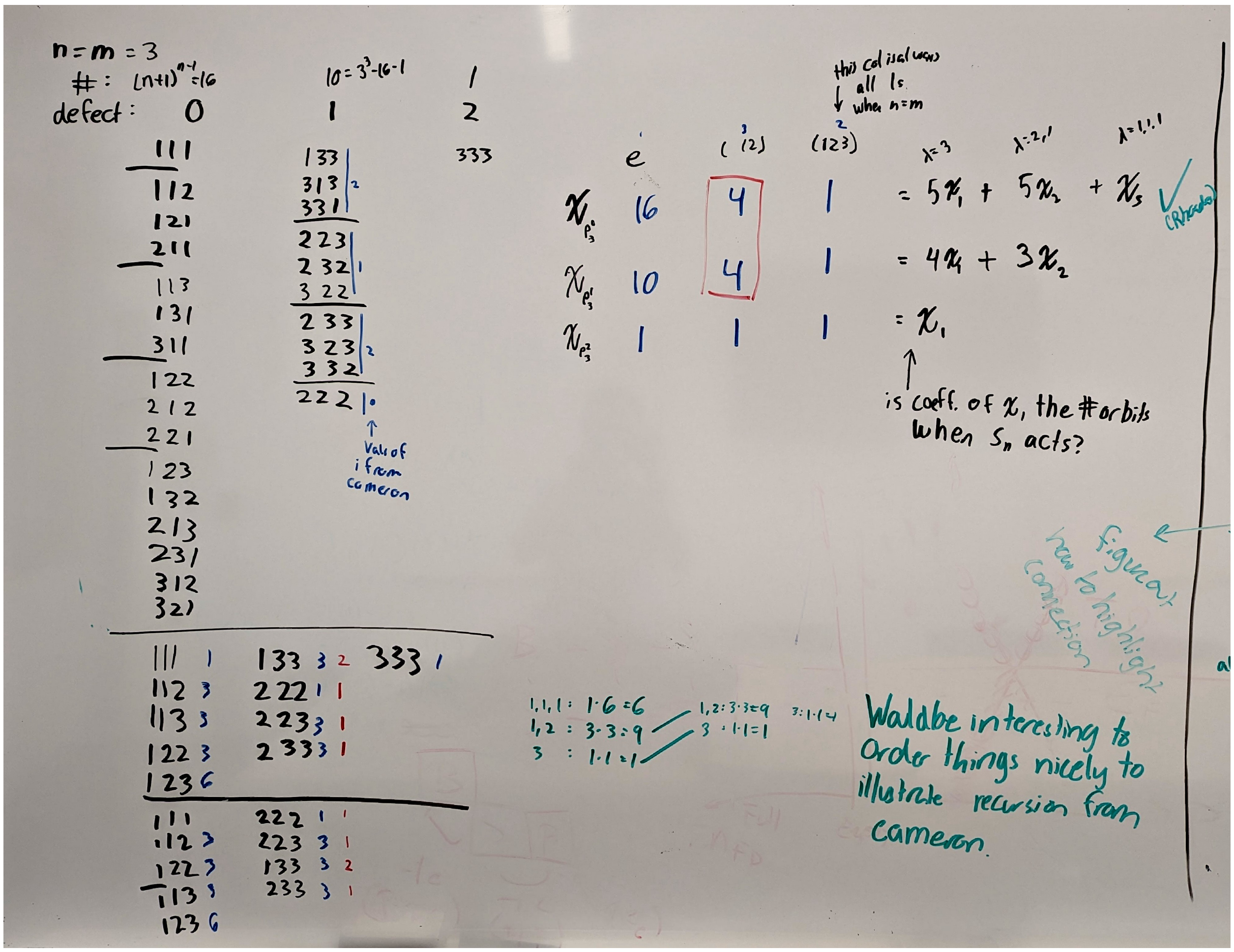
- Weyl groups keep showing up... And are almost always immediately followed by a description of Φ... **Question to Prof O:** Do you know what Weyl groups are? I tend to get lost almost immediately every time they come up, which makes me feel like there might be some background/common knowledge that I'm missing. They seem to be explained in the *people have already seen this* kind of way

- What's B_n and D_n?

- Parking functions are related to a quotient of a polynomial ring (but tensored with the sign representation. The quotient-y bit is kinda related to symmetric functions but with 2 variables, maybe?

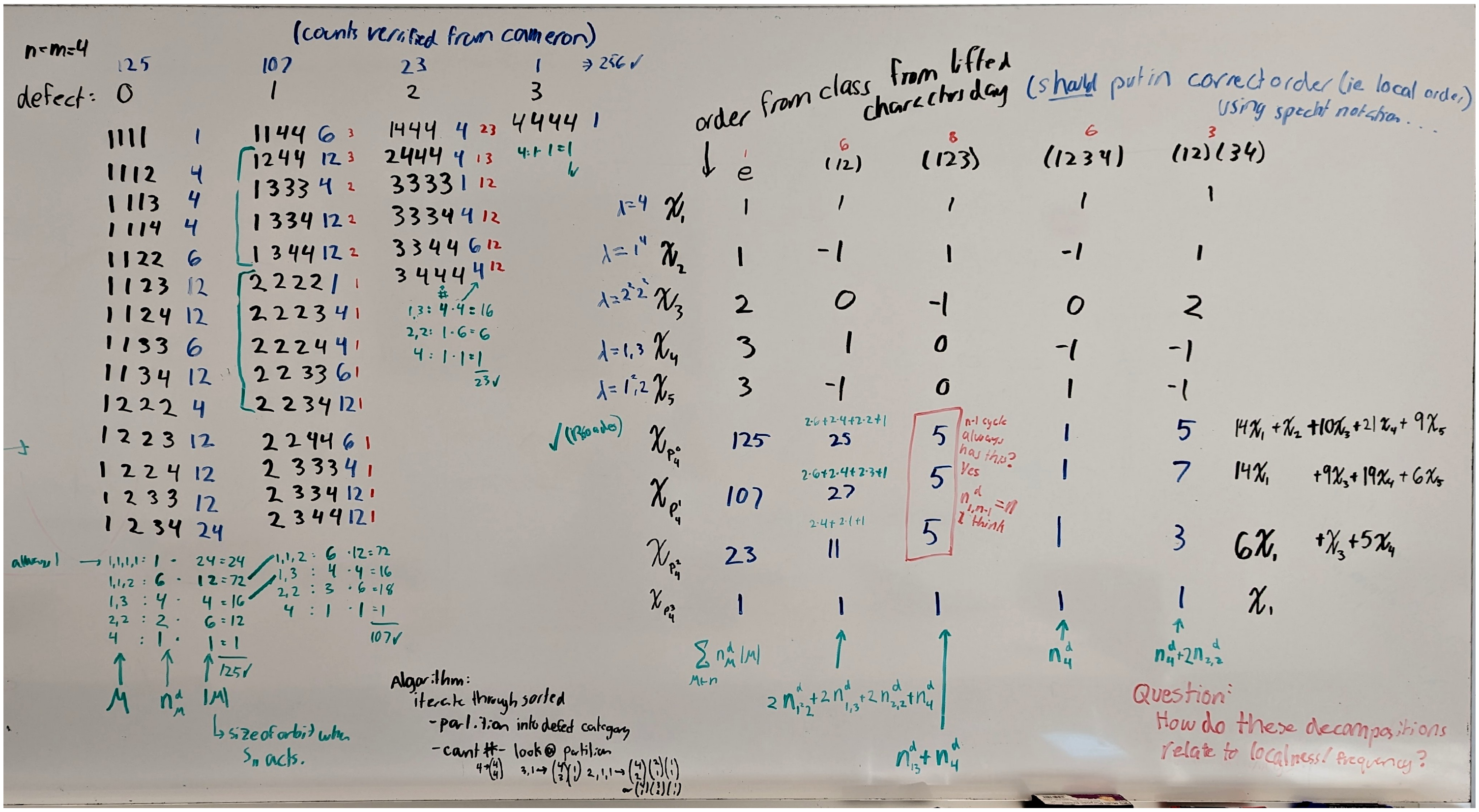
Afternoon computations :)

Computation of the characters of the associated representation for S_3 acting on d defect preference lists for 3 cars in 3 spots



Computation of the characters of the associated representation for S_4 acting on d defect preference lists for 4 cars in 4 spots.

Three steps: Count the partitions, calculate character, decompose into irreducibles



Questions/thoughts raised from afternoon computations

- It could be interesting to order the list of preference lists nicely to see if you could see/highlight the recursion from Cameron

- If I wanted, at this point, I think that I could write code to calculate the corresponding characters for parking functions of defect d

- For both both S_3 and S_4 acting on parking functions, one of the things that I noticed is that the character decomposition of the resulting character into irreducibles seems like it starts out somewhat high frequency and then becomes lower frequency as the defect number increases. It would be interesting to see if you could maybe show this in general... (Question for Prof O: Could you look at the decomposition into irreducibles for S_4 with local-ness in mind? I am shaky on both the irreducible specht modules and what order they come in for local-ness)

- From today's calculation, the key way that I was able to count things was by counting the number of orbits of S_n corresponding to particular partitions for each defect (on the board n_μ^d in green). It would be interesting to see if there was a good way of counting these. For example, for $n_{1,n-1}^d$, I think that $n_{1,n-1}^d = n$, for all defects d (please excuse the fact that n means 2 different things in this expression). Also, all of the permutations are 0 defect parking functions, so $n_{1^n}^d = \begin{cases} 1 & d = 0 \\ 0 & d \neq 0 \end{cases}$. An interesting starting point for this would just be to count the total number of orbits of S_n for each partition

- Also, looking at one of my early questions from working with S_3: is the multiplicity of χ_1 (the trival character) the number of orbits when S_n acts? The answer is yes for S_4 also. Is there an easy way of seeing why this is true??? (Ask Prof O if this seems obvious or surprising) Could I show it is true in general? Do the multiplicities of the other irreducible characters answer other questions? If so, that's also an interesting combinatorial constraint on the coefficents of n_μ^d for computing the character table

- Today, the characters that I computed were all for $n = m$. But it would be interesting to look at what happens when you have a different number of cars and spots

- Also, from trying to compare my decomposition to the one in the Rhoades paper introduction, I remembered that that particular decomposition was not into irreducibles. It would be interesting to look more closely at that decomposition with the things that I've been thinking about regarding wreath products in mind.

Things I didn't do from yesterday:

- Update references

- skim/read Konvalinka

- Use theory about wreath products to work on the example of constructing the irreducibles of $\mathbb{Z}_2 \wr S_3$ (help from page 99)

- Slow down and write things down better (Stanley exercises)