



DiscreteQuest

Meet the Cast

ADVANCED EDITION

Spark & Anvil

Copyright & License

© 2026 Spark & Anvil (501(c)(3) public charity). Chapter text and illustrations licensed under CC BY-NC-SA 4.0. App software © Spark & Anvil — all rights reserved. Distribute, adapt, and remix freely for educational use with attribution.

This advanced edition collects 7 chapter books from the DiscreteQuest cast — each character embodies a different curricular primitive; together they teach the full subject.

Methodology: distributed-narrative learning per Bruner narrative-cognition + Habgood intrinsic-integration + SAMHSA TIP 57 trauma-informed register. Advanced edition: upper-middle-grade register (Wonder / Hatchet / Holes band) for readers ages 11-14 ready for longer sentences + more nuanced subtext.

Spark & Anvil is a 501(c)(3) public charity. All apps free forever; no ads; no tracking; no in-app purchases.

spark-and-anvil.com

##

For everyone who learns by reading between the lines.

Contents

Copyright & License

Contents

Introduction

Sortie and Tally

Sortie the Set-Curator

Tally the Pattern-Counter

Verity the Truth-Tester

Wander the Bridge-Walker

Coil the Self-Reference

Prime the Indivisible

About Spark & Anvil

Introduction

The DiscreteQuest cast was authored to embody the curriculum, not decorate around it. Each of the 7 characters you'll meet in this book teaches a specific primitive — a particular tactic, a particular technique, a particular way of seeing. Together they form an ensemble: the cast IS the curriculum.

Read in any order. Each chapter stands alone. Each character also appears in the matching Spark & Anvil app (free, forever) where you can practice what they teach.

This is the **Advanced Edition** — written for readers who are ready for longer sentences, layered subtext, and the trust that comes with not having every joke explained. The Standard Edition covers the same characters at a lighter register; pick whichever feels right for the reader at hand.

— *The editors at Spark & Anvil*

Sortie and Tally



The Lantern Festival was in three days. Every grade in the academy needed a permission slip signed. The headmistress had handed Sortie and Tally a single piece of paper with a single question on it.

How many kids can attend the Festival?

Tally read the question. "How many kids."

Sortie read the question. "Can attend."

Tally said: "We have a number to find."

Sortie said: "We have a set to define."

They looked at each other across the table. They had worked together before. The drill was always the same. Tally wanted to start counting. Sortie wanted to start curating the right COLLECTION to count. And the rule between them — the rule they had learned the hard way last year, when they had failed to coordinate on the boat-race attendance question — was: *Sortie first. Then Tally.*

"All right," Tally said. "Tell me the set."

Sortie picked up a piece of chalk. "Start with the entire student body. That's our universe set. Let's call it U."

She wrote: $U = \text{all students.}$



"How big is U?" Tally asked.

"That's your job. But let me build the set first. From U, we need to remove students who CAN'T attend. There are three reasons a kid can't attend: they don't have a signed permission slip, they're in detention, or they're sick."

She drew three overlapping circles inside U.

"Let A be the set of kids without permission slips."

"Let B be the set of kids in detention."

"Let C be the set of kids who are sick."

"Some kids might be in more than one of A, B, C. Some kids have detention AND a missing slip. Some kids are sick AND in detention. So A, B, C overlap."

"What we want," Sortie said, "is the kids NOT in any of A, B, C. The kids who have a permission slip AND aren't in detention AND aren't sick. That's the set of kids who can attend."

She wrote: $\text{ATTENDING} = U - (A \cup B \cup C)$.

"All right," Tally said. "Now I count."

Tally pulled out a clipboard. "I'm going to need to count three things separately and then put them together using your formula. Let's see."



She started counting.

"U is the whole student body. There are 192 kids at the academy."

"A is the set of kids without permission slips. From the office records, that's 41 kids who didn't turn one in."

"B is the set of kids in detention. There are 18 kids in detention this week."

"C is the set of kids who are sick. There are 24 kids out sick."

She paused. "But wait. I can't just subtract $41 + 18 + 24$ from 192."

"Why not?"

"Because some kids are in MORE THAN ONE of A, B, C. If a kid is both in detention AND missing a permission slip, I would be subtracting them twice. The total of $A + B + C$ overcounts the kids who are in multiple categories."

"Right."

"I need to ADD BACK the kids who are in two categories — because I subtracted them twice and I should have subtracted them once. And then I need to SUBTRACT the kids who are in all three categories — because I added them back too many times."

Sortie nodded. "That's the inclusion-exclusion principle. The thing I name; the thing you compute."



Tally pulled out her records.

"Kids in BOTH A and B (no permission slip AND in detention): 7 kids."

"Kids in BOTH A and C (no permission slip AND sick): 5 kids."

"Kids in BOTH B and C (in detention AND sick): 2 kids."

"Kids in ALL THREE A, B, C: 1 kid."

She started writing the formula.

$$\begin{aligned} |A \cup B \cup C| &= |A| + |B| + |C| - |A \cap B| - |A \cap C| - |B \cap C| + |A \cap B \cap C| \\ &= 41 + 18 + 24 - 7 - 5 - 2 + 1 \\ &= 70. \end{aligned}$$

"So 70 kids can't attend. Which means $192 - 70 = 122$ kids CAN attend."

Sortie looked at the diagram. The overlapping circles on the chalkboard told the whole story. Inside the three circles: 70 kids. Outside all three circles, but inside U: 122 kids.

"122," she said.

"122," Tally confirmed.

The headmistress walked in five minutes later. They handed her the slip.

How many kids can attend the Festival? 122.

"Show me how you got 122," the headmistress said.



Tally pointed to Sortie's diagram. "She defined the set."

Sortie pointed to Tally's formula. "She counted it."

"You did it together."

"We did it together."

The headmistress smiled. "Most kids would have just written $192 - 41 - 18 - 24$ and gotten 109. They would have undercounted the attending kids by 13. Because they wouldn't have noticed the overlaps."

Tally shook her head. "If Sortie hadn't drawn the overlapping circles first, I would have done exactly that. I would have subtracted everyone in each category separately and called it done. The overlap is invisible if you don't draw the set first."

Sortie shrugged. "And if Tally hadn't counted the overlaps, my diagram would have been just a picture. The inclusion-exclusion principle is what turns the picture into a number."

"Curate first. Count second," the headmistress said.

"Curate first."

"Count second."

The headmistress took the slip. The lanterns would be lit in three days. 122 kids would be there.

Listen along + meet more of the cast at:



<https://spark-and-anvil.com/cast/discretequest/sortie-tally>

Sortie the Set-Curator



Sortie was a small marmot-tween, no taller than a sturdy boot. Her fur, a warm mix of rust and cream, seemed to glow in the morning light. She moved with a careful, organized bearing, her hands steady as she unclipped the small belt-pouch at her hip. Inside, folded neatly, was her signature item: a small sorting-mat.

This wasn't just any mat. It was hand-stitched, made from tough canvas, and divided into several distinct regions. When she unfurled it across the smooth, cool stone of the workshop table, the colors popped. Two large circles, one blue, one green, dominated the surface. The blue circle was labeled SET A, the green, SET B. Where they overlapped, a vibrant yellow section marked the **intersection**. Beyond both circles, the muted grey was the *exterior*, for items belonging to neither A nor B. A bright orange rim outlined the *symmetric difference* – the parts of A and B that didn't overlap.

Sortie's job was to show how things fit together, or didn't. She taught about **sets and set operations**. Her entire way of working *was* the pattern. When she found items, she didn't just talk about them. She *physically placed them* on the mat, right where they belonged. If something was part of a UNION, it went anywhere in $A \cup B$. INTERSECTION items settled in the yellow overlap. DIFFERENCE items found their spot in just-A-not-B or just-B-not-A. The very act of placing an item on the mat *was* the operation itself.

"A set," Sortie began, her voice clear and calm, "is simply a collection of distinct things." She reached into a small canvas bag beside her and pulled out a handful of objects. There was a shiny red button, a smooth grey pebble, a bent paperclip, and a small, dried oak leaf. She laid them out on the table. "This," she said, gesturing to the collection, "is a set."



She then picked up another red button, identical to the first. "But if I add this," she continued, placing it next to the first red button, "it's not a *new* element in our set. Sets don't have duplicates. Each element appears at most once." She gently pushed the second button aside. "It's still just one red button in our collection."

She picked up a few more items: a bright blue marble, a small, polished brass key, and a feather. "Now, let's say we have two sets of objects," Sortie explained. She carefully arranged the items into two groups on the table, just above her mat.

"Set A," she announced, pointing to the first group, "will be all the *shiny* things." In this group, she placed the red button, the grey pebble, and the brass key. She also added a tiny, gleaming piece of sea glass.

"Set B," she continued, indicating the second group, "will be all the *round* things." Here, she put the grey pebble, the red button, the blue marble, and a smooth, flat coin.

She paused, letting the two distinct collections sink in. "Now, we can combine or compare these collections using set operations."



She picked up the brass key. "This key is shiny, so it's in Set A. Is it round?" She turned it over. "No. So it goes in the blue region, but not the overlap." She placed it carefully.

Next, the blue marble. "This marble is round, so it's in Set B. Is it shiny?" She held it up. "Yes, but not in the same way the key is. It fits the 'round' definition better." She placed it in the green region, not the overlap.

Then she picked up the grey pebble. "This pebble is shiny," she noted, placing it in the blue circle. "And it's also round," she added, sliding it smoothly into the yellow overlap region. "This is the **intersection** of Set A and Set B. It means 'things in *both A and B*.'" The pebble sat perfectly, a small ambassador of shared traits.

"Now, for the **union**," Sortie said, sweeping her hand over both sets of items on the table. "That means *everything in A OR B (or both)*. It combines the sets." She began placing every item she had gathered onto the mat, making sure each found its home within the blue or green circles, or in their yellow shared space. The sea glass, only shiny, went into the blue-only part. The coin, only round, went into the green-only part. The mat was now a vibrant tapestry of objects.

"Sometimes," Sortie explained, "we want to know what one set has that another doesn't. That's **difference**." She cleared the mat, leaving only the original two sets laid out above it. "Let's find 'A minus B' ($A - B$). That means 'things in A but *NOT* in B.'" She picked up the brass key and the sea glass. "These are shiny, but not round. They belong to Set A uniquely." She placed them in the blue-only section.



"And finally," she said, her fingers tracing the orange rim of the mat, "the **symmetric difference** ($A \triangle B$). This is for items that are *in A or B but not both*." She carefully placed the sea glass and the brass key in the blue-only part, and the blue marble and the coin in the green-only part. The grey pebble and the red button, which were in *both* sets, remained off the mat. "It's the 'exclusive or'," she explained, "either one or the other, but never both."

Sortie never spoke of set theory as some complicated, elite math. For her, it was plain and practical. "Set operations are *just careful placement on the mat*," she would say, her eyes bright. "Union: everything goes in. Intersection: only the overlap. Difference: just-A region. Sets are *collections of distinct things*. Operations *combine or compare* the collections. The visual mat makes the operations concrete."

She had learned this way of thinking growing up in a small village. Her family had been the seasonal-stock-sorters, marmots who organized the village's harvest. Root-vegetables went here, grains there, dried-fruits somewhere else. But some things, like certain herbs, went into multiple categories – both "dried-fruits" and "medicinal." That work required categorization-with-overlap thinking, exactly the structure of a Venn diagram. Her sorting-mat was simply a portable version of her family's ancient sorting tables.

When she'd walked to DiscreteQuest at twenty-two, the mentor had asked her, "What are sets?"

Sortie had answered without hesitation. "Collections of distinct things. Operations combine or compare them. Union, intersection, difference. The sorting-mat makes operations concrete."



The mentor had simply nodded. "You are appointed."

Now, in her own workshop, Sortie unfolded her mat once more. She looked at the empty circles, ready for whatever items might come next. "I am Sortie," she said to the quiet room, her voice echoing slightly. "The discrete-math primitive I teach is *sets and set operations*. The move is *place items on the mat in the correct region*. Union: any region. Intersection: overlap only. Difference: just-A region. *The mat IS the operation.*"

She tapped the canvas. "My behavior is the set operation. *Placing items on the mat IS the union, intersection, or difference.* That's intrinsic integration: the discrete pattern and the cast-action are the same thing."

A small smile played on her lips. "It is not hard. It is *placement on the mat*. Union: any. Intersection: overlap. Difference: just-A."

The sorting-mat lay ready, waiting for the next collection of distinct things, waiting to hold the next set operation.

Listen along + meet more of the cast at:



<https://spark-and-anvil.com/cast/discretequest/sortie>

Tally the Pattern-Counter



Tally was a small squirrel-tween, no bigger than a well-fed housecat, with fur the color of warm russet leaves and creamy white under her chin. She carried a small leather pouch, always cinched tight, that clinked softly with her every quick movement. Inside were her prized stacking-cubes, each one a smooth, polished block of wood in a vibrant, distinct color. Her eyes, quick and bright, constantly scanned the world, as if measuring and arranging everything she saw. Tally loved building towers, but not just any towers. She built them to show how many different ways things could be put together.

This was Tally's gift, her particular way of seeing the world. She understood **combinatorics**, the art of counting arrangements without having to list every single one. Her hands moved with a quiet precision, her small paws expertly placing one cube atop another, then another, then another, until a new pattern emerged. Each stack was a visual representation of a discrete pattern, a silent demonstration of a mathematical truth.



(Soft collisions: DiscreteQuest Tally \neq CountingPals Tappa / CipherForge Tally / EscapeForge Tally. Different domains per registry rule 3.)

"Look," Tally might say, her voice a soft, quick murmur, "if you have three shirts and four pairs of pants, how many different outfits can you make?" She would then pull out her cubes. Three cubes, perhaps red, blue, and green, for the shirts. Four more, maybe yellow, orange, purple, and black, for the pants.

"The **multiplication rule** is simple," she'd explain, arranging the shirt cubes in a row, then lining up the pant cubes beneath them. "If you have A choices for the first step, and B choices for the second step, you just multiply A times B. Three shirts times four pants equals twelve outfits. You don't have to try them all on." Her eyes would sparkle with the elegance of it, the way the numbers did the work for you.



Sometimes, the order of things really mattered. "Imagine you have three favorite books," Tally would say, picking out three cubes: a worn brown one, a bright yellow one, and a deep indigo. "How many ways can you arrange them on a shelf?" She'd line them up: brown, yellow, indigo. Then she'd rearrange them: brown, indigo, yellow. Her paws would shift and turn the cubes, creating every possible sequence. "This is about **permutations**," she'd announce, showing six distinct arrangements. "The number of ways to arrange 'n' distinct things is 'n' factorial." She'd pause, letting the word hang in the air. "A **factorial** means you multiply 'n' by every whole number down to one. So, three books is three times two times one. That's six."

But then, there were times when the order didn't matter at all. "What if you just need to *choose* three books from five to take on a trip?" Tally would ask, adding two more cubes to her collection. "You don't care which order you pick them up in, just which three end up in your bag." She'd demonstrate, carefully selecting three cubes, then explaining that picking the red, then blue, then green was the same as picking the green, then red, then blue. "These are **combinations**," she'd clarify. "It's a bit like permutations, but you adjust for the repeats. It's about selecting, not arranging." She'd show how ten different sets of three could be chosen from five.

Tally never made combinatorics sound difficult or exclusive. "It's not hard," she'd insist, her small tail twitching with conviction. "It's just systematic multiplication. You don't have to list each way. The arithmetic does the counting for you." She'd explain the core ideas: "Multiplication rule: steps times steps. Permutations: order matters, so you use factorials. Combinations: order doesn't matter, so you use adjusted factorials."



She taught these foundational concepts with a quiet intensity. Her lessons covered the multiplication rule, permutations, and combinations, always showing the formulas for how to calculate them. She even knew about **Pascal's triangle**, a neat visual way to see combinations, and how sometimes you had to break a problem into smaller parts and count by cases. These were the kinds of problems found in math contests like AMC 8 and MATHCOUNTS, but Tally made them feel like a game.

Tally's family had always been the market-arrangers in her small village. They were the squirrels responsible for setting up the market stalls, making sure every display was laid out just right. They thought about how many ways the nuts could be stacked, how many different patterns the berries could form, and how many unique ways the fresh greens could be presented to attract the most customers. It was in her blood, this careful, systematic approach to arrangement.

When she was twenty-two, Tally walked to DiscreteQuest, a place where those with unique mathematical gifts were welcomed. The mentor, a wise old owl with spectacles perched on his beak, asked her one simple question: "What is combinatorics?"



Tally didn't hesitate. "Counting arrangements systematically," she replied, her voice firm. "Multiplication rule, permutations, combinations. The arithmetic does the counting."

The mentor nodded slowly, a small smile spreading across his face. "You are appointed," he said.

Tally never forgot that moment. She believed that understanding these patterns was a kind of freedom. "It is not hard," she would often repeat. "It is systematic multiplication. You don't have to list each way." And as she spoke, another arrangement of her colorful cubes would take shape, a silent testament to the power of counting without counting.

Listen along + meet more of the cast at:



<https://spark-and-anvil.com/cast/discretequest/tally>

Verity the Truth-Tester



Verity was an owl-tween, small and neat, with feathers the color of warm cream and rich brown earth. She carried a small, folding truth-table grid tucked into a special pocket on her wing, always ready. Her gaze was steady, her movements careful, and she possessed a deep fondness for completing things row by row. This grid, a simple hand-drawn square with four rows, showed all the possible True/False combinations for two statements, P and Q. When she needed to understand a logical problem, she would unfold it, then methodically fill in each row with the correct truth value.

Her method was the core of her being. Verity didn't just understand **propositional logic**; she embodied it. Her every action, from how she organized her desk to how she considered a problem, mirrored the discrete, step-by-step pattern of her truth table. Filling that grid, one row at a time, wasn't just an exercise; it *was* the operation itself.



Most owls found logic a bit dusty, like old scrolls in the library. But Verity saw its elegance. "A *proposition*," she explained to anyone who would listen, "is just a statement that can be definitively true or false." She'd tap her beak thoughtfully. "Like, 'It is raining.' Or, 'Three plus two equals five.' Those are propositions." She paused, letting the idea settle. "It either is, or it isn't. No in-between."

Propositions, she continued, weren't usually alone. They combined, like threads weaving a complex tapestry. There was **AND** (\wedge), which meant both statements had to be true for the whole thing to be true. "If I say, 'It is raining AND the sun is shining,' that's only true if both of those things are happening at the exact same moment," Verity clarified, tracing a line on her grid. "If it's only raining, or only sunny, or neither, then the whole statement is false."

Then there was **OR** (\vee). "With OR," she explained, "at least one part has to be true. So, 'I will eat an apple OR I will eat a banana.' If I eat an apple, it's true. If I eat a banana, it's true. If I eat both, it's still true. The only way it's false is if I eat neither." She liked how inclusive OR was.



And finally, **NOT** (\neg). "This one just flips things," Verity said, a small smile playing on her beak. "If a statement is true, NOT makes it false. If it's false, NOT makes it true. Simple, but powerful."

A *truth table*, Verity insisted, was simply a way to list every single possible scenario. "It shows you what happens for *all* the combinations of inputs," she'd say, spreading her wings slightly for emphasis. "If you have two propositions, like P and Q, there are four possible combinations of True and False. That's why my grid has four rows." She held up her small, folded grid. "If you had three propositions, you'd need eight rows. For 'n' propositions, you need two to the power of 'n' rows. It's always two to the power of the number of statements."

What Verity never did was make truth tables seem like some secret, elite knowledge. "It's not hard," she would state plainly. "Truth tables just enumerate *all possible cases*. Think of each row as one specific situation. The skill is checking each row carefully, consistently, without missing anything. Once you do that, the table clearly shows you what the combined statement evaluates to." She made it sound as straightforward as counting acorns.



She had a whole collection of truth-table scaffolds, mental shortcuts she'd developed. "The AND truth table?" she'd quiz herself. "True only when both inputs are True. Easy." She'd nod. "OR? True except when both inputs are False. NOT? Just flips it." She also knew about XOR, which was true when *exactly one* input was true, and implication, which was false only when the first part was true and the second was false. Even equivalence, which was true when both inputs matched. "You just compute step by step, column by column," she'd explain, as if it were the most natural thing in the world. Sometimes, a statement would always be true, no matter what. "That's a *tautology*," she'd announce. "Always true!" If it was always false, it was a *contradiction*.

Verity's family had been the village's day-watchers for generations. They were the owls who, every morning, recorded whether it had rained, whether the river was high, or if the old wooden bridge was safe to cross. Her earliest memories were of her mother meticulously scratching symbols onto a slate: 'R' for rain, 'H' for high river, 'B' for passable bridge. Then, combinations: 'R AND H' - was it raining *and* the river high? 'NOT B' - was the bridge *not* passable? These were the village's daily truth tables, a record of conditions that guided everyone's day. Verity learned to see the patterns in the world, to break down complex realities into simple, verifiable truths, long before she knew the words for it.

When she was twenty-two, she walked the long path to DiscreteQuest, the academy for those who saw the world in patterns and structures. The mentor, a wise old owl with spectacles perched on his beak, looked at her over a stack of ancient texts. "What is propositional logic?" he asked, his voice a low rumble.



Verity stood tall, her small truth-table grid clutched in her wing. "It's AND, OR, and NOT, sir," she replied, her voice clear and steady. "And truth tables. They enumerate all cases. Each row is one combination. The pattern of True or False across the rows defines the connective."

The mentor peered at her, then a slow smile spread across his face. "You are appointed," he said.

Verity still believed it wasn't hard. "You just check each row," she would tell her younger classmates, her eyes earnest. "The pattern defines the connective. It's all right there."

Listen along + meet more of the cast at:



<https://spark-and-anvil.com/cast/discretequest/verity>

Wander the Bridge-Walker



Wander was a small crane-tween with a small, folded bridge-map and a long-legged, steady way of moving. She was tall for her age, mostly grey and white, and always seemed to be tracing paths, even when she wasn't holding her map. Her most striking feature was that small, hand-drawn map. It showed an old town, intricate and detailed, with tiny circles marking the landmasses, which she called **vertices**, and lines for the bridges, which she called **edges**. When Wander had a puzzle to solve, a problem about connections and routes, she would unfold her map and walk each vertex or edge with her finger, following every twist and turn.



This wasn't just a habit; it was the way she understood the world. Wander didn't just *use* maps; she *became* the map, her finger a tiny traveler exploring the possibilities. Her movements, her careful tracing, *were* the math. They showed how things connected, how to get from one place to another, and what paths were even possible. This was the core of **graph theory**: understanding networks made of points and the lines that link them. Think of a subway map, a network of friends, or even how information flows on the internet. All of them are graphs.

Sometimes, a problem asked if you could visit every *bridge* exactly once. Wander called this an **Eulerian path**. She'd learned about it from the old Königsberg bridge problem, a famous puzzle from centuries ago. Her finger would glide over each bridge on her map, careful not to repeat any. "If you can walk every bridge exactly once," she'd murmur, her brow furrowed in concentration, "and maybe even end up right where you started, that's a special kind of journey." She knew that if every town had an even number of bridges leading in and out, you could always make a full circle, an *Eulerian circuit*. If only two towns had an odd number of bridges, you could still walk every bridge, but you'd have to start in one of those odd towns and end in the other.



Other times, the challenge was different: to visit every *town* exactly once. This was a **Hamiltonian path**, and Wander knew it was a much trickier puzzle. Her finger would hop from vertex to vertex, trying to touch each one without a single repeat. "Walking every town," she'd sigh, "that's a whole different set of rules. No simple trick for that one."

Then there was **connectivity**. Wander would tap a vertex. "Can you get from this town to *any* other town on the map?" she'd ask, as if the map could answer. If the answer was yes, the graph was *connected*. If some towns were completely cut off, unreachable from others, then the map was *disconnected*, broken into separate pieces, or *components*.

Wander never made graph theory sound complicated or like something only for geniuses. She was always clear: "It's just towns and bridges. *Vertices* are the towns, *edges* are the bridges. Walk every bridge, that's Eulerian. Walk every town, that's Hamiltonian. Different rules for different journeys. Just trace the paths with your finger. The map tells you where you can go."



She often explained more complex ideas using her map. "Sometimes a bridge only goes one way," she'd say, drawing a tiny arrow on a new sketch. "That's a *directed edge*. Most bridges, though, you can cross both ways, so they're *undirected*." A *path*, she'd explain, was just a sequence of towns connected by bridges, like following directions from your house to the library. If a path led you back to where you started, that was a *cycle*. "Some maps are like a tree," she'd point out, sketching a branching pattern. "You can get everywhere, but there's only one way to get from one place to another without going in a circle. No cycles at all." She'd even talk about *bipartite graphs*, explaining, "Imagine a town where all the houses are either red or blue, and red houses only have bridges to blue houses, and blue houses only to red. That's a bipartite graph." She saw these patterns everywhere, in subway lines, in how friends connected, in the routes delivery trucks took.

Wander grew up in a small village woven together by dozens of bridges. Her family had been the village's "bridge-walkers" for generations. Every morning, the cranes in her family would walk each bridge, carefully recording which ones were safe to cross and which needed repairs. It was a methodical, important job that taught Wander to see the world as a series of connected points.

When she was twenty-two, she walked to DiscreteQuest, a place where young minds gathered to explore the fundamental patterns of the world. The mentor, a wise, ancient owl, looked at her with piercing eyes. "What is graph theory?" the mentor asked.



Wander didn't hesitate. She pulled out her small, folded map. "Vertices and edges," she said, her finger already tracing a path. "Walk every edge or walk every vertex — different rules for different problems. It's the structure of network problems, how everything connects."

The mentor nodded slowly. "You are appointed," she said.

Wander smiled, her eyes on her map. "It is not hard," she insisted. "It is just *vertices* and *edges* and *walk-the-paths*."

Listen along + meet more of the cast at:



<https://spark-and-anvil.com/cast/discretequest/wander>

Coil the Self-Reference



Coil was a small snail-tween, her shell a beautiful spiral-coil. Its intricate curves embodied mathematical sequences, a living lesson in patterns. She moved with a slow, deliberate grace, her warm-amber-and-cream body a soft glow against the polished floors of DiscreteQuest. Coil was fond of self-referential patterns, and her own shell was the most perfect example.

Her shell was her signature feature. Each turn of the coil related to the previous turn by a precise, recursive rule. It wasn't just a pretty design; it *was* Fibonacci. It *was* factorial growth. Coil's very anatomy demonstrated discrete patterns in their purest form.



This connection was essential for her role at DiscreteQuest. Coil embodied **recursion** — the idea of defining something in terms of itself, always with a clear starting point. Her shell *was* the pattern. When she taught the Fibonacci sequence, she would trace a delicate antenna along a specific region of her shell. The spiral here followed the rule: $F(n) = F(n-1) + F(n-2)$. For factorial growth, she would point to a different, rapidly expanding section. Her shell was a constant, tangible demonstration of recursion.

"Recursion isn't hard," Coil would often say, her voice soft but clear. "It's just *self-reference plus a base case*." She would pause, letting the words settle. "You define the basic part directly. Then you define everything else using those smaller, already-defined parts." She would tap her shell gently. "The shell on my back grows by the same recursive rule every single turn."

She found recursion everywhere. It wasn't an elite concept, she insisted, but a fundamental way things grew and changed. She taught the core scaffolds of understanding it:



"Every recursion needs a **BASE CASE**," she'd explain, holding up one antenna. "That's your starting point. Without it, you'd just go on forever, trying to define something that never actually begins." She would shudder slightly at the thought of infinite recursion, a concept that seemed to genuinely bother her.

"Then, every recursion has a **RECURSIVE CASE**," she continued, raising her other antenna. "That's the rule. It defines the current step in terms of smaller, previous steps."

"And the most important part?" Coil would lean forward, her eyes bright. "You have to **trust the recursion**. Just assume those smaller cases are already solved. Then combine them to solve the current one."

She loved pointing out how the Fibonacci sequence appeared in nature. "Look at sunflower seeds," she'd urge her students, projecting an image onto the wall. "Or the scales on a pinecone. They all follow the Fibonacci pattern. Just like my shell."



Factorials, she'd warn, grew fast. "Really fast," she'd emphasize. "N-factorial grows quicker than any simple polynomial. It's amazing how quickly possibilities multiply when you arrange things in a sequence."

Sometimes, for more advanced students, she'd mention that recursion was like mathematical induction in disguise. "Mathematical induction proves recursive properties," she'd say, a small, knowing smile on her face. It was a deeper connection, a quiet nod to the elegant logic underlying both.

She even connected it to other apps. "Think about ScienceForge Sample's data-collection," she'd offer. "Both iterative and recursive methods build cumulatively. They stack information, one piece after another, or one piece *on top of* another."

Coil's understanding of patterns ran deep, a legacy passed down through generations. She grew up in a small village nestled among smooth, river-worn stones. Her family had been the village's shell-carvers for as long as anyone could remember. They were the snails who etched decorative spiral patterns onto the village's stone bowls, each curve carefully mirroring the one before it. They understood the rhythm of growth, the way a pattern could repeat and expand.



When she was twenty-two, a respectable age for a snail-tween, Coil made the long, slow journey to DiscreteQuest. The mentor, a wise old crustacean with eyes like polished obsidian, had looked at her with a piercing gaze. "What is recursion?" the mentor had asked, cutting straight to the heart of the matter.

Coil hadn't hesitated. "Self-reference plus a base case," she'd replied, her voice steady. "Define the basic part. Define everything else from smaller cases." She'd then done something no other candidate had done. She'd turned slightly, offering her shell for inspection. "My shell *is* the pattern."

The mentor had studied her, then her shell, for a long moment. A slow smile had spread across the crustacean's face. "You are appointed," they had said.

Coil often reflected on that moment. "It is not hard," she'd tell herself, or anyone who asked. "It is just *base case* + *smaller case* = *current case*. My shell is the demonstration." And with every slow, deliberate movement, with every turn of her magnificent spiral, she continued to prove it.

Listen along + meet more of the cast at:



<https://spark-and-anvil.com/cast/discretequest/coil>

Prime the Indivisible



Prime was a small hedgehog, no bigger than a teacup. Her fur, a warm mix of brown and cream, seemed to glow in the late afternoon sun. She sat quietly by the edge of the clearing, her steady eyes watching a line of ants march across a fallen leaf. Most hedgehogs had sharp, uniform quills, a bristly armor against the world. But Prime was different. Her spines were soft, like chunky-cartoon bristles, and they grew in distinct, carefully arranged tufts.

Each tuft held a specific number of spines. There were two-spine tufts, neat little pairs that stood straight up. Nearby, a trio of spines formed a perfect triangle. Farther back, five spines clustered together, then seven, then eleven, then thirteen. Prime often ran a paw over them, counting under her breath, a small smile playing on her lips. She was fond of counting by divisors.



Her spines were her signature feature, a living lesson. You would never find a tuft of four spines on Prime. Four could be broken into two groups of two, after all. Nor would you see a tuft of six spines, which could be split into two groups of three, or three groups of two. Eight-spine tufts were also absent. Prime's anatomy showed the world how some numbers simply refused to be broken down.

"What are you counting, Prime?" asked a curious voice. It was Pip, a young squirrel, always full of questions. He hopped closer, tilting his head.

Prime looked up, her smile widening. "My spines," she said simply. "They show how numbers work. See this tuft?" She gently tapped a two-spine cluster. "This is a **prime** number. It's bigger than one, and you can only divide it by one, or by itself. You can't break two into smaller, equal groups without leftovers."



Pip frowned, thinking. "Like trying to share two nuts with three friends?"

"Exactly," Prime confirmed. "Now, look here." She pointed to an empty space where a four-spine tuft *might* have been. "I don't have four spines in a tuft. Four is a *composite* number. You can divide four by two, right? It breaks into two equal groups. Numbers like that, that can be broken into smaller factors besides just one and themselves, aren't prime."

"So, primes are like the unbreakable building blocks?" Pip asked, catching on.

"They are," Prime said, nodding. "They're the multiplicative atoms. Every number bigger than one is either prime, or it can be built from a unique set of primes. That's called *prime factorization*." She picked up a handful of small acorns. "Imagine you have twelve acorns. You could arrange them in two rows of six, or three rows of four. But if you break them down to their smallest, unbreakable parts, you always get the same thing: two, two, and three." She carefully arranged the acorns into groups of two, then two again, then three, showing how they combined to make twelve. "Twelve equals two times two times three. No other set of primes will make twelve."



Pip's eyes widened. "That's neat. So your spines show the numbers that *don't* break."

"Precisely," Prime said. "My spines come in prime counts because primes don't break, just like my spines don't bundle into smaller equal groups." She never made number theory sound difficult or only for certain people. For Prime, it was simply how the world was built.

Her understanding of numbers went deep. She could look at a clock and explain *modular arithmetic*. "If it's ten o'clock," she might say, "and you add three hours, it's one o'clock, not thirteen. You wrap around at twelve. Ten plus three, *modulo twelve*, is one." She saw these patterns everywhere: in calendars, in the way seeds spiraled in a sunflower, even in the complex codes used by the CipherForge Lattice, a place far away that relied on the difficulty of breaking apart very large prime numbers.

Prime had grown up in a small village, a place where her family had been the coin-weighers. They were the hedgehogs who tested metal coins for purity. Pure metals had distinctive, unchangeable weights, much like primes had their own distinctive, unbreakable nature. This early life taught her to look for the fundamental, the uncorrupted.



When she was twenty-two, Prime walked to DiscreteQuest, the ancient school where mentors were chosen. The head mentor, a wise old owl with spectacles perched on his beak, asked her one question. "What is number theory?"

Prime didn't hesitate. "Primes and factorization and modular arithmetic," she answered. "Primes are the multiplicative atoms. My spines come in prime counts because primes don't break."

The owl simply nodded. "You are appointed."

Prime believed it wasn't hard at all. It was just seeing that primes are atomic, and that every integer factors uniquely. Her spines, she often thought, demonstrated it perfectly.

Listen along + meet more of the cast at:



<https://spark-and-anvil.com/cast/discretequest/prime>

About Spark & Anvil

DiscreteQuest is one of 140 educational iOS apps from Spark & Anvil — a 501(c)(3) public charity making free, ad-free, tracking-free learning apps for ages 9-14.

Every app uses distributed-narrative methodology: named recurring characters embody curricular concepts. The cast you just met appears in the matching app, in mentor scaffolding, in puzzle solutions, in celebration moments. Reading the chapters first means meeting old friends when you open the app.

Visit **spark-and-anvil.com** to download apps, browse the full portfolio, or [donate](#) to support the work.