
ROMЧ

The MathILy-Er Record of Mathematics (RoM)

Issue 4: July 26, 2020

Edited by Annie, Jamin, & Lydia

In this Issue (from amazing to boring to awesome…)

Daily Gathers

Turning up the Volume on Playlists

By: Grant

Alice gave Monday's daily gather, and she began by reminding us of a problem that we had seen before in root - Dylan's Playlists/Classical Drum and Bass Fusion (terminology note for Nate's Root: a playlist is a CDBF song, and a song corresponds to a note, with the song's length (one or two minutes) corresponding to one or two beats/slots). Straight away though, a twist was added in: Alice introduced volume.

The way that volume works is that every minute of the playlist is associated with a certain number. If the song is one minute long, its volume can be any positive number up to its minute's associated number (If a one minute song is in minute zero, it will use minute zero's number, if it is in minute one, it will use minute one's number, etc.). If the song is two minutes long, its volume must be one. Perhaps predictably, we want to know the number of ways we can make a playlist accounting for all the different volumes a song can take on.

We came up with a way that we might solve this problem. If we define v_n as the number of potential volumes of the nth slot (starting at zero) and p(n) as the number of ways to make a playlist that ends on the nth slot, we proved that $p(n)=v_n p(n-1)+p(n-2)$, where $p(0)=v_0$ and $p(1)=v_0 v_1+1$. This is because any n-slot song ends in a one minute song (in which case the volume of the ending song can take on v_n values, for a total of $p(n-1)v_n$ such n-slot songs) or a two minute song (in which case there is only one possible volume, 1,) so there are p(n-2) such n-slot songs. Summing over the two possible cases, we see that $p(n)=p(n-1)v_n+p(n-2)$.

Alice then noted that this form kinda looked like a numerator of a fraction in the form of $\frac{p_{n-2}+p_{n-1}v_n}{?} = \frac{?}{?} + \frac{?}{?}$. What are the question marks? We came up with the two equations $\frac{p_{n-2}+p_{n-1}v_n}{v_n} = \frac{p_{n-2}}{v_n} + \frac{p_{n-1}}{1}$ and $\frac{p_{n-2}+p_{n-1}v_n}{v_n p_{n-2}} = \frac{p_{n-1}}{p_{n-2}} + \frac{1}{v_n}$. Alice then came up with a really weird fraction: $v_n + \frac{1}{v_{n-1}+\frac{1}{v_{n-2}+\frac{1}{v_{n-3}+\ldots}}}$ which yields $\frac{p(n)}{p(n-1)}$. We then did this in reverse with $v_0 + \frac{1}{v_1+\frac{1}{v_2+\frac{1}{v_3+\ldots}}}$ which had numerator $p(n)$ (note that we didn't prove this to be true, although we did one example). After that, we ran out of time, but there are certainly many things to ponder about these strange fractions.

Chip-Firing Games on Graphs

By: Rezza

The tour game involved ruins, mega-cities, hyperloop tunnels, tour buses, and tour groups! The game is set up as a random graph made up of dots and lines, where dots are mega-cities and lines are hyperloop tunnels. Between two mega-cities, there can be two different hyperloop tunnels between them, but a megacity cannot have a hyperloop tunnel return to itself. Then a number (any integer) is assigned to each mega-city randomly; these represent the number of tour groups a mega-city has, where negative numbers represent how many the city needs. At any point, a move consists of a mega-city sending out tour buses to mega-cities that are connected to it through hyperloop tunnels, where each bus carries one tour group and each hyperloop tunnel carries one bus.

To turn this into a two-player game, we start with a random graph again, but without numbers. The first player decides which numbers to put on each city, and then the second player removes one tour group from a mega-city of choice. The first player then gets any number of moves to remove all the negative numbers from the graph. If it is possible, then player one wins.

We came to the conclusion that it is possible for player one to win for all loop-free connected graphs starting with 1 tour group on all mega-cities. This was because in this case we could just keep moving tour buses and always not get back to a position we started in. The final question was how this would work for general numbers of tour groups and what types of ruins would need lots of tour groups.

While you're playing around with this tour game, consider the following questions:

1. For which ruins can Player 1 win with only 1 tour group?

2. For which ruins can Player 1 win with only 2 tour groups?

3. What types of ruins will require lots of tour groups?

4. Classify some group of similar ruins by how many tour groups they ne to win. Interesting groups to consider: ruins where every mega-city has hyperloop tunnel to every other mega-city, ruins where one mega-city ha a hyperloop tunnel to every other mega-city, ruins with as many hyperlo tunnels as mega-cities, or any other similar group that you can come up with and classify!