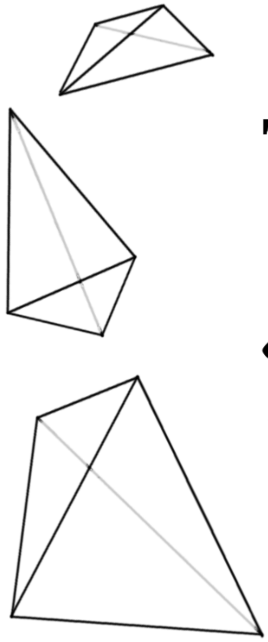
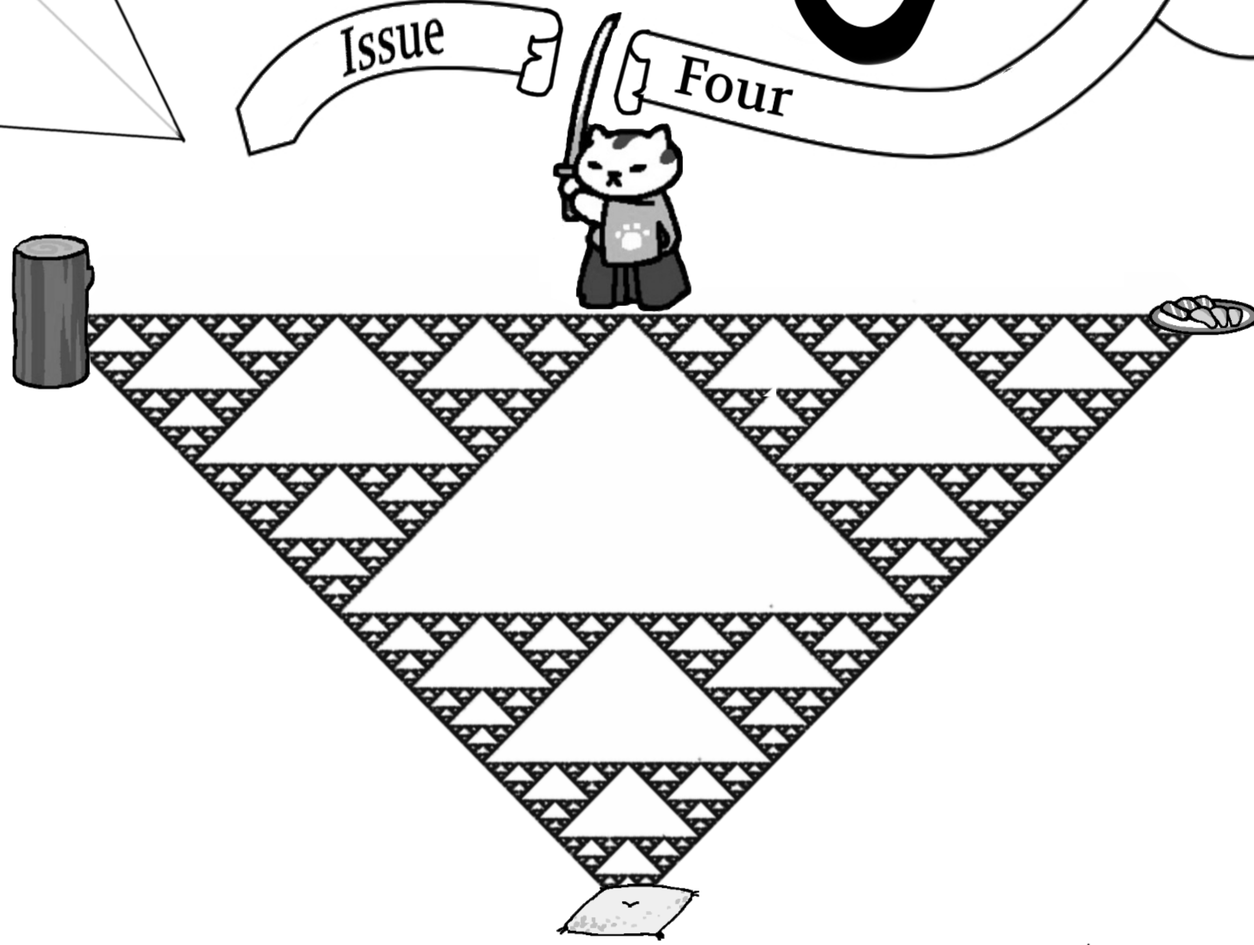
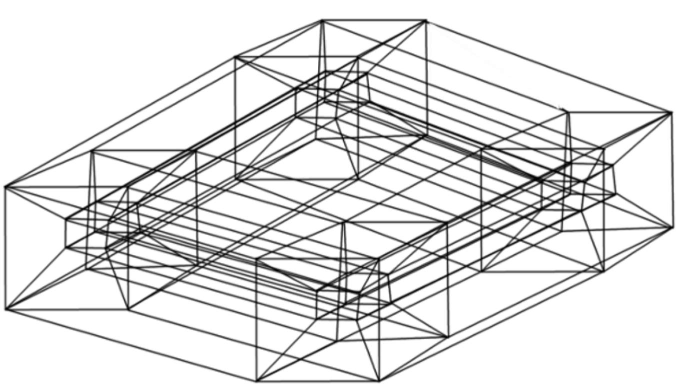
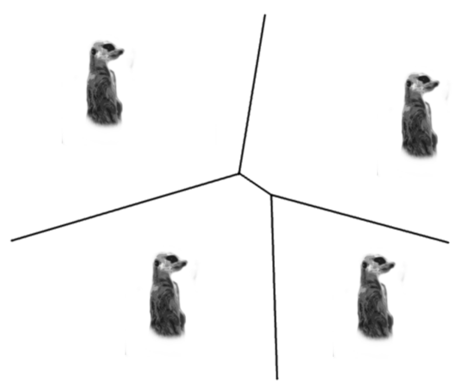


The MathILy Record of Mathematics

Issue Four







cover design by Sean Harrigan

Contents

The Dangers of Pokémon Go!

Talk by Hallie, article by Karol

Pokémon Go! is a dangerous game. This is why a local architect that plans on building new PokéGyms asks you for help. He wants to build n PokéGyms by starting with three that form a triangle and then triangulating it (as we please) so that the other gyms are vertices of the other triangles. But there are restrictions. Our gyms come in three different colors — red, blue and yellow — and the three outer gyms must each have different colors. A gym on the edge between color a and b (of the outer triangle) can only be of color a or b, so for example on the edge between red and blue there can only be red or blue gyms. However, the gyms on the inside of the biggest triangle can be of any color. To reduce fighting between gyms, our architect wants any "smallest" triangle to have vertices of at most 2 distinct colors.

Moreover, Pokémon trainers can travel between areas bounded by gyms. To get from one region to another, they must cross an edge between two gyms. Red trainers can travel through red-yellow edges, blue trainers can travel through blue-red edges, and yellow trainers can travel through yellow-blue edges. Furthermore, each trainer can cross a certain edge exactly once.

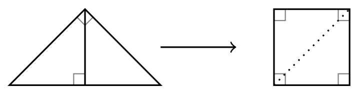
With this information we can ask two questions: "Can the configuration demanded by the architect be achieved?", and "Can PokéTrainers get inside the area of PokéGyms and get out?"

During the Daily Gather, we made conjectures about both questions. We proved that it is impossible to fulfill the architect's original request. We also showed that the number of triple-colored triangles is always odd. Regarding the second question, we made observations such as: "you can exit the area only on the side you entered it", "for each 'smallest' triangle you can travel through at most two edges of that triangle," and "to be able to escape all entrances must be paired." We also showed examples where certain trainers could or could not exit the area.

Cutting up Not Me

Talk by Brian, article by Kaili

Is Brian's face a polytope? In Wednesday's daily gather, Brian revealed that his face is not a polytope and introduced us to "scissor congruence." Two polygons P_1 and P_2 are scissors-congruent if we can cut P_1 a finite number of times into a finite number of pieces, rearrange the pieces, and connect them using anti-scissors to create P_2. For example, an isosceles right triangle is scissors-congruent to a square of the same area, since we can cut along an altitude of the triangle to create two smaller right triangles and connect them to create a square.

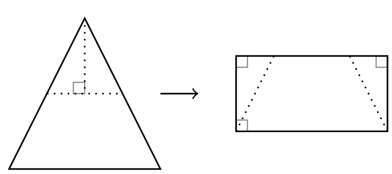


After we investigated several examples of possibly scissors-congruent polygons, an important conjecture arose: any two polygons of the same area are scissors-congruent. To prove this, we first proved that it is possible to turn any polygon into a square through the following steps.

1. Triangulate the polygon.
2. Turn the resulting triangles into rectangles.
3. Turn the rectangles into squares.
4. Merge the squares into a super-square.

To prove that any polygon can be triangulated (Step 1), we inducted on the number of vertices, v, in a polygon. For our base case, $v = 3$, the claim was true since a triangle itself is triangulated, and this is a trivial case. Our inductive hypothesis was that any k-gon with $k \leq n$ for some n can be triangulated. In our inductive step we showed that any $(k+1)$-gon can be triangulated. We considered 3 adjacent vertices U, V, W in our polygon, such that U was between V and W and the angle at U is less than π. If $\triangle UVW$ were completely contained within the polygon, then we could remove it and apply the inductive hypothesis. If $\triangle UVW$ were not completely contained in the polygon, we could orient the polygon so that U became the leftmost vertex of U V and W, and choose X, the leftmost vertex contained within $\triangle UVW$. We could then cut along $\overline{\text{UX}}$ to get two k-gons with $k \leq n$ and apply the inductive hypothesis. Thus, we proved that Step 1 was possible for any polygon.

Next, we confirmed that any triangle can be transformed to a rectangle of equal area (Step 2). Cutting the triangle along its mid-line M and the altitude perpendicular to M gave us two triangles and one quadrilateral, which we rearranged and connected to form a rectangle as in the figure below.



Then, we proved that it is possible to turn any rectangle into a square of equal area (Step 3). There were two cases to consider. The first case is demonstrated in Figure 1. To transform the rectangle into the dotted square, we cut along the thick line. By ASA

Congruence, we showed that $\triangle BCJ \cong \triangle GFH$ and $\triangle FDC \cong \triangle HIJ$. Thus, we can move $\triangle BCJ$ and $\triangle FDC$ to the positions of their respective congruent triangles to create the square. In the case that the rectangle could not be cut and rearranged in this way (Figure 2), we could cut the rectangle in half, stack the two pieces, and connect them. We could repeat this process until the side lengths are within a factor of 4 to each other. Then, we perform the same cut and connections as in the first case. Thus, we proved that Step 3 was possible for any rectangle.

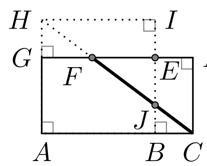


Figure 1: *

Figure 1

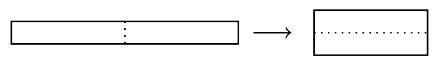
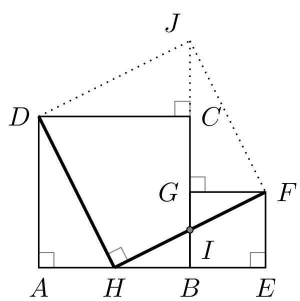


Figure 2: *

Figure 2

Lastly, we proved that any two squares can be turned into a single, larger square (Step 4). We first considered a configuration of two squares $ABCD$ and $BEFG$ as in the diagram below and cut along the thick lines. Again, we used ASA Congruence to show that $\triangle DAH \cong \triangle HEF \cong \triangle DCJ \cong \triangle JGF$. Thus, we could move $\triangle DAH$ and $\triangle HEF$ to the positions of $\triangle DCJ$ and $\triangle JGF$ and connect them to create a single, larger square $DHFJ$. Using this algorithm, we could merge all the squares resulting from Step 3 to form the super-square in Step 4.



We ultimately proved our original claim by using our proofs of Steps 1-4 in conjunction with the logic that if we can turn polygon P_1 into a super-square and turn the super-square into polygon P_2, then we can turn P_1 into P_2. Brian declared that this proved claim is known as the 2-Dimensional Theorem; he then posed a final question. That is, what are the analogues of this theorem in 1 dimension and 3 dimensions? Brian closed his presentation by imparting to us that the so-called “3-Dimensional Theorem” is false.

Life Lessons Learned from Primality Testing

Talk by Rachel Shorey, article by Adam

The Thursday's Daily Gather was delivered by Rachel Shorey from *The New York Times*. She talked about the significance of the prime numbers and showed us some algorithms that can check whether or not certain number is a prime.

Our first idea of checking the primality of a certain number n was to divide it by every number smaller than n. Immediately, we figured out that dividing by every smaller number doesn't make any sense and that we only have to divide by the primes that are smaller than n. Rachel challenged this simple idea, informing us that the run time of this algorithm would be at best 1024 bits - approximately something between 2^{1023} and 2^{1024}. In the best-case scenario the run time would be around $9.480752e + 44$. At this point it is worth mentioning that the universe is only 4.3233912+17 old...

Then, Rachel showed us the Monte Carlo Algorithm — a randomized algorithm whose running time is deterministic, but whose output may be incorrect with a certain (typically small) probability. An example of such algorithms is the Miller-Rabin primality test — one of the most common algorithms for checking whether or not certain number is a prime. The Miller–Rabin test relies on an equality or set of equalities that hold true for prime values, then checks whether or not they hold for a number that we want to test for primality. Here is the example, which shows how this algorithm actually works:

The Rabin-Miller Test — Examples

$\boldsymbol{n = 252601}$, $n-1 = 2^3 \cdot 31575$. Choose $a = 85132$. $a^{31575} \equiv 191102 \pmod{n}$ $a^{2\cdot 31575} \equiv 184829 \pmod{n}$ $a^{2^2\cdot 31575} \equiv 1 \pmod{n}$ *Conclusion:* n is ***composite***. (184829 is a square root of 1, mod n, different from ±1.)	$\boldsymbol{n = 3057601}$, $n-1 = 2^6 \cdot 47775$. Choose $a = 99908 \pmod{n}$. $a^{47775} \equiv 1193206 \pmod{n}$ $a^{2\cdot 47775} \equiv 2286397 \pmod{n}$ $a^{2^2\cdot 47775} \equiv 235899 \pmod{n}$ $a^{2^3\cdot 47775} \equiv 1 \pmod{n}$ *Conclusion:* n is ***composite***. (235899 is a square root of 1, mod n, different from ±1.)

In summary, this Daily Gather was very interesting, informative, and inspiring. Not only did we encounter algorithms that are widely used in cryptography, but we also met an amazing personality. Rachel's life shows that majoring in math opens up many different opportunities in life and for that reason is extremely valuable. No one would suppose that after graduating in math you will be able to work at one of the best-known newspapers in the world. In fact, she works there as a software engineer, which essentially means that she analyzes and cleans huge amounts of data trying to find interesting information. The conclusion of this Daily Gather is simple: study math.