Discrete Structures

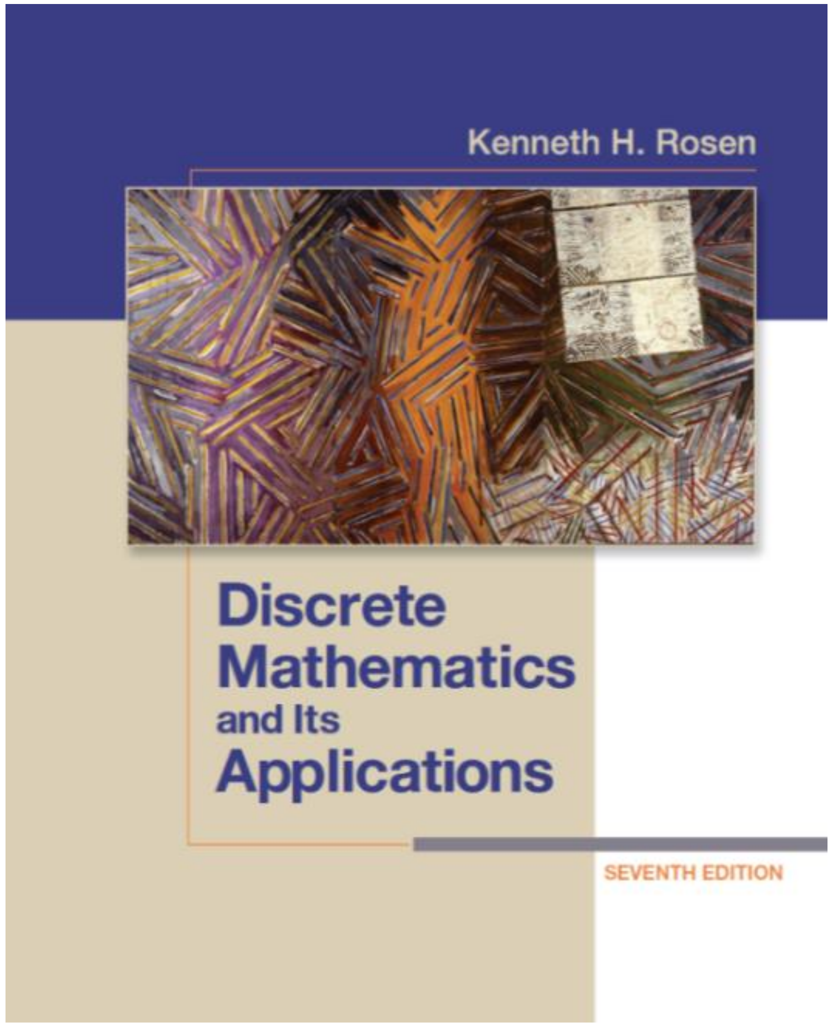
Abdulhakeem Mohammed

(Hakeem)

Textbook

- Kenneth H. Rosen,
  Discrete Mathematics and Its Applications,
  7th edition, McGraw-Hill.

- Also available as PDF.



Course Requirements

Attendance & participation	10%
Homework Assignments	30%
Midterm Exam #1 (tentatively Tuesday, Oct. 10th)	20%
Midterm Exam #2 (tentatively Tuesday, Nov. 14th)	20%
Final Exam (Wed., Dec 13th, 12:45PM - 3:00PM)	20%

Homework and Exams

- **Homework**:
  - ~10 assignments.
  - Will be announced in class and posted on the course website (usually you will have a week to work on it).
  - Unexcused late homework is not accepted.
  - Final HW grade: total # received points / # total possible points.
- **Exams**:
  - 3 exams, including the final exam
  - Closed book, closed notes.
  - Exams will in large part be based on the homework.

Office Hours

- MSB 352 @ Tuesday and Thursday: 3:30pm - 4:30pm

- By appointment.
  - amohamm4@kent.edu
  - Send email to arrange an appointment.
  - Put “Discrete Structures” somewhere in the email subject.

- **Grader**:
  - Betis Baheri
  - **Email :** bbaheri@kent.edu

Grading Scale

Scale:	0%	60%	67%	70%	73%	77%	80%	83%	87%	90%	93%
Grade:	F	D	D+	C-	C	C+	B-	B	B+	A-	A
GPA:	0.00	1.00	1.30	1.70	2.00	2.30	2.70	3.00	3.30	3.70	4.00

Course Site & Contact

- **Course site**
  - http://personal.kent.edu/~amohamm4/ds-f2017/
  - topics and slides
  - Homework
  - Important dates

Tips

Homework… Do it!

- Do it by yourself.
- If you do it in groups… do it small groups.
  - Try it on your own first.
  - Discuss approaches, not solutions.
  - Do not copy.
- Fully explain your answers. Show your work to receive full credit.

General

- Students are required to attend every class.
- Check the course website for upcoming deadlines
- Do the odd numbered problems at the end of each section
  - The solutions are in the end of the book
- Read the Syllabus for more details.
- Have a question!
  - Feel free to interrupt me in class for asking questions.
  - Sign up for Piazza, a Q&A web service where you can ask and answer questions related to the class.

Discrete Mathematics

What is Discrete Mathematics?

Discrete mathematics is the study of mathematical structures that are fundamentally discrete rather than continuous.

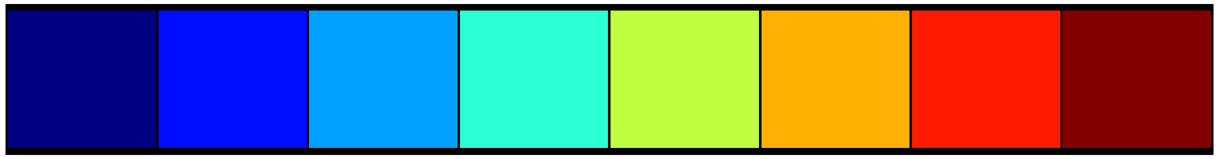
Discrete

- Given any two numbers, there does not exist a third in between.
- Countable

Examples

- Number of students.
- The results of rolling 2 dice:

(can only have the values 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 and 12)

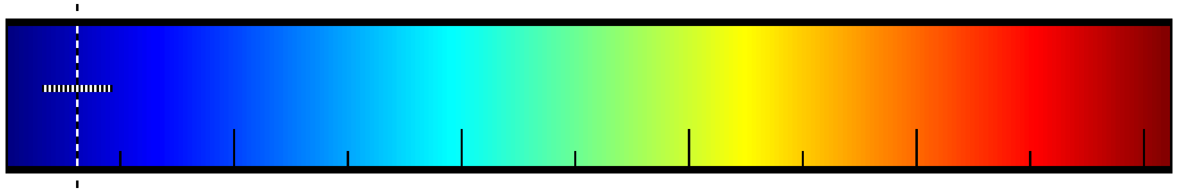


Continuous

- Given any two numbers, can always find numbers in between.
- Measurable

Examples

- Height of a person
- Speed of a car



Topics in Discrete Math

- **Logic** - used in programming, artificial intelligence, circuit design, puzzles.

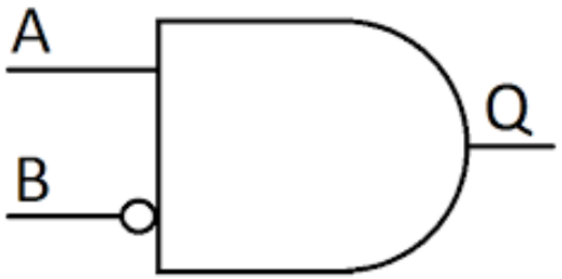
- **Counting** - used in probability, games, analysis of algorithms.

- **Graph theory** - used for computer and wireless networks, data structures, path finding.

- **Number theory** - cryptography

Topic 1. Logic & Proofs

- Logic: propositional logic, first order logic
- Proof: induction, contradiction
- Used in artificial intelligence, programming, circuits, algorithms, and puzzles



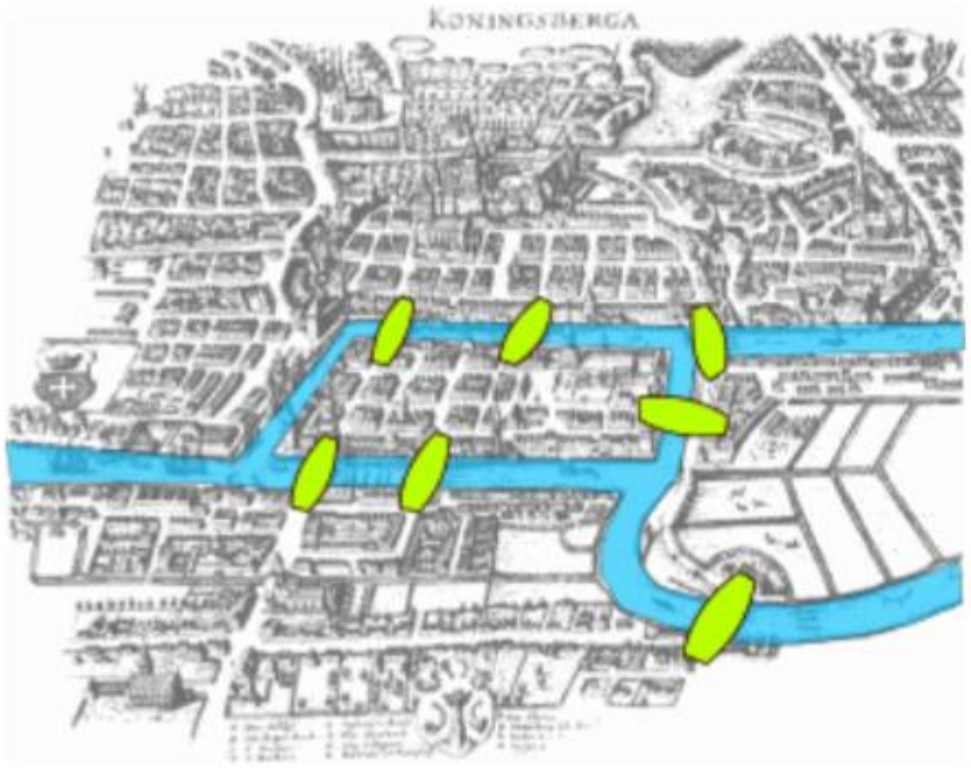
5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

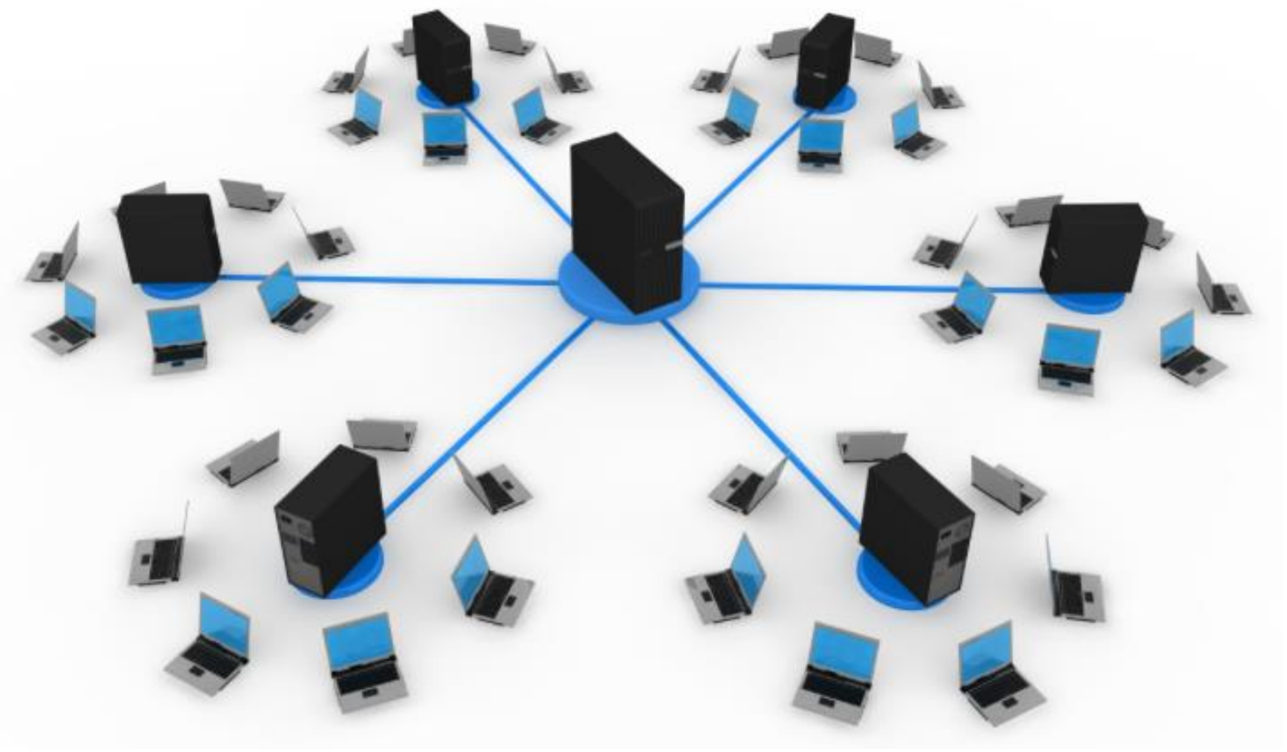
Topic 2. Counting

- How many combinations/permutations exist?
- How many steps are needed to sort *n* numbers?
- Sets
- Combinations, Permutations, Binomial theorem
- Functions
- Pigeonhole principle
- Recursions, generating functions
- Used in probability, algorithms, analysis of algorithms, data structures

Topic 3. Graph Theory

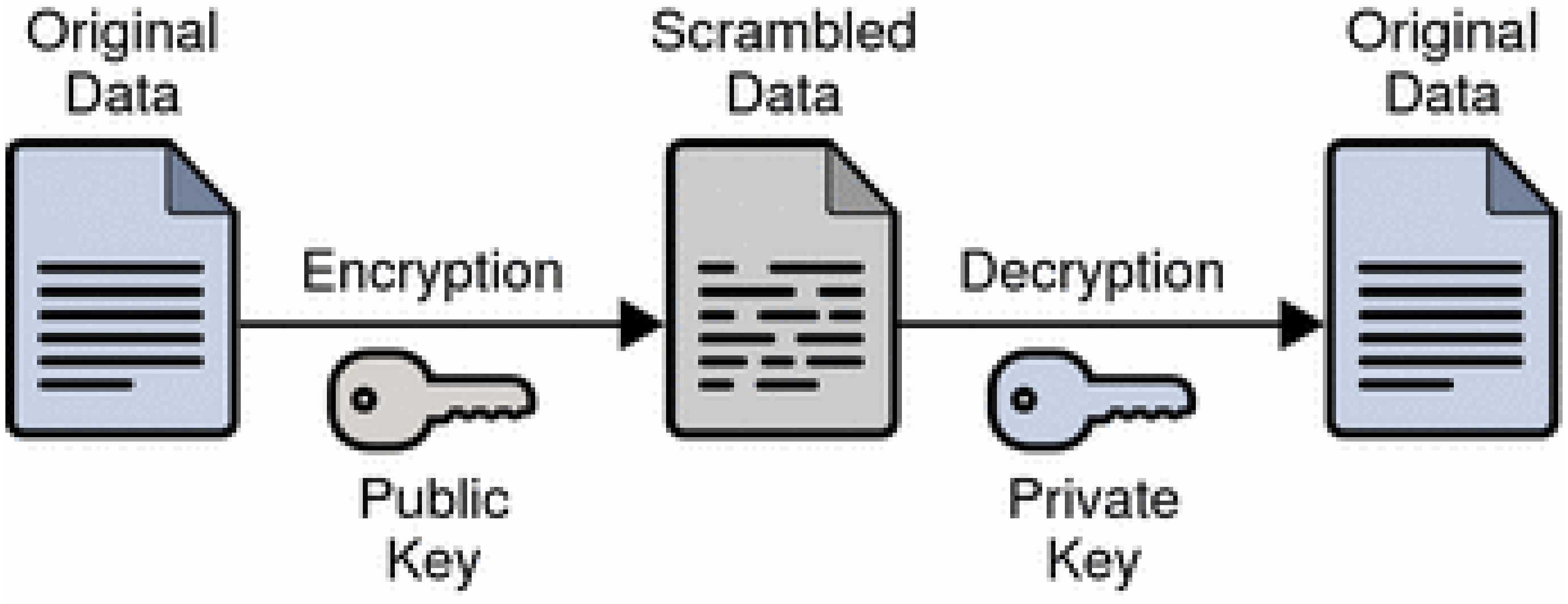
- What's the best path?
- Relations, graphs, trees
- Degree sequence, isomorphism, Eulerian graphs
- Computer networks, circuit design, data structures





Topic 4. Number Theory

- Number sequence
- Euclidean algorithm
- Prime number
- Modular arithmetic
- Cryptography, coding theory, data structures



Why Discrete Math?

- Understand and create mathematical arguments.
- Discrete mathematics provides the mathematical foundations for many computer science courses including:
  - Data structures
  - Algorithms
  - Computer security
  - Operating systems
  - ... And more
- Design efficient computer systems and programs.
  - How fast will the program run?
  - Can it be faster?
  - How do you know it will work?