

Discrete Mathematics *Standards of Learning* - 2023 Overview of Revisions

This overview includes a summary of the content embedded in four content strands.

Logical Reasoning

- Use reasoning to develop and apply logical arguments[†]
- Apply logic and proof techniques in the construction of a sound argument[†]
- Apply Boolean algebra to represent and analyze the function of logical gates and circuits[†]
- Use mathematical induction to prove formulas and mathematical statements

Set and Number Theory

- Identify and use the properties of sets and set operations[†]
- Apply the formulas of combinatorics[†]
- Use Pascal's Triangle to analyze numerical patterns and relationships

Graph Theory

- Represent problems using vertex-edge graphs. The concepts of degree, connectedness, paths, planarity, and directed graphs will be analyzed.[†]
- Solve problems through analysis and application of circuits, cycles, Euler paths, Euler circuits, Hamilton paths, and Hamilton circuits. Optimal solutions will be determined using existing algorithms and student-created algorithms.[†]
- Apply graphs to conflict-resolution problems, such as map coloring, scheduling, matching, and optimization[†]
- Recognize and apply algorithms to solve configuration, conflict-resolution, and sorting problems
- Use algorithms to schedule tasks to determine a minimum project time

Computational Methods

- Describe and apply sorting and searching algorithms used in processing and communicating information[†]
- Use recursive processes[†]
- Identify and apply cryptographic methods
- Analyze the limitations of algorithms and their contextual relationships in computing.

[†] Content intended for a one-semester course only.

Comparison of Discrete Mathematics *Standards of Learning* – 2016 to 2023

The following standards outline the content of a one-year course in Discrete Mathematics.

If a one-semester course is desired, the standards with a dagger (†) would apply.

2016 <i>Standards of Learning</i> Essential Knowledge and Skills (EKS) Graphs	2023 <i>Standards of Learning</i> Knowledge and Skills (KS) Logical Reasoning (LR)
<p>[Moved from DM.9†]</p>	<p>DM.LR.1† The student will use reasoning to develop and apply logical arguments.</p> <ul style="list-style-type: none"> a) Use Venn diagrams to codify and solve logic problems. b) Express logical statements in symbolic form. c) Represent a conditional statement as its converse, inverse, and contrapositive. d) Describe how symbolic logic can be used to map the processes of computer applications. e) Construct a truth table to display all possible input combinations and their outputs. f) Identify the rules of inference and model basic logical statements including De Morgan’s Law. g) Apply logical reasoning to model contextual situations and make decisions.
<p>[New Expectation]</p>	<p>DM.LR.2† The student will apply logic and proof techniques in the construction of a sound argument.</p> <ul style="list-style-type: none"> a) Apply informal logical reasoning to contextual problems including (e.g., predicting the behavior of software, solving puzzles).

<p align="center">2016 Standards of Learning Essential Knowledge and Skills (EKS) Graphs</p>	<p align="center">2023 Standards of Learning Knowledge and Skills (KS) Logical Reasoning (LR)</p>
	<ul style="list-style-type: none"> b) Outline the basic structure of a proof technique (e.g., direct proof, proof by contradiction, induction). c) Deduce the best type of proof for a given problem. d) Use the rules of inference to construct direct proofs and proofs by contradiction. e) Construct induction proofs involving summations and inequalities. f) Use a truth table to prove the logical equivalence of statements.
<p>[Moved from DM.9[†]; New Expectations]</p>	<p>DM.LR.3[†] The student will apply Boolean algebra to represent and analyze the function of logical gates and circuits.</p> <ul style="list-style-type: none"> a) Explain basic properties of Boolean algebra: duality, complements, and standard forms. b) Represent verbal statements as Boolean expressions. c) Apply Boolean algebra to prove identities and simplify expressions. d) Generate truth tables that encode the truth and falsity of two or more statements. e) Explain the operation of discrete logic gates. f) Describe the relationship between Boolean algebra and electronic circuits. g) Analyze a combinational network using Boolean expressions. h) Design simple combinational networks that use NAND (AND followed by NOT), NOR (OR followed by NOT), and XOR (exclusive-OR) gates.

<p align="center">2016 Standards of Learning Essential Knowledge and Skills (EKS) Graphs</p>	<p align="center">2023 Standards of Learning Knowledge and Skills (KS) Logical Reasoning (LR)</p>
<p>[Moved from Mathematical Analysis]</p>	<p>DM.LR.4 The student will use mathematical induction to prove formulas and mathematical statements.</p> <ul style="list-style-type: none"> a) Compare and contrast inductive and deductive reasoning. b) Explain the relationship between weak and strong induction. c) Construct induction proofs involving a divisibility argument. d) Prove the Binomial Theorem through mathematical induction.

<p align="center">2016 Standards of Learning Essential Knowledge and Skills (EKS) Set and Number Theory</p>	<p align="center">2023 Standards of Learning Knowledge and Skills (KS) Set and Number Theory (SNT)</p>
<p>[Moved from DM.9[†]; New Expectations]</p>	<p>DM.SNT.1[†] The student will identify and use the properties of sets and set operations.</p> <ul style="list-style-type: none"> a) Compare and contrast sets, relations, and functions. b) Express relationships between sets using Venn diagrams. c) Describe a set using set-builder notation. d) Construct new sets using the set operations intersection, union, difference, and complement. e) Identify the laws of set theory (e.g., associative, commutative, distributive, De Morgan’s Law). f) Use the principle of inclusion and exclusion to determine the size of a set. g) Use the properties of set operations to prove set equality.

<p align="center">2016 Standards of Learning Essential Knowledge and Skills (EKS) Set and Number Theory</p>	<p align="center">2023 Standards of Learning Knowledge and Skills (KS) Set and Number Theory (SNT)</p>
<p>[Moved from DM.13a-d]</p>	<p>DM.SNT.2† The student will apply the formulas of combinatorics.</p> <ul style="list-style-type: none"> a) Create a tree diagram to represent relationships between independent events. b) Use the Fundamental (Basic) Counting Principle to determine the number of possible outcomes of an event. c) Determine the number of combinations possible when subsets of r elements are selected from a set of n elements without regard to order. d) Determine the number of permutations possible when r objects selected from n objects are ordered. e) Use the pigeonhole principle to solve packing problems to facilitate proofs. f) Construct a proof by induction using principles of combinatorics.
<p>[Moved from Mathematical Analysis]</p>	<p>DM.SNT.3 The student will use Pascal’s Triangle to analyze numerical patterns and relationships.</p> <ul style="list-style-type: none"> a) Construct Pascal’s Triangle. b) Expand binomials having positive integral exponents, using the Binomial Theorem and Pascal’s Triangle. c) Compare the binomial coefficient to the calculation of combinations. d) Identify the Fibonacci numbers within Pascal’s Triangle.

<p style="text-align: center;">2016 Standards of Learning Essential Knowledge and Skills (EKS) Graph Theory (GT)</p>	<p style="text-align: center;">2023 Standards of Learning Knowledge and Skills (KS) Graph Theory (GT)</p>
<p>DM.1† The student will model problems, using vertex-edge graphs. The concepts of valence, connectedness, paths, planarity, and directed graphs will be investigated.</p> <ul style="list-style-type: none"> ● Determine the valence of each vertex in a graph. ● Use graphs to model situations in which the vertices represent objects, and edges (drawn between vertices) represent a particular relationship between objects. ● Represent the vertices and edges of a graph as an adjacency matrix, and use the matrix to solve problems. ● Investigate and describe valence and connectedness. ● Determine whether a graph is planar or nonplanar. ● Use directed graphs (digraphs) to represent situations with restrictions in traversal possibilities. 	<p>DM.GT.1† The student will represent problems using vertex-edge graphs. The concepts of degree, connectedness, paths, planarity, and directed graphs will be analyzed.</p> <ol style="list-style-type: none"> a) Illustrate the basic terminology of graph theory (e.g., vertex, edge, graph, degree of a vertex). b) Use graphs to map situations in which the vertices represent objects, and edges represent a particular relationship between objects. c) Identify and describe degree and connectedness. d) Determine whether a graph is planar or nonplanar. e) Analyze the relationship between faces, edges, and vertices using Euler’s formula ($F = E - V + 2$). f) Use directed graphs (digraphs) to represent situations with restrictions in traversal possibilities. g) Determine when graphs are trees.
<p>DM.2† The student will solve problems through investigation and application of circuits, cycles, Euler paths, Euler Circuits, Hamilton paths, and Hamilton circuits. Optimal solutions will be sought using existing algorithms and student-created algorithms.</p> <ul style="list-style-type: none"> ● Determine whether a graph has an Euler circuit or path, and determine it, if it exists. ● Determine whether a graph has a Hamilton circuit or path, and determine it, if it exists. ● Count the number of Hamilton circuits for a complete graph with n vertices. 	<p>DM.GT.2† The student will solve problems through analysis and application of circuits, cycles, Euler paths, Euler circuits, Hamilton paths, and Hamilton circuits. Optimal solutions will be determined using existing algorithms and student-created algorithms.</p> <ol style="list-style-type: none"> a) Determine whether a graph has an Euler circuit or path, and determine the circuit or path, if it exists. b) Determine whether a graph has a Hamilton circuit or path, and determine the circuit or path, if it exists. c) Count the number of Hamilton circuits for a complete graph with n vertices.

<p style="text-align: center;">2016 Standards of Learning Essential Knowledge and Skills (EKS) Graph Theory (GT)</p>	<p style="text-align: center;">2023 Standards of Learning Knowledge and Skills (KS) Graph Theory (GT)</p>
<ul style="list-style-type: none"> ● Use an Euler circuit algorithm to solve optimization problems. 	<p>d) Use an Euler circuit algorithm to solve optimization problems.</p>
<p>DM.3[†] The student will apply graphs to conflict-resolution problems, such as map coloring, scheduling, matching, and optimization.</p> <ul style="list-style-type: none"> ● Model projects consisting of several subtasks, using a graph. ● Use graphs to resolve conflicts that arise in scheduling. ● Determine the chromatic number of a graph 	<p>DM.GT.3[†] The student will apply graphs to conflict-resolution problems, such as graph coloring, scheduling, matching, and optimization.</p> <ul style="list-style-type: none"> a) Model projects consisting of several subtasks, using a graph. b) Use graphs to resolve conflicts that arise in scheduling. c) Use graph coloring to determine the chromatic number of a graph.
<p>DM.4 The student will apply algorithms relating to trees, networks, and paths. Appropriate technology will be used to determine the number of possible solutions and generate solutions when a feasible number exists.</p> <ul style="list-style-type: none"> ● Use Kruskal’s algorithm to determine the shortest spanning tree of a connected graph. ● Use Prim’s algorithm to determine the shortest spanning tree of a connected graph. ● Use Dijkstra’s algorithm to determine the shortest spanning tree of a connected graph 	<p>DM.GT.4 The student will recognize and apply algorithms to solve configuration, conflict-resolution, and sorting problems.</p> <ul style="list-style-type: none"> a) Recognize algorithms such as nearest neighbor, brute force, and cheapest link as they apply to graphs. b) Use Kruskal’s algorithm to determine the shortest spanning tree of a connected graph. c) Use Prim’s algorithm to determine the shortest spanning tree of a connected graph. d) Use Dijkstra’s algorithm to determine the shortest spanning tree of a connected graph.
<p>[Moved from DM.10]</p>	<p>DM.GT.5 The student will use algorithms to schedule tasks to determine a minimum project time.</p> <ul style="list-style-type: none"> a) Specify in a digraph the order in which tests are to be performed. b) Identify the critical path to determine the earliest completion time (minimum project time). c) Use the list-processing algorithm to determine an optimal schedule.

2016 Standards of Learning Essential Knowledge and Skills (EKS) Graph Theory (GT)	2023 Standards of Learning Knowledge and Skills (KS) Graph Theory (GT)
	d) Create and test scheduling algorithms.

2016 Standards of Learning Essential Knowledge and Skills (EKS) Election Theory and Fair Division	2023 Standards of Learning Knowledge and Skills (KS)
DM.5[†] The student will analyze and describe the issue of fair division in discrete and continuous cases. <ul style="list-style-type: none"> ● Investigate and describe situations involving discrete division (e.g., estate division). ● Use an algorithm for fair division for a group of indivisible objects. ● Investigate and describe situations involving continuous division of an infinitely divisible set (e.g., cake cutting). ● Use an algorithm for fair division of an infinitely divisible set. 	[DELETED]
DM.6[†] The student will investigate and describe weighted voting and the results of various election methods. These may include approval and preference voting as well as plurality, majority, runoff, sequential runoff, Borda count, and Condorcet winners. <ul style="list-style-type: none"> ● Determine in how many different ways a voter can rank choices. ● Investigate and describe the following voting procedures: <ul style="list-style-type: none"> ○ weighted voting; ○ plurality; ○ majority; ○ sequential (winners runoff); ○ sequential (losers are eliminated); 	[DELETED]

<p align="center">2016 Standards of Learning Essential Knowledge and Skills (EKS) Election Theory and Fair Division</p>	<p align="center">2023 Standards of Learning Knowledge and Skills (KS)</p>
<ul style="list-style-type: none"> ○ Borda count; and ○ Condorcet winner. ● Compare and contrast different voting procedures. ● Describe the possible effects of approval voting, insincere and sincere voting, a preference schedule, and strategic voting on the election outcome. 	
<p>DM.7 The student will identify apportionment inconsistencies that apply to issues such as salary caps in sports and allocation of representatives to Congress. Historical and current methods will be compared.</p> <ul style="list-style-type: none"> ● Compare and contrast the Hamilton and Jefferson methods of political apportionment with the Hill-Huntington method (currently in use in the U.S. House of Representatives) and the Webster-Willcox method. ● Solve allocation problems, using apportionment methods. ● Investigate and describe how salary caps affect apportionment. 	<p>[DELETED]</p>

<p align="center">2016 Standards of Learning Essential Knowledge and Skills (EKS) Computer Mathematics</p>	<p align="center">2023 Standards of Learning Knowledge and Skills (KS) Computational Methods (CM)</p>
<p>DM.8 The student will describe and apply sorting algorithms and coding algorithms used in sorting, processing, and communicating information.</p> <ul style="list-style-type: none"> ● Select and apply a sorting algorithm, such as a 	<p>DM.CM.1† The student will describe and apply sorting and searching algorithms used in processing and communicating information.</p>

<p style="text-align: center;">2016 Standards of Learning Essential Knowledge and Skills (EKS) Computer Mathematics</p>	<p style="text-align: center;">2023 Standards of Learning Knowledge and Skills (KS) Computational Methods (CM)</p>
<ul style="list-style-type: none"> o bubble sort; o merge sort; and o network sort. ● Describe and apply a coding algorithm, such as <ul style="list-style-type: none"> o ISBN numbers; o UPC codes; o zip codes; and o banking codes. 	<ul style="list-style-type: none"> a) Select and apply a sorting algorithm, such as a bubble sort, merge sort, or network sort. b) Describe the advantages and disadvantages of various sorting algorithms. c) Analyze the knapsack and bin-packing problems. d) Select and apply search algorithms to analyze problems. e) Determine the average, best-case, and worst-case reasoning for different searches.
<p>[Moved from DM.12a,c,d,e]</p>	<p>DM.CM.2† The student will use recursive processes.</p> <ul style="list-style-type: none"> a) Compare and contrast iterative and recursive processes. b) Use recursive processes to model growth and decay. c) Use recursive processes to create fractals. d) Use recursive processes to generate the Fibonacci sequence. e) Determine if a recursive solution is more efficient than an iterative solution.
<p>[New Expectation]</p>	<p>DM.CM.3 The student will identify and apply cryptographic methods.</p> <ul style="list-style-type: none"> a) Compare and contrast ciphers and codes. b) Describe the evolution of cipher systems. c) Identify the Fundamental Theorem of Arithmetic. d) Describe how the complexity of prime factorization is used in cryptography. e) Describe modular arithmetic in context (e.g., clocks, days of the week, measures of time). f) Analyze the relationship between divisibility and modulus.

<p align="center">2016 Standards of Learning Essential Knowledge and Skills (EKS) Computer Mathematics</p>	<p align="center">2023 Standards of Learning Knowledge and Skills (KS) Computational Methods (CM)</p>
	<p>g) Determine congruence within modular arithmetic.</p> <p>h) Perform operations within modular arithmetic.</p> <p>i) Apply modular arithmetic to problems in context (e.g., cryptography, International Standard Book Number (ISBN), International Bank Account Number (IBAN)).</p>
<p>[New Expectation]</p>	<p>DM.CM.4 The student will analyze the limitations of algorithms and their contextual relationships in computing.</p> <p>a) Discuss maximum complexity of an algorithm using Big O notation.</p> <p>b) Describe Turing machines and how they are used to test the limits of computation.</p> <p>c) Describe the halting problem and explain how it characterizes the fundamental limitations of computation and undecidability.</p> <p>d) Explain the P versus NP problem and defend a justification for equality, inequality, or undecidability.</p> <p>e) Analyze how the equivalence of P- and NP-class problems might impact society.</p>
<p>DM.9[†] The student will select, justify, and apply an appropriate technique to solve a logic problem.</p> <ul style="list-style-type: none"> ● Generate truth tables that encode the truth and falsity of two or more statements. ● Use Venn diagrams to represent set relationships, such as intersection and union. ● Interpret Venn diagrams. ● Use Venn diagrams to codify and solve logic problems. 	<p>[Moved to DM.LR.1[†]; DM.LR.3[†]]</p>

<p align="center">2016 Standards of Learning Essential Knowledge and Skills (EKS) Computer Mathematics</p>	<p align="center">2023 Standards of Learning Knowledge and Skills (KS) Computational Methods (CM)</p>
<ul style="list-style-type: none"> ● Use matrices as arrays of data to solve logic problems. 	

<p align="center">2016 Standards of Learning Essential Knowledge and Skills (EKS) Recursion and Optimization</p>	<p align="center">2023 Standards of Learning Knowledge and Skills (KS)</p>
<p>DM.10 The student will use algorithms to schedule tasks in order to determine a minimum project time. The algorithms will include critical path analysis, the list-processing algorithm, and student-created algorithms.</p> <ul style="list-style-type: none"> ● Specify in a digraph the order in which tests are to be performed. ● Identify the critical path to determine the earliest completion time (minimum project time). ● Use the list-processing algorithm to determine an optimal schedule. ● Create and test scheduling algorithms. 	<p>[Moved to DM.GT.5]</p>
<p>DM.11 The student will solve linear programming problems.</p> <ul style="list-style-type: none"> ● Model practical problems with systems of linear inequalities. ● Identify the feasibility region of a system of linear inequalities with no more than four constraints. ● Identify the coordinates of the corner points of a feasibility region. ● Determine the maximum or minimum value of the system. ● Describe the meaning of the maximum or minimum value in terms of the original problem. 	<p>[Included in AFDA]</p>

<p style="text-align: center;">2016 Standards of Learning Essential Knowledge and Skills (EKS) Recursion and Optimization</p>	<p style="text-align: center;">2023 Standards of Learning Knowledge and Skills (KS)</p>
<p>DM.12 The student will use the recursive process and difference equations with the aid of appropriate technology to generate</p> <p>a) compound interest; b) sequences and series; c) fractals; d) population growth models; and e) the Fibonacci sequence.</p> <ul style="list-style-type: none"> ● Use finite differences and recursion to model compound interest and population growth situations. ● Model arithmetic and geometric sequences and series recursively. ● Compare and contrast the recursive process, and create fractals. ● Compare and contrast the recursive process and the Fibonacci sequence. ● Determine a recursive relationship that generates the Fibonacci sequence. 	<p>[Moved to DM.CM.2[†]; Included in Mathematical Analysis]</p>
<p>DM.13 The student will apply the formulas of combinatorics in the areas of</p> <p>a) the Fundamental (Basic) Counting Principle; b) knapsack and bin-packing problems; c) permutations and combinations; and d) the pigeonhole principle</p> <ul style="list-style-type: none"> ● Determine the number of combinations possible when subsets of r elements are selected from a set of n elements without regard to order. ● Use the Fundamental (Basic) Counting Principle to determine the number of possible outcomes of an event. 	<p>[Moved to DM.SNT.2[†]; DM.CM.1]</p>

<p style="text-align: center;">2016 <i>Standards of Learning</i> Essential Knowledge and Skills (EKS) Recursion and Optimization</p>	<p style="text-align: center;">2023 <i>Standards of Learning</i> Knowledge and Skills (KS)</p>
<ul style="list-style-type: none"> ● Use the knapsack and bin-packing algorithms to solve practical problems. ● Determine the number of permutations possible when r objects selected from n objects are ordered. ● Use the pigeonhole principle to solve packing problems to facilitate proofs. 	

2023 Discrete Mathematics SOL – Summary of Changes

Discrete Mathematics (2016 SOL to 2023 SOL Numbering)	Parameter Changes/Clarifications (2023 SOL)
<ul style="list-style-type: none"> ● DM.1† → DM.GT.1† ● DM.2† → DM.GT.2† ● DM.3† → DM.GT.3† ● DM.4 → DM.GT.4 ● DM.5† → [Deleted] ● DM.6† → [Deleted] ● DM.7 → [Deleted] ● DM.8 → DM.CM.1† ● DM.9† → DM.LR.1; DM.LR.3†; DM.SNT.1† ● [New Expectation] → DM.LR.2† ● [Moved from Mathematical Analysis] → DM.LR.4 ● DM.10 → DM.GT.5 ● DM.11 → [Included in AFDA] ● DM.12a,c,d,e → DM.CM.2† ● DM.12b → [Included in Mathematical Analysis] ● DM.13a,c,d → DM.SNT.2† ● DM.13b → DM.CM.1† ● [Moved from Mathematical Analysis] → DM.SNT.3 ● [New Expectation] → DM.CM.3 ● [New Expectation] → DM.CM.4 	<ul style="list-style-type: none"> ● DM.GT.1† - Analyze the relationship between faces, edges, and vertices using Euler’s formula ($F = E - V + 2$) ● DM.CM.2† - Compare and contrast both iterative and recursive processes

Deletions from Discrete Mathematics (2016 SOL)	Additions to Discrete Mathematics (2023 SOL)
<ul style="list-style-type: none"> ● DM.5[†] - Analyze and describe fair division in discrete and continuous cases [Deleted] ● DM.6[†] - Investigate and describe weighted voting and the results of various election methods [Deleted] ● DM.7 - Identify apportionment inconsistencies [Deleted] ● DM.8 [KS] - Describe and apply a coding algorithm [Deleted] ● DM.9[†] [KS] - Use matrices as arrays of data to solve logic problems [Included in Mathematical Analysis] ● DM.11 – Solve linear programming problems [Included in AFDA] ● DM.12b – Use recursive processes to generate sequences and series [Included in Mathematical Analysis] 	<ul style="list-style-type: none"> ● DM.LR.2[†] - Apply logic and proof techniques in the construction of a sound argument, including DeMorgan’s Law ● DM.LR.3[†] - Apply Boolean Algebra to represent and investigate the function of logical gates and circuits [New Expectation] ● DM.LR.4 - Use Mathematical Induction to prove formulas and mathematical statements [Moved from Mathematical Analysis] ● DM.SNT.1[†] - Investigate the properties of sets, their construction, and set operations [New Expectation] ● DM.SNT.3 - Use Pascal’s Triangle to explore numerical patterns and relationships [Moved from Mathematical Analysis] ● DM.CM.3 - Investigate and apply cryptographic methods [New Expectation] ● DM.CM.4 - Explore the limitations of algorithms and their contextual relationships in computing [New Expectation]

KEY: LG = Logical Reasoning; SNT = Set and Number Theory; GT= Graph Theory; CM = Computational Methods; EKS =Essential Knowledge and Skills (2016); KS = Knowledge and Skills (2023); US = Understanding the Standard; [†] This standard should be included in a semester course.