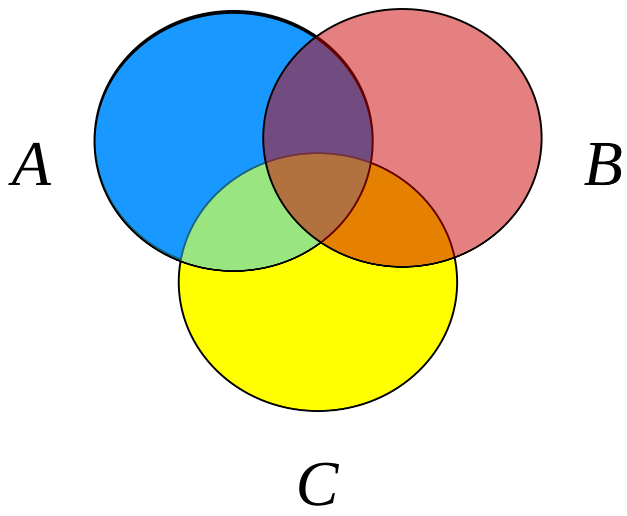
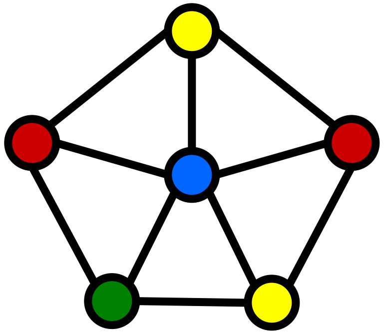
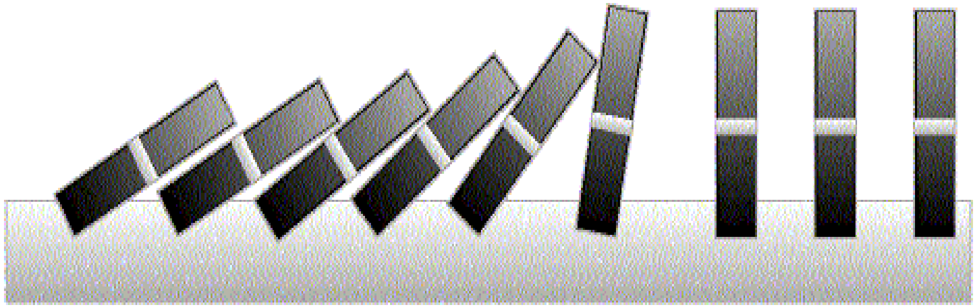
About the Course







Overview

- Course Information and Arrangement
- Topics of the Course
- Lecture Slides will be Uploaded to Blackboard the Week Before*
- Course Objective

Please read the course outline carefully!!!

* Lecture slides are adapted in part from
L. C. Lau's original ones from CUHK

Assessment Scheme

- Assignments 20%
  - Fundamental questions (5%) will be provided every Monday
  - 4 assignments (15%) will be given in this course
- Mid-term Exam 30%
  - Tentatively on 6 November 2020 (Friday)
- Final Exam 50%

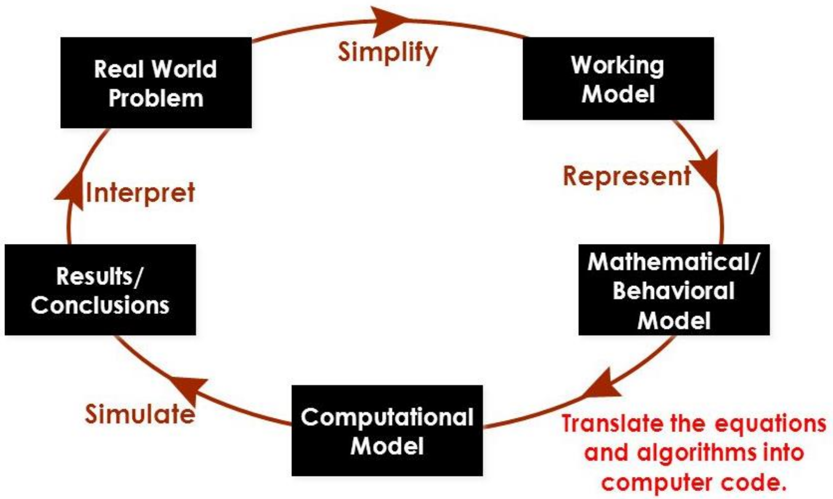
References

- Susanna S. Epp, Discrete Mathematics with Applications (4th Edition), Brooks/Cole, Cengage Learning, 2011.
- Eric Lehman, F. Thomson Leighton, Albert R. Meyer, Mathematics for Computer Science, 2016.
- Richard A. Brualdi, Introductory Combinatorics (5th Edition), Pearson Education, Inc., 2010.
- Kenneth H. Rosen, Discrete Mathematics and Its Applications (7th Edition), McGraw-Hill, 2012.

Problems in computer science

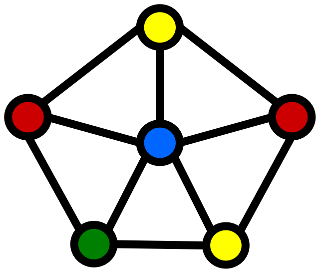
1. Problem modelling
2. Model analysis (feasibility, complexity, etc.)
3. Design algorithms
4. Algorithm analysis (efficiency, bugs, etc)
5. Problem dectection
6. ......

Discrete mathematics deals with all of these



Discrete Mathematics

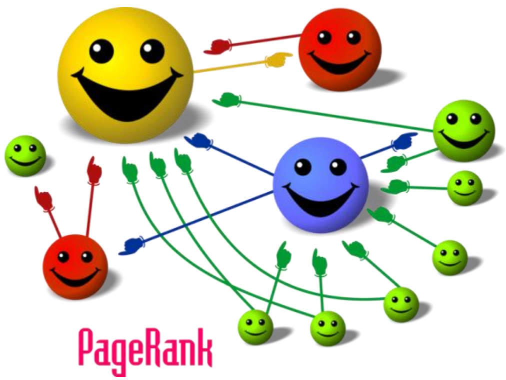
What is discrete mathematics?



discrete mathematics	continuous mathematics
integers	real numbers
graphs	geometric space
induction	calculus
logic	

These two areas are not disjoint, e.g. calculus can be used to solve discrete problems (generating functions).

Discrete Mathematics



Why discrete mathematics?

In computer science we usually deal with finite, discrete objects. For example,

- we cannot store a real number (infinite precision) in a computer but can only store bits (finite precisions).

- we often model a computer network as a graph, and use the knowledge and techniques in dealing with graphs to solve problems in networks.

The problems and the techniques are often different (e.g. induction, recursion).

Logic and Proofs

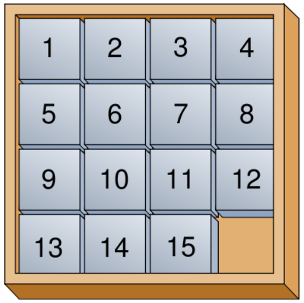
How do computers (and humans) think?

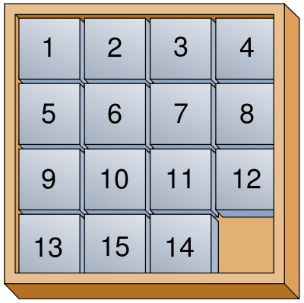
Logic: propositional logic, first order logic

Proof: **induction**, contradiction

$$\forall x \exists y, z \quad x = y + z$$

$$\frac{x_1 + x_2 + \ldots + x_n}{n} \geq \sqrt[n]{x_1 \cdot x_2 \cdots x_n}$$

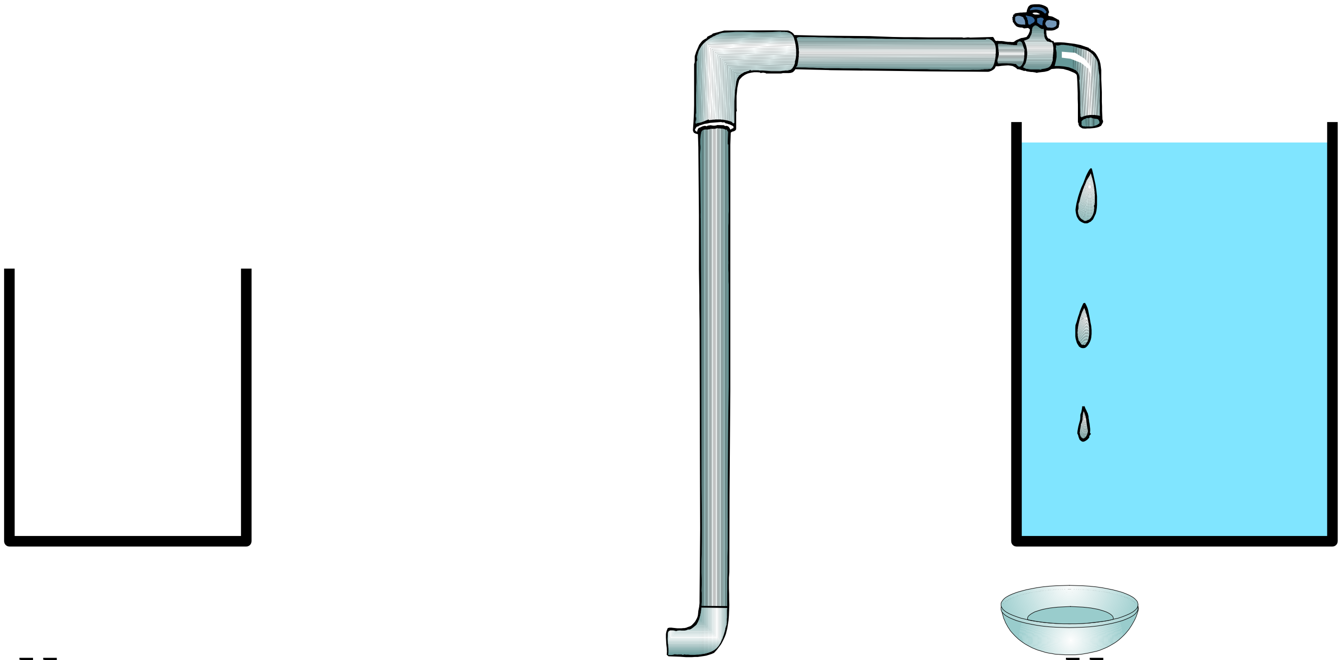




Applications: artificial intelligence, database, circuit, algorithms

Objective: to reason rigorously and learn basic proof techniques (e.g. induction)

Number Theory

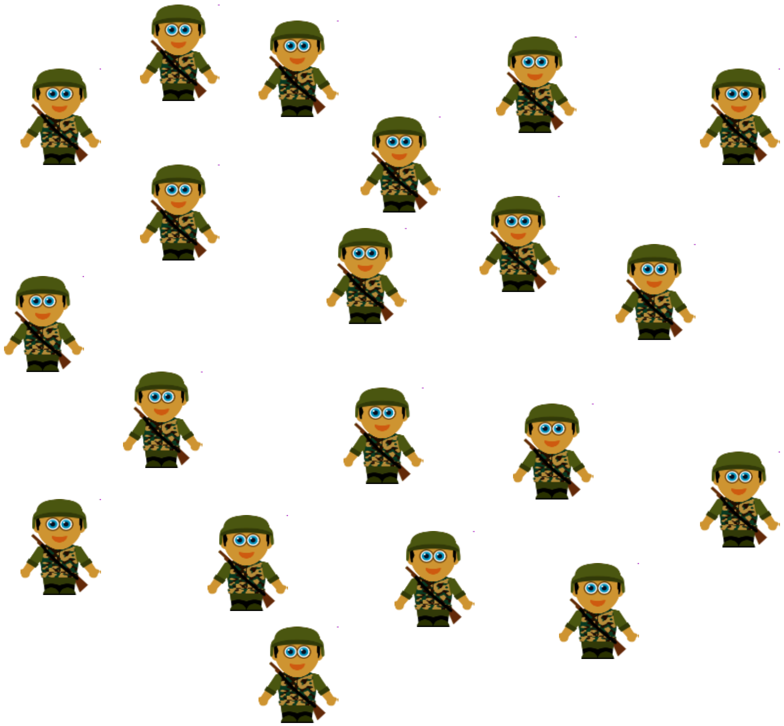


3 Gallon Jug

5 Gallon Jug

How to get 4 gallons?

Number Theory



We have 1073 soldiers.

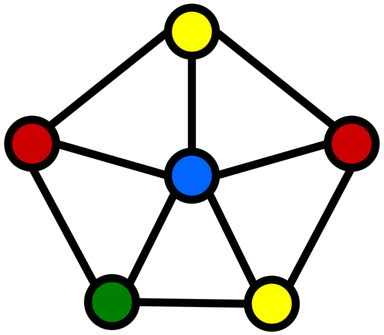
How could he figure it out?!

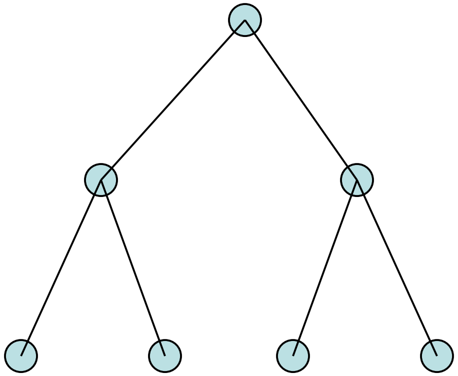
Applications: divisibility, crytography

Objective: to learn elementary number theory and classical results

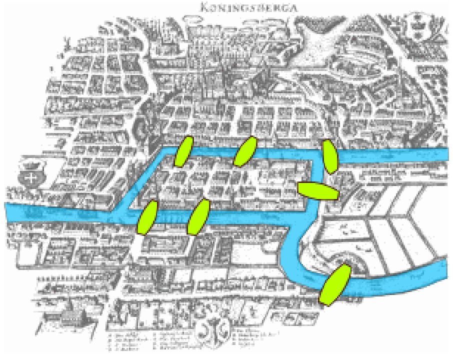
Graph Theory

- Graphs (consist of nodes and edges)
- Degree sequence, Eulerian graphs, isomorphism
- Trees
- Matching
- Coloring





Bridges of Konigsberg: devise a walk through the city that would cross each of those seven bridges once and only once



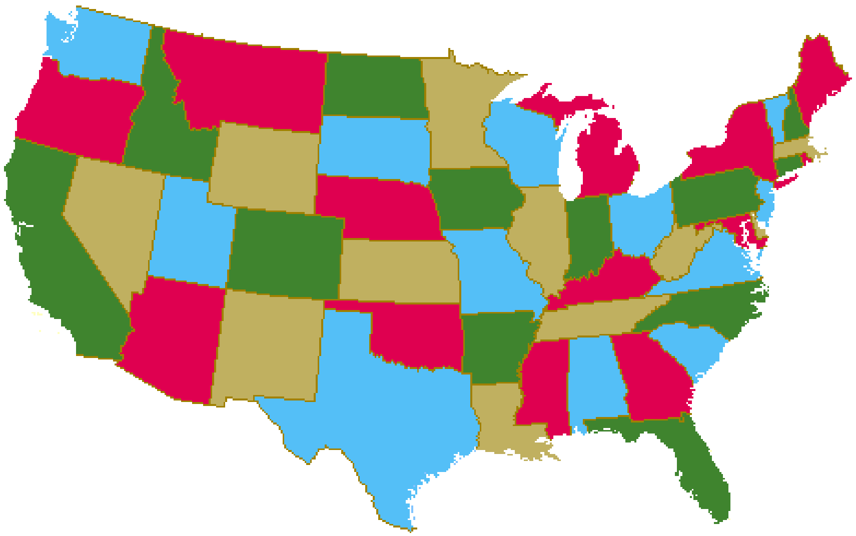
By Bogdan Giuşcă - Public domain (PD),based on the image, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=112920

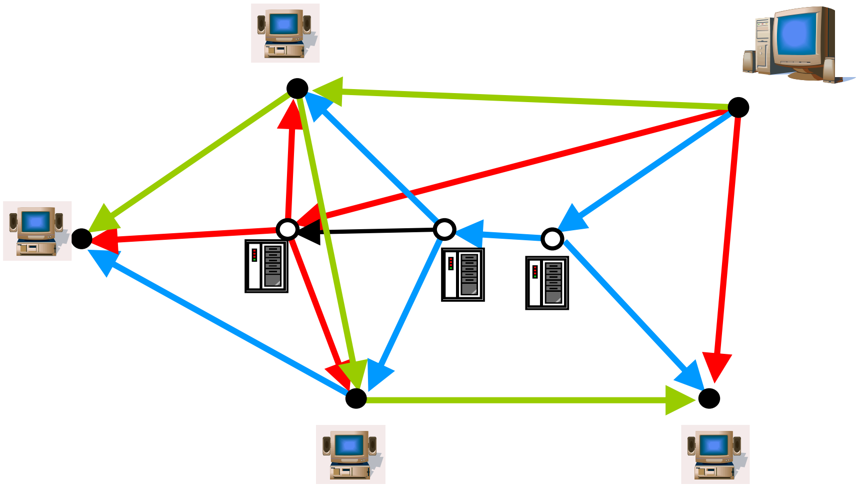
Applications: Computer networks, circuit design, data structures

Graph Theory

How to color a map?

How to schedule exams?



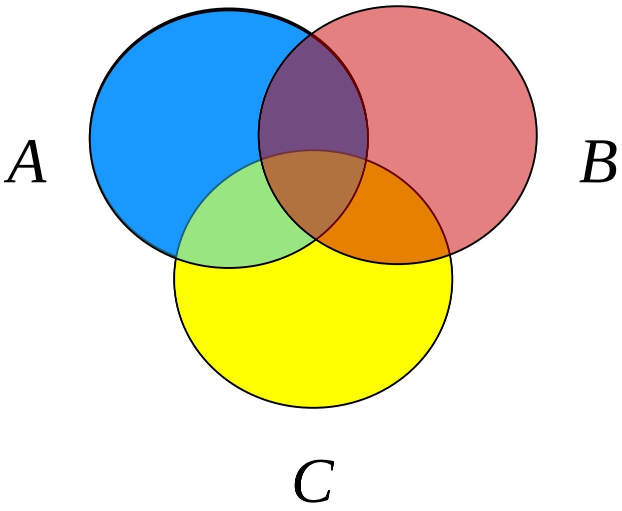


How to send data efficiently?

Objective: to model problems and learn basic concepts and knowledge

Counting

- Sets and Functions
- Combinations, Permutations, inclusion-exclusion
- Counting by mapping, pigeonhole principle
- Recursions



Applications: probability, data structures, algorithms

Objective: to learn basic concepts (set, functions) and fundamental techniques.

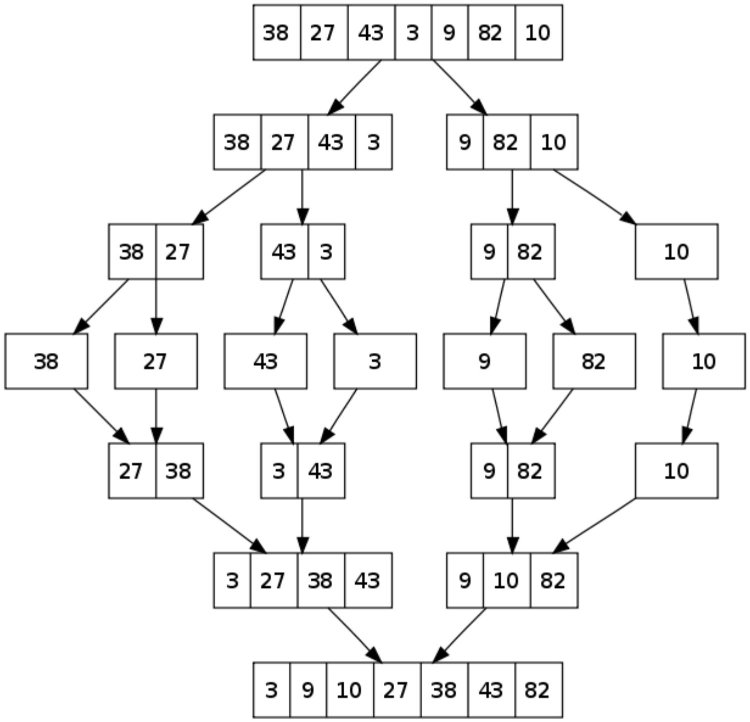
Counting

How many steps are needed to sort n numbers?

Algorithm 1 (Bubble Sort):

Every iteration moves the i-th smallest number to the i-th position

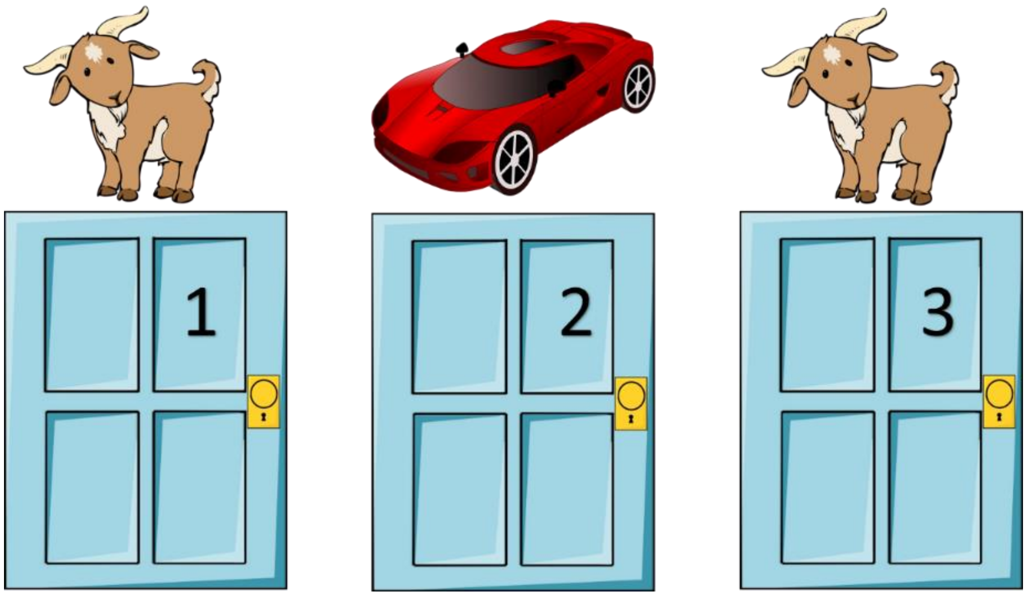
Algorithm 2 (Merge Sort):



Which algorithm runs faster?

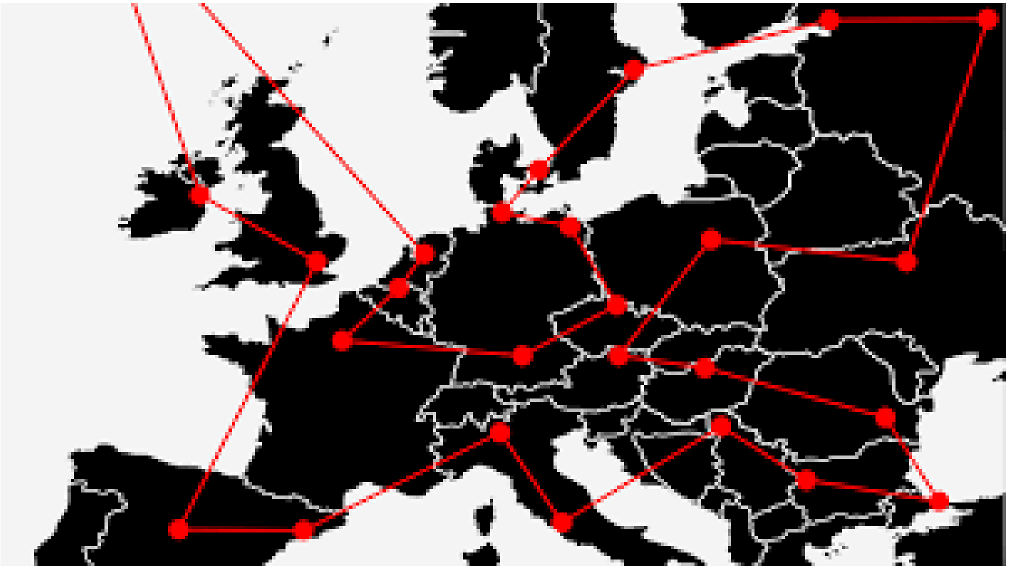
Solving the recursion

The Monty Hall Problem



- Wants to win a car: there are 3 doors, and you pick one (not knowing what's behind the doors)
- Host opens another door which has a goat
- You are given the option of switching to another door
- Should you switch?

Travelling Salesman Problem



Need to visit each city once

Find the shortest route

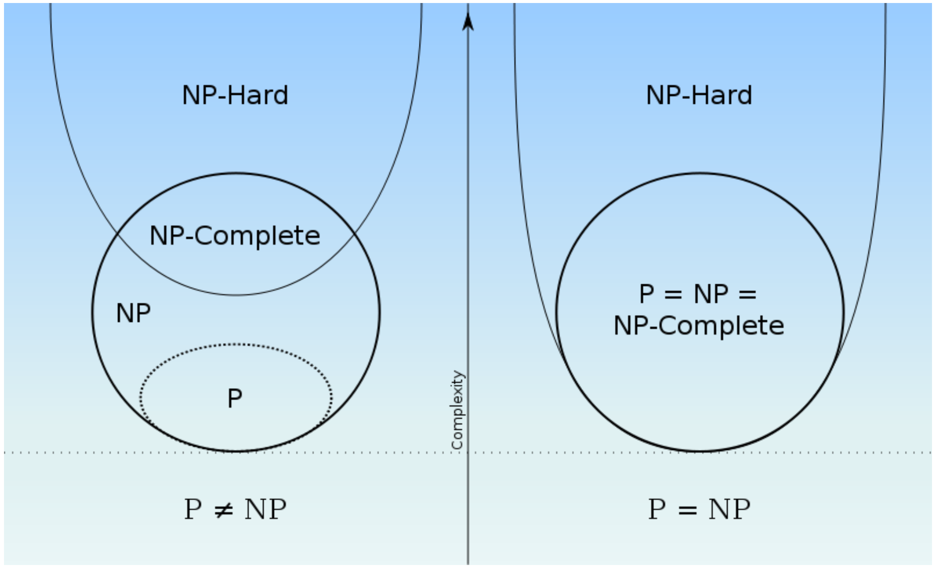
Stirling's Approximation

$$n! \sim \sqrt{2\pi n}\left(\frac{n}{e}\right)^n$$

$$20! \approx 2.43 \text{ x } 10^{18}$$

Computational Complexity: P=NP?

- Decision Problem: A problem with a **yes** or **no** answer
  - *Given a set of integers, is there a non-empty subset whose sum is zero?*
- The general class of problems for which some algorithm can provide a solution in polynomial time is called **P**
- The class of problems for which a solution can be ***verified*** or ***checked*** in polynomial time is called **NP**

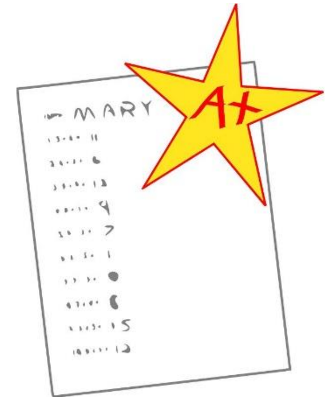


A decision problem X is **NP-complete** if:

1. X is in NP, and
2. Every problem in NP is reducible to X in polynomial time

A problem satisfying condition 2 is said to be **NP-hard**, whether or not it satisfies condition 1 (i.e. X may not be quickly checkable). NP-hard problems are "at least as hard as the NP-complete problems "

How to Perform Well?



Which of the following is correct?	Correction
1. $\emptyset \in \mathbb{R}$	$\emptyset \subseteq \mathbb{R}$
2. $x^2 \neq y^2 \Leftrightarrow x \neq y$	$x^2 \neq y^2 \Rightarrow x \neq y$
3. $a > b, c > d \Rightarrow ac > bd$	"ac" can be > or < or = "bd"
4. The sum of odd number of consecutive positive integers is odd.	1+2+3 is odd??

The most important thing in this course is NOT theory BUT a CLEAR MIND!!!

Think twice for every step you have done!

Sometimes, completing the solution maybe far away from being correct..

A Little Puzzle

$$1 = \sqrt{1} = \sqrt{(-1)(-1)}$$

$$= \sqrt{-1}\sqrt{-1} = \sqrt{-1}^{2}$$

$$= -1$$