Discrete Math for Computer Science Collaborative Instructional Framework

The following Collaborative Instructional Framework is meant to serve as a guide for teachers and districts as they organize the curriculum for the school year. Unlike traditional pacing guides, the instructional framework consists of clusters of standards that are meant to be adapted to various schools and contexts. Writers of the instructional framework used research on students' learning, expertise of North Carolina mathematics educators, and content experts in mathematics to create and order clusters of standards that are taught together. While there is a suggested order for teaching the clusters, we recognize that schools differ in their contexts and may wish to switch the order around. In those cases, we have given guidance regarding alternative clusterings; however, we note when certain clusters involve understandings and skills that depend on another and should be taught in a certain order.

The Collaborative Instructional Framework was created over a five-month period, beginning in July 2019. Twenty-eight mathematics teachers, district leaders, and university faculty who had served on the Standards Writing and Review teams for 4th level courses or for NC Math 1 –3 worked together to a) read research about pacing guides, student learning progressions, and standards, b) determine the best clusterings based upon research, when possible, and c) wrote this draft of the framework. The team surveyed educators across the state to provide feedback on the instructional framework. As a team, the members read each piece of feedback gathered, discussed the feedback in relation to standards from previous and subsequent courses, research says is best for math learning, and determined how to address the feedback to improve the instructional framework. The members of the Fourth Course Framework Team include: Lauren Baucom, Margaret Borden, Chad Broome, Stefanie Buckner, Alicia Conklin, Arren Duggen, Dr. Cyndi Edgington, Charles Hall, Emily Hare, Maria Hernandez, Michael Hoyes, Patrick Kosal, Hema Lalwani, Dr. Katie Mawhinney, Dr. Allison McCulloch, Emily Myers, Christina Pennington, Todd Rackowitz, Martha Ray, Audrea Saunders, Dr. Catherine Schwartz, Gayle Scott, Chase Tuttle, Jennifer Williams, Carmen Wilson, Dr. Travis Weiland, Dr. Holt Wilson, and Bill Worley. These mathematics professionals represent the four main regions of NC as well as urban, rural, and charter schools. Special thanks to Joseph Reaper and Lisa Ashe from NC DPI for providing guidance and checking for consistency among the framework and DPI resource documents.

Standards for Mathematical Practice

1. Make sense of problems and persevere in solving them.
2. Reason abstractly and quantitatively.
3. Construct viable arguments and critique the reasoning of others.
4. Model with mathematics.
5. Use appropriate tools strategically.
6. Attend to precision.
7. Look for and make use of structure.
8. Look for and express regularity in repeated reasoning.

The Standards for Mathematical Practice are critical ways of acting and communicating in classrooms that should be instilled in students throughout the school year. Whether students are learning to reason proportionally or statistically, they should be obliged to make sense of the problems posed (SMP1) and create a mathematical solution that can contribute to their peers' and their own learning (SMP3). When solving a problem, such as which company is the cheapest when comparing the prices of t- shirts, students should be able to create a viable argument for their choice, with mathematical evidence to defend their solution (SMP3). Students should be able to move among various representations, reasoning quantitatively with symbols (SMP2), and create models of both every day and mathematical situations they encounter (SMP4). Teachers should provide opportunities for students to reason with a variety of tools (SMP5), including technologies that are specific to mathematics (e.g., calculators, Desmos, GeoGebra, etc.). Attending to precision (SMP6) is a practice in which students attempt to present clear arguments, definitions, and meanings for symbols as they explain their reasoning to others. Finding patterns and structure is crucial throughout the standards as students attempt to mathematize complex problem situations (SMP7). Finally, students should attempt to find regularity in repeated reasoning (SMP8), as with this repetition they are able to generalize their findings from one instance to multiple instances.

Course Overview

Unit Name	Duration
Building Mathematical Community	2 - 3 days
Combinatorics	1 - 2 weeks
Graph Theory	3 - 4 weeks
Matrices	2 - 3 weeks
Logic	2 weeks
Set Theory	2 weeks
Number Theory	2 weeks
Recursion	2 - 3 weeks

Unit Name: Building Classroom Community
Duration: Approximately 3 days
Content: It is recommended that the first week of the school year be spent engaging students with open-ended mathematics problems designed to support the students' growth mindset. This first week is also an opportune time for setting up the classroom expectations and norms for collaborating with classmates and participating in whole class discussions.
Mathematical Practices: 1. Make sense of problems and persevere in solving them 2. Reason abstractly and quantitatively 3. Construct viable arguments and critique the reasoning of others 4. Model with mathematics 5. Use appropriate tools strategically 6. Attend to precision 7. Look for and make use of structure. 8. Look for and express regularity in repeated reasoning.
What is the Mathematics? ● Develop mathematicians with positive attitudes about their ability to do mathematics by: ○ Creating opportunities to develop an appreciation for mistakes ○ Seeing mistakes as opportunities to learn ○ Teaching students to take responsibility for their learning ● Develop mathematicians who respect others by: ○ Demonstrating acceptance, appreciation, and curiosity for different ideas and approaches ○ Establishing procedures and norms for productive mathematical discourse ○ Consider other solution paths ● Develop mathematicians with a mindset for problem solving by: ○ Encouraging student authority and autonomy when problem solving ○ Emphasizing questioning, understanding, and reasoning about math, not just doing math for the correct answer ○ Asking follow-up questions when students are both right and wrong ○ Allowing students to engage in productive struggle and moving them along by questioning, not telling
Important Considerations: For success, significant time should be spent setting up the classroom. This includes: ● Developing classroom norms for communication (ex: non-verbal signals, listening and speaking expectations, talk moves for math discussions)

- Developing math routines
- Setting various expectations for the structure of the math block (ex: expectations for whole class instruction, cooperative learning, independent learning, effective integration of technology, etc.)
- Math discourse needs explicit modeling and practice. This includes students:
  - Sharing their thinking
  - Actively listening to the ideas of others
  - Connecting to others' ideas
  - Asking questions to clarify understanding

Global Perspectives:

Resources (Open Access):

- See https://www.youcubed.org/ for suggested activities on building classroom community

Unit Name: Combinatorics
Duration: 1-2 weeks
Standards (Content): **DCS.SP.1 Apply combinatorics concepts to solve problems.** ● **DCS.SP.1.1** Implement the Fundamental Counting Principle to solve problems. ● **DCS.SP.1.2** Implement procedures to calculate a permutation or combination.
Mathematical Practices: 1. Make sense of problems and persevere in solving them **2. Reason abstractly and quantitatively** 3. Construct viable arguments and critique the reasoning of others 4. Model with mathematics 5. Use appropriate tools strategically 6. Attend to precision 7. Look for and make use of structure. **8. Look for and express regularity in repeated reasoning.**
What is the Mathematics? ● Students expand their understanding of counting possibilities of outcomes. It is important to develop the rationale for why these counting procedures are useful by having the students begin by **creating a list** of possibilities to demonstrate the difficulty in ensuring an exhaustive list. Students can also create **tree diagrams** to visualize the possible outcomes which they are familiar with from Math 2. {How many 3-digit lottery tickets can be created using only the numbers 1,2, and 3?}, {How many ways are there to get exactly 2 heads when flipping a coin 3 times?} ● It is important to include scenarios that require the use of multiple counting methods. {A team of students to represent a school will be chosen from each of the four classes: Freshmen, Sophomore, Junior, and Seniors. There are 3 Freshmen candidates, 4 Sophomore candidates, 5 Junior candidates, and 10 senior candidates. There will be 2 students from each class chosen and there will be 1 senior captain and 1 senior co-captain of the team. How many teams can be created?} **Vocabulary** ● Counting ● Combination ● Permutation ● Factorial ● Fundamental Counting Principle
Important Considerations: ● This unit follows the Building Classroom Community because it is accessible to all students. There are numerous applications across mathematics and computer science,

including graph theory, probability, coding, and cryptography. In addition, the study of combinatorics provides many opportunities to engage in the standards of mathematical practice

- This unit lays a foundation for the course and introduces ideas that should be used as tools in future units. For example, combinations are useful when determining the number of edges in a complete graph with *n* vertices in the Graph Theory Unit. In the Set Theory unit, combinatorics can be used to determine the total number of possible elements in a set. Recursion and recursive thinking can be introduced when exploring the Fundamental Counting Principle, i.e., 5! = 5*4!

Global Perspectives:

- Combinatorics is an essential part of Computer Science. Graph Theory, the study of objects and connections, is a branch of combinatorics. Combinatorics are useful for enumerating all possibilities, an essential part of brute force algorithms like Kruskal's and Prim's algorithms in graph theory. Combinatorial reasoning is an essential part of the study of algorithms, analyzing data structures, and optimization problems.

Resources (Open Access)

Wendy's Letter Project (Walking in Mathland Blog)

Open Text: Unit One

Combinatorics Tasks

Unit Name: Graph Theory

Duration: 3-4 weeks

Standards (Content):

DCS.GT.1 Understand graph theory to model relationships and solve problems.

- **DCS.GT.1.1** Explain real world situations using a vertex-edge graph, adjacency matrix, and vertex table.
- **DCS.GT.1.2** Test graphs and digraphs for Euler paths, Euler circuits, Hamiltonian paths, or Hamiltonian circuits.
- **DCS.GT 1.3** Interpret a complete digraph to determine rank.

DCS.GT.2 Apply graph theory to solve problems.

- **DCS.GT.2.1** Implement critical path analysis algorithms to determine the minimum project time.
- **DCS.GT.2.2** Implement the brute force method, the nearest-neighbor algorithm, and the cheapest-link algorithm to find solutions to a Traveling Salesperson Problem.
- **DCS.GT.2.3** Implement vertex-coloring techniques to solve problems
- **DCS.GT.2.4** Implement Kruskal and Prim's algorithms to determine the weight of the minimum spanning tree of a connected graph.

Mathematical Practices:

1. Make sense of problems and persevere in solving them
2. Reason abstractly and quantitatively
3. Construct viable arguments and critique the reasoning of others
4. **Model with mathematics**
5. Use appropriate tools strategically
6. Attend to precision
7. Look for and make use of structure.
8. Look for and express regularity in repeated reasoning.

What is the Mathematics?

Students will learn to use vertex-edge graphs to create visual models for real world situations. The intent of this unit is to introduce students to central concepts of graph theory and use those ideas to model and solve a range of problems. It provides a context for algorithmic thinking, a focus of Discrete Math for Computer Science. This unit allows students multiple opportunities to explore and "play" with mathematics. It is important that the tasks that are chosen support student exploration. Social networks provide relevant connections for students.

Vocabulary

- Planar
- Connected
- Euler circuits and paths

• Tournament graph	• Complete	• Hamiltonian Circuits and paths
• Edges	• Trees	• Bipartite graphs
• Degree/valence	• Digraphs	• Chromatic number
• Paths	• Minimum spanning tree	• Kruskal's Algorithm
• Circuits		• Prim's Algorithm
• Cycle	• Markov Chains	• Traveling Salesperson Problem
• Adjacent	• Critical Path	• Nearest Neighbor Algorithm
• Loops	• Earliest start time/Latest start time	• Cheapest Link Algorithm
• Node	• Adjacency matrix	• Brute force method

Important Considerations:

- DCS.GT.1.1 should be taught first in order for students to develop the concepts of vertices and edges. Unlike the geometries that most students have experienced before, it is important to make explicit that the "intersection" of two or more edges does not imply a vertex. Social Networking is a useful context for understanding graph theory and for seeing its relevance.
- GT 1.2, 1.3, and 2.2 should be clustered together. While Euler and Hamiltonian circuits and paths may be taught separately in any order, 1.3 and 2.2 are specific applications of Hamiltonian circuits.
- GT1.1 and GT1.3 (determining rank from a digraph) refer to tournament graphs, where each vertex represents a competitor and each directed edge represents the result of the competition between the connected competitors. For example, if competitor A defeated competitor B, the edge would flow from competitor A ro competitor B. From the tournament graph, students should be able to interpret the ranking of the competitors in relation to each other.
- Applications of graph theory can be completed in any order after the foundation has been built: GT 2.1, 2.3, 2.4 (critical path analysis, vertex-coloring, and minimum spanning trees)
- There is a strong connection between graph theory and matrices. Graphs can be

represented as a matrix, where one row and one column of a matrix represents one vertex of the graph. If two vertices, say A and B, are connected, it is represented by a 1 in the Ath row, Bth column of the matrix. If A and B are not connected, it is represented by a 0.

- Brute Force algorithms (eg. Kruskal, Prim) provide a connection to combinatorics

Global Perspectives:
Graph theory provides an introduction to algorithmic thinking which should be viewed as one of the overall foci of Discrete Mathematics for Computer Science. While the unit focuses on some specific graph theory terminology and ideas, algorithmic thinking and decision making are metacognitive concepts that should be made explicit to students. Computer scientists use graph theory and the accompanying thinking skills in their work daily. Graphs are central in understanding and solving problems related to planning and scheduling, modeling networks, data organization, and to analyze algorithms.

Resources (Open Access):
Seven Bridges of Konigsberg ("Math is Fun")
TedEd Video: Seven Bridges of Konigsberg

"Four Is Not Enough": https://www.quantamagazine.org/the-numbers-and-geometry-behind-a-math-coloring-puzzle-20180618/

Open Text: Unit 4

NCTM Graph Creator

Overview of TSP/Intro to TSP

https://www.sciencealert.com/amateur-solves-decades-old-maths-problem-about-colours-that-can-never-touch-hadwiger-nelson-problem

Additional Resources (Closed Access):
Core Plus Book One, Unit Four "Vertex-Edge Graphs"
Core Plus Book Two, Unit 6 "Network Optimization"

"Vertex-Edge Graphs: An Essential Topic in High School Geometry" (requires NCTM membership to access)

Unit Name: Matrices
Duration: 2-3 weeks
Standards (Content): **DCS.N.1 Apply operations with matrices and vectors.** ● **DCS.N.1.1** Implement procedures of addition, subtraction, multiplication, and scalar multiplication on matrices. ● **DCS.N.1.2** Implement procedures of addition, subtraction, and scalar multiplication on vectors ● **DCS.N.1.3** Implement procedures to find the inverse of a matrix. **DCS.N.2 Understand matrices to solve problems.** ● **DCS.N.2.1** Organize data into matrices to solve problems ● **DCS.N.2.2** Interpret solutions found using matrix operations including Leslie Models & Markov Chains, in context. ● **DCS.N.2.3** Represent a system of linear equations as a matrix equation. ● **DCS.N.2.4** Use inverse matrices to solve a system of linear equations with technology.
Mathematical Practices: 1. Make sense of problems and persevere in solving them 2. Reason abstractly and quantitatively 3. Construct viable arguments and critique the reasoning of others **4. Model with mathematics** 5. Use appropriate tools strategically **6. Attend to precision** **7. Look for and make use of structure.** 8. Look for and express regularity in repeated reasoning.

What is the Mathematics?

- In this unit, students learn to use matrices to organize information and solve problems from a context. Labeling the rows and columns of a matrix and referring to particular elements of a matrix support students in maintaining the connection between a matrix and context.
- Students should explore what properties of operations for matrices are similar and different from the real numbers. Remaining grounded in a context is important for students to develop an understanding of matrix multiplication. Examining the number of dimensions of two matrices and developing a rule to see if their product exists should follow from an understanding of the dimensions represented by rows and columns of the matrices. Make explicit for students that unlike multiplication of real numbers, matrix multiplication is noncommutative.
- To determine if a matrix has an inverse, students will find its determinant to see if its value is nonzero. Students should use appropriate technology to calculate determinants of square matrices larger than a 3x3.
- Using inverse matrices, students will learn to represent and solve a system of equations using matrix equations.
- Matrices are used to represent vectors and model phenomena in the physical, natural, and social sciences. For example, Markov Chains use transition matrices to represent probabilities of transitioning among a finite number of states and predict stable state vectors after a sequence of possible events. Leslie models are useful for modeling population growth and also use transition matrices to represent the proportion of members of a certain age group that survive and enter the next age group.

Vocabulary/Concepts/Skills

- Row
- Column
- Dimensions
- Elements
- Determinant
- Inverse Matrix
- Identity Matrix
- Scalar
- Transpose
- Vector
- Matrix Operations
- Leslie Model
- Markov Chain

Important Considerations:

- DCS.N.2.1 and DCS.N.1.1 introduce the usefulness of matrices for organizing data and operating with data to solve problems and should come first
- For DCS.N.1.3, students should build conceptual understanding of inverse matrices for 2x2 matrices and build procedural fluency with technology for larger matrices

• DCS.N.2.2, DCS.N.2.3, & DCS.N.2.4 are uses of matrices to solve problems. Students learn to use matrix equations to solve systems of linear equations, complementing approaches students learned in Grade 8 and NC Math 1. Markov Chains and the Leslie model use matrix equations to solve problems in a variety of contexts in the physical, life, and social sciences • Students should add and subtract vectors written in the form $ai + bj$ as well as with matrices. Students are not expected to perform other vector operations.
Global Perspectives: Matrices are essential tools for computer scientists. Matrices are used to organize and retrieve data, model networks, detecting errors in data transmission, and in algorithms like Google's original page rank algorithm, compression techniques, and mining data. Matrices are also central to computer graphics, animation, and face recognition technologies.
Resources (Open Access): Cryptography: Harry Potter The Leslie Model Article: Train Delays and Markov Chains Matrix Multiplication Tasks http://news.mit.edu/2013/explained-matrices-1206
Additional Resources (Closed Access) Core Plus Book Two, Unit 2 "Matrix Methods"

Unit Name: Logic
Duration: Approximately 2 weeks
Standards (Content): **DCS.L.1 Evaluate mathematical logic to model and solve problems.** ● **DCS.L.1.1** Construct truth tables that encode the truth and falsity of two or more statements. ● **DCS.L.1.2** Critique logical arguments. ● **DCS.L.1.3** Check 1s and 0s to determine whether a statement is true or false using Boolean logic circuits. ● **DCS.L.1.4** Judge whether two statements are logically equivalent using truth tables.
Mathematical Practices: 1. Make sense of problems and persevere in solving them 2. Reason abstractly and quantitatively **3. Construct viable arguments and critique the reasoning of others** 4. Model with mathematics 5. Use appropriate tools strategically **6. Attend to precision** 7. Look for and make use of structure. 8. Look for and express regularity in repeated reasoning.
What is the Mathematics? Mathematical logic is the foundation of many areas of mathematics and computer science. In this unit, students investigate the truthfulness of a proposition by examining its assumptions and justifications. Truth tables are useful for determining the validity of an argument by enumerating all possible combinations of its premises (up to three in this course – p, q, and r). They examine conditional (if-then) and biconditional (if and only if) statements, evaluate more complex arguments using the laws of syllogism and detachment, and demonstrate the logical equivalence of two statements by showing they have identical truth tables. They also learn and use Boolean Algebra, a branch of algebra where variables have only two possible values, 0 (false) and 1 (true), and operations such as conjunction (and), disjunction (or), and negation (not) relating them. Boolean logic circuits represent functions that map one or more Boolean variables as inputs through a gate that represents a Boolean operator to a single output. In addition to having direct applications to electric circuits, this representation is also useful in verifying the truthfulness of statements. **Vocabulary** ● Argument ● Premise ● Conclusion ● Validity ● Tautology ● Contradiction ● Inverse ● Converse

• Contrapositive	• Conditional	• Negation	• Biconditional
• Proposition	• Consequent	• Fallacy	• If and only if
• Truth	• False	• Invalid	• Law of Detachment
• Inference	• Deduction	• Boolean gates	• Law of Syllogism
• Assertion	• Exclusion & inclusion	• Disjunction	• Implication

Important Considerations:

- You can support students in learning to construct truth tables (DCS.L.1.1) by beginning with a context that has meaning for them and evaluating simple claims. Abstraction is an important goal in this unit but should not come at the expense of context.
- As students encounter and critique more complex arguments (DCS.L.1.2), they should recognize the converse, inverse, and contrapositives of conditional statements, biconditional statements.
- (DCS.L.1.4) Students can examine logical arguments to identify contradictions and tautologies as well as verify that two statements are logically equivalent by showing that their truth tables are identical.
- Boolean logic circuits (DCS.L.1.3) are another representation used to examine the truthfulness of statements. In this representation, logical operators (and, or, not) are represented by different shapes known as gates. Each assumption is represented by a Boolean variable with values of 0 and 1 as an input that a gate maps to a single Boolean variable output.

Global Perspectives:

Mathematical logic is the underlying principle of coding. Boolean logic gates and circuits encode statements and create conditions in electrical engineering and computers. Understanding Boolean logic creates more powerful search criteria when using search engines such as Google, Bing, and DuckDuckGo. Evaluating logical arguments is an increasingly important skill for being an informed participant in a democracy. For example, logic is useful in determining the truthfulness of statements made by political candidates, pundits, and political broadcasters.

Resources (Open Access):

Online Stanford Textbook

Open Text: Unit Three

Stanford Intro to Logic Course for Secondary Education
Resources (Closed Access): Core Plus Book 3, Unit One “Reasoning Strategies” (for a basic intro to reasoning and logic) NCTM: “A Truth Table on the Island of Truthtellers and Liars” (access with NCTM membership)

Unit Name: Set Theory
Duration: Approximately 2 weeks
Standards (Content): **DCS.N.3 Understand set theory to solve problems.** • **DCS.N.3.1** Recognize sets, subsets, and proper subsets. • **DCS.N.3.2** Implement set operations to find unions, intersections, complements and set differences with multiple sets. • **DCS.N.3.3** Represent properties and relationships among sets using Venn diagrams. • **DCS.N.3.4** Interpret Venn diagrams to solve problems. **DCS.N. 4 Understand statements related to number theory and set theory.** • **DCS.N.4.3** Conclude that sets are equal using the properties of set operations.
Mathematical Practices: 1. Make sense of problems and persevere in solving them **2. Reason abstractly and quantitatively** 3. Construct viable arguments and critique the reasoning of others **4. Model with mathematics** 5. Use appropriate tools strategically 6. Attend to precision 7. Look for and make use of structure. 8. Look for and express regularity in repeated reasoning.
What is the Mathematics? Students have studied some ideas from set theory in the context of data and probability with Venn diagram representations in 7th grade and NC Math 2 (NC.M2.S-CP.6-8). This unit builds upon these prior understandings and experiences to scaffold a more formal understanding of sets and set operations. Students will use Venn diagrams that represent 2 or 3 sets to understand the meanings of sets, subsets, and proper subsets, their properties, relationships among them, and operations on them (unions, intersections, complements, and differences). Along with Venn diagrams, they use set properties to investigate if sets are equivalent. **Vocabulary** • Venn Diagram • Union (∪) • Intersection (∩) • Set Difference (−) • Empty Set { } • Universal Set U • Null Set ∅ • Proper Subset (⊂) • Complement S^C • Element ($\in$ or $\notin$) • Subset (⊆)

Important Considerations:

- Students have had formal and informal experiences with sets prior to this course. Students should recognize subsets and proper subsets (DMC.N.3.1) of finite and infinite sets. For example, students should recognize the real numbers and the imaginary numbers proper subsets of the complex numbers and the natural numbers as a subset of the set of positive integers greater than 0.
- Venn diagrams are useful representations of sets, and students should use them to investigate set properties, operations on two or more sets, and the relationships among them (DMC.N.3.2-4).
- Students should operate on sets in contexts that are relevant and of interest to them (). They should explore set properties and relationships and use them to solve problems (DCS.N.3.3) and show that two sets are equal (DCS.N.4.3)

Global Perspectives:

Set theory is a fundamental idea of computer science. Computer scientists use set theory as a tool to reason formally about computation the objects of computation. Sets are the underlying mathematical tool for many common computing functions like matching and searching. Set theory is used to design data structures that are efficient and conserve storage space, network security systems, and is the basis for programming languages.

Resources (Open Access)

Open Text: Unit 0

Unit Name: Number Theory
Duration: 1-2 weeks
Standards (Content): **DCS.N. 4 Understand statements related to number theory and set theory.** ● **DCS.N.4.1** Use the Euclidean Algorithm to identify the greatest common factor and least common multiple. ● **DCS.N.4.2** Use the Fundamental Theorem of Arithmetic to solve problems. ● **DCS.N.4.4** Explain theorems related to greatest common factor, least common multiple, even numbers, odd numbers, prime numbers, and composite numbers.
Mathematical Practices: 1. Make sense of problems and persevere in solving them 2. **Reason abstractly and quantitatively** 3. **Construct viable arguments and critique the reasoning of others** 4. Model with mathematics 5. Use appropriate tools strategically 6. Attend to precision 7. Look for and make use of structure. 8. Look for and express regularity in repeated reasoning.
What is the Mathematics? This unit builds on students' experiences with numbers since the elementary grades, their understanding of sets, and knowledge of real number operations to explore some aspects of number theory relevant to computer science and engage in algorithmic thinking. IN this course, students will explore methods for determining the GCF and LCM, including the Euclidean algorithm. They will also investigate and generalize properties of numbers from patterns to classify sets of numbers (e.g., even, odd, prime, and composite) and the set of numbers that result from arithmetic operations on them (e.g. the sum of two odd numbers is even, the product of two odd numbers is odd. Using the Fundamental Theorem of Arithmetic, students will solve problems by representing any integer as a unique product of prime numbers. **Vocabulary:** ● Fundamental Theorem of Arithmetic ● Euclidean Algorithm

Important Considerations:

- This unit is not intended to be a deep dive into Number Theory. Rather, it is intended to be an introduction to ideas central to number theory ideas that are useful in computer science.
- Students should investigate the Euclidean algorithm and using snap cubes or an area model with graph paper to show the process of finding the greatest common divisor of two numbers and justifying the result is the greatest (DCS.N.4.1)
- The Fundamental Theorem of Arithmetic (DCS.N.4.2) shows that every integer can be uniquely expressed as the product of prime numbers. Students should use this result and skills involving exponents to solve problems involving multiplication, division, even and odd, prime composite, perfect squares, and perfect cubes.
- Students should explore and explain algorithms for finding, and theorems about, the greatest common factors, least common multiples, even and odd numbers, and prime and composite numbers. They should develop solid arguments using words, figures, or algebra to explain why each works (DCS.N.4.4)

Global Perspectives:

Computer scientists use Number Theory when developing new algorithms and data structures. Results from number theory offer insight into underlying relationships among numbers which often simplifies algorithms and makes them more efficient. Number theory plays a fundamental role in cryptography, from encryption algorithms to cryptocurrencies.

Resources (Open Access):

Open Text: Unit Five

Unit Name: Recursively Defined Functions
Duration: 2-3 weeks
Standards (Content): **DCS.F.1 Apply recursively-defined relationships to solve problems** ● **DCS.F.1.1** Implement procedures to find the nth term in an arithmetic or geometric sequences using spreadsheets. ● **DCS.F.1.2** Represent the sum of a sequence using sigma notation. ● **DCS.F.1.3** Implement procedures to find the sum of a finite sequence. ● **DCS.F.1.4** Implement procedures to find the sum of an infinite sequence and determine if the series converges or diverges. ● **DCS.F.1.5** Interpret the solutions to arithmetic and geometric sequences and series problems, in context.
Mathematical Practices: 1. Make sense of problems and persevere in solving them 2. Reason abstractly and quantitatively 3. Construct viable arguments and critique the reasoning of others 4. Model with mathematics 5. Use appropriate tools strategically 6. Attend to precision 7. Look for and make use of structure. 8. Look for and express regularity in repeated reasoning.
What is the Mathematics? In this unit, students extend their knowledge of recursively defined arithmetic and geometric sequences from NC Math 1 using technology (e.g. spreadsheets) to find the nth term of an arithmetic and geometric sequence. They will develop an understanding of methods and formulas for computing sums of finite arithmetic and geometric series, use them to solve problems, and represent series in sigma notation. They will determine if an infinite geometric series is convergent or divergent and find the sum if it converges. **Vocabulary** ● Arithmetic sequence ● Arithmetic series ● Geometric sequence ● Geometric series ● Closed form/Explicit form ● Convergent ● Divergent

Important Considerations:

- DCS.F.1.1 extends students' understanding of arithmetic and geometric sequences as linear and exponential functions. In this course, students learn to use technology to find the nth term of sequences. Technology can assist students in making connections between arithmetic and geometric growth by comparing graphical, tabular, and algebraic representations of a sequence. These representations are useful in developing intuition about convergent and divergent series (DCS.F.1.4)
- Students should know that a series is the sum of a number of terms of a sequence and understand the reasoning behind common formulas for arithmetic and geometric series (DCS.F.1.3). For example, recognizing the average value of pairs of terms in an arithmetic sequence leads to the formula for the sum of an arithmetic sequence. Another line of reasoning is that one can form a sequence of partial sums and the nth term of this sequence is equal to the sum of the first n terms of the series. For example, the series 1 + 3 + 5 + … + (2n + 1) is equivalent to the sequence of partial sums 1, 4, 9, ...n2. Using this fact, the sum of the first n terms of the series 1 + 3 + 5 + … + (2n +1) = n2
- Just as NC Math 1 students begin with Next-Now notation but move to more formal ways of notating recursively defined sequences, students in this course should learn a more formal notation for series. Assist students in recognizing underlying structure and expressing it using sigma notation (DCS.F.1.2). Sigma notation is new for students, and it is to highlight and discuss when it is appropriate to use 0 or 1 as initial value. With sigma notation, make explicit that the index counts the number of items in the set and not the number of intervals between the first and last element. It is also important to make the distinction between indexing the term number in a series and the natural numbers themselves.
- Students should have opportunities to develop understandings of the ideas in this unit as well as solve problems and interpret solutions in meaningful contexts (DCS.F.1.5).

Global Perspectives:
In computer science, recursion is a concept where solutions to a problem rely on solutions to smaller instances of the same problem. Computer scientists use recursion in when creating models and simulations, when solving complex computational problems, defining dynamic data structures, developing algorithms, and generating computer graphics. Recursive thinking involves decomposing a large task into smaller processes to simplify calculations and improve efficiency.

Resources (Open Access):
Handshake Problem/Handshake Problem Reflections

Sequences Puzzle

Open Text: Unit 3
Resources (Closed Access): Core Plus Book Three, Unit 7 "Recursion and Iteration"