

# Network Optimization

Jin Hyun Kim  
Fall 2019

# This Class

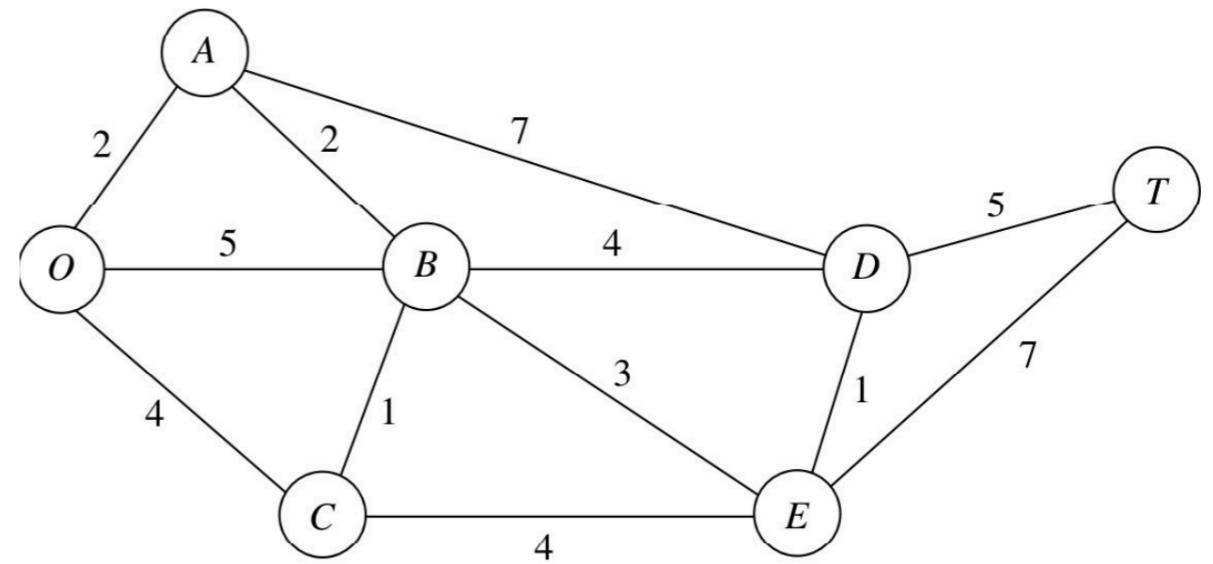
- Tue 7-9 (3:00pm - 6:00pm)
- Open Office Hour: By appointment at mentoring time
- Textbook:
  - Lecture slides
  - Papers

# This Class

- Will focus on problem solving techniques

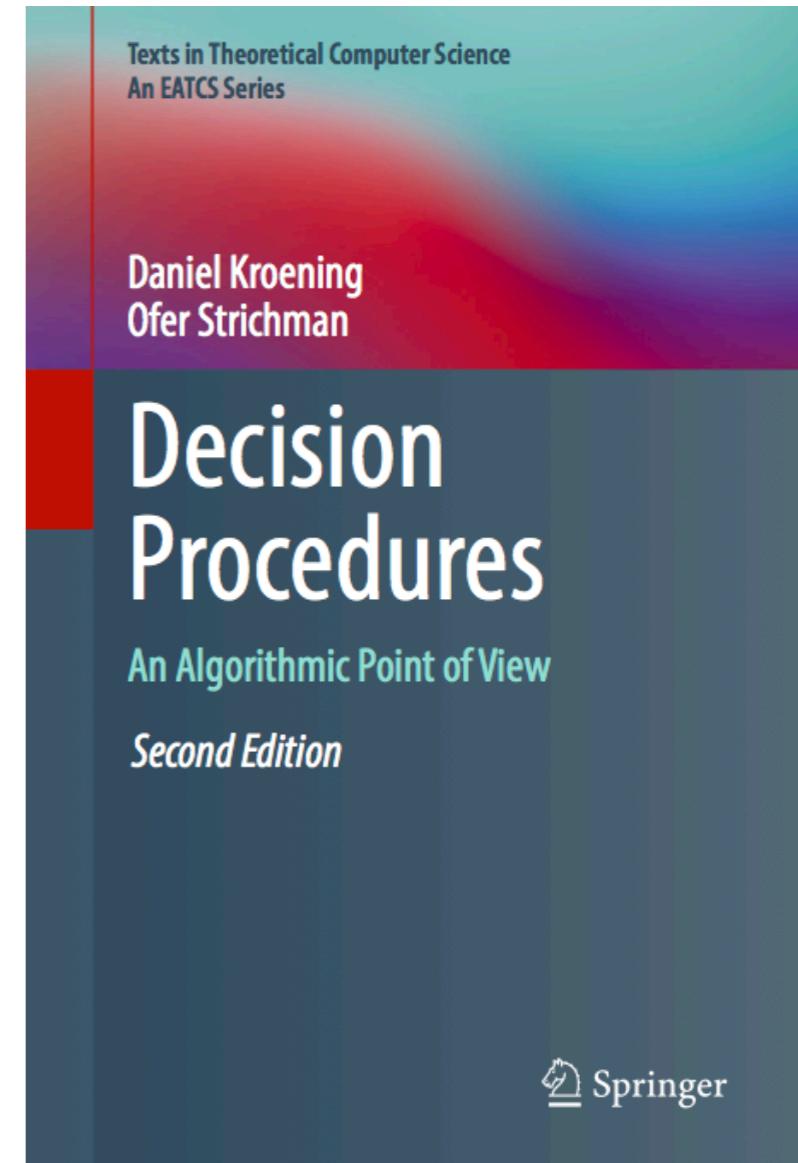
# An Example

- Seervada Park has a limited amount of sightseeing and backpack hiking
  - O: entrance of the park
  - T: station with scenic wonder
  - Determine route from park entrance to station T with smallest total distance for the



# Decision Procedures

- Satisfiability (SAT) - Propositional Logic
- Satisfiability Modulo Theory (SMT) - First Ordered Logic ...
- Integer Linear Programming (ILP) - Mathematical Optimization for integer variables
- Constrained Programming (CP) - No objective
- Machine Learning (ML)
- Model-based (Machine) Learning (MML)



# Programming Language History

## 1940년 이전 [편집]

초기의 프로그래밍 언어는 현대의 컴퓨터에서 거슬러 올라간다. 초기 1801년에 발명된 [지카드식 문작기](#)는 자동으로 장식 패턴을 발생시켰다. 1842년~1843년의 9개월의 기간에 걸쳐 [에이다 러브레이스](#)는 이를 자세하게 일련의 주석으로 추가했는데, 이는 일부 역사가들이 세계 최초의 컴퓨터 부호들은 이들을 응용하여 만들어졌다.

## 1940년대 [편집]

이 시기에 개발된 일부 중요 언어는 다음을 포함한다:

- 1943년 - [Plankalkül](#)
- 1943년 - [애니악 코딩 시스템](#)<sup>[2]</sup>
- 1949년~1954년 - 나중에 [유니박](#)으로 발전함.<sup>[3]</sup>

## 1950년대 ~ 1960년대 [편집]

1950년대에는 다음의 세 가지 현대의 프로그래밍 언어가 설계되었다:

- [포트란](#) (1954년)
- [리스프](#) (1958년)
- [코볼](#) (1959년)

그 밖의 주요 언어들은 다음과 같다:

- 1951년 - 지역 어셈블리어(Regional Assembly Language)
- 1952년 - [오토코드](#)
- 1954년 - [IPL](#) (리스프의 선구자)
- 1955년 - [FLOW-MATIC](#) (코볼의 선구자)
- 1957년 - [COMTRAN](#) (코볼의 선구자)
- 1958년 - [알골 58](#)
- 1959년 - [FACT](#) (코볼의 선구자)
- 1959년 - [RPG](#)
- 1962년 - [APL](#)
- 1962년 - [시뮬라](#)
- 1962년 - [SNOBOL](#)
- 1963년 - [CPL](#) (C의 선구자)
- 1964년 - [BASIC](#)
- 1964년 - [PL/I](#)
- 1967년 - [BCPL](#) (C의 선구자)

## 1968년대 ~ 1978년대 [편집]

1960년대 말에서 1970년대 말의 기간 동안 다음의 다섯 가지 각이 기간 동안 개발된 주요 언어들은 다음과 같다:

- [시뮬라](#)(Simula, 1960년대 말 발명)
- [C](#)
- [스몰토크](#) (1970년대 중반)
- [프롤로그](#) (1972년)
- [ML](#)

그 밖의 주요 언어로는 다음과 같다.

- 1968년 - [로고](#)
- 1969년 - [B](#) (C의 선구자)
- 1970년 - [파스칼](#)
- 1970년 - [포스](#) (Forth)
- 1975년 - [스ქ](#) (Scheme)
- 1978년 - [SQL](#) (처음에는 쿼리 언어일 뿐이었으나, 훗날 프로세서로 기간 동안 개발된 주요 언어들은 다음과 같다:

## 1980년대 [편집]

이 기간 동안 개발된 주요 언어들은 다음과 같다:

- 1980년 - [C++](#) (처음 이름은 C with classes였으나 1983년에 [에이다](#)로 변경)
- 1983년 - [커먼 리스프](#)
- 1984년 - [MATLAB](#)
- 1985년 - [에펠\(Eiffel\)](#)
- 1986년 - [오브젝티브-C](#)
- 1986년 - [얼랭](#)
- 1987년 - [펄](#)
- 1988년 - [Tcl](#)
- 1988년 - [매스매틱\(Mathematica\)](#)
- 1989년 - [FL](#) (Backus-Naur Form)

## 1990년대 [편집]

이 기간 동안 개발된 주요 언어들은 다음과 같다:

- 1990년 - [하스켈](#)
- 1991년 - [파이썬](#)
- 1991년 - [비주얼 베이직](#)
- 1991년 - [HTML](#) (마크업 언어)
- 1993년 - [루비](#)
- 1993년 - [루아](#)
- 1994년 - [CLOS](#) (ANSI 커먼 리스프의 일부)
- 1995년 - [자바](#)
- 1995년 - [델파이 \(오브젝트 파스칼\)](#)
- 1995년 - [자바스크립트](#)
- 1995년 - [PHP](#)
- 1996년 - [WebDNA](#)
- 1997년 - [Rebol](#)
- 1999년 - [D](#)

## 현재의 경향 [편집]

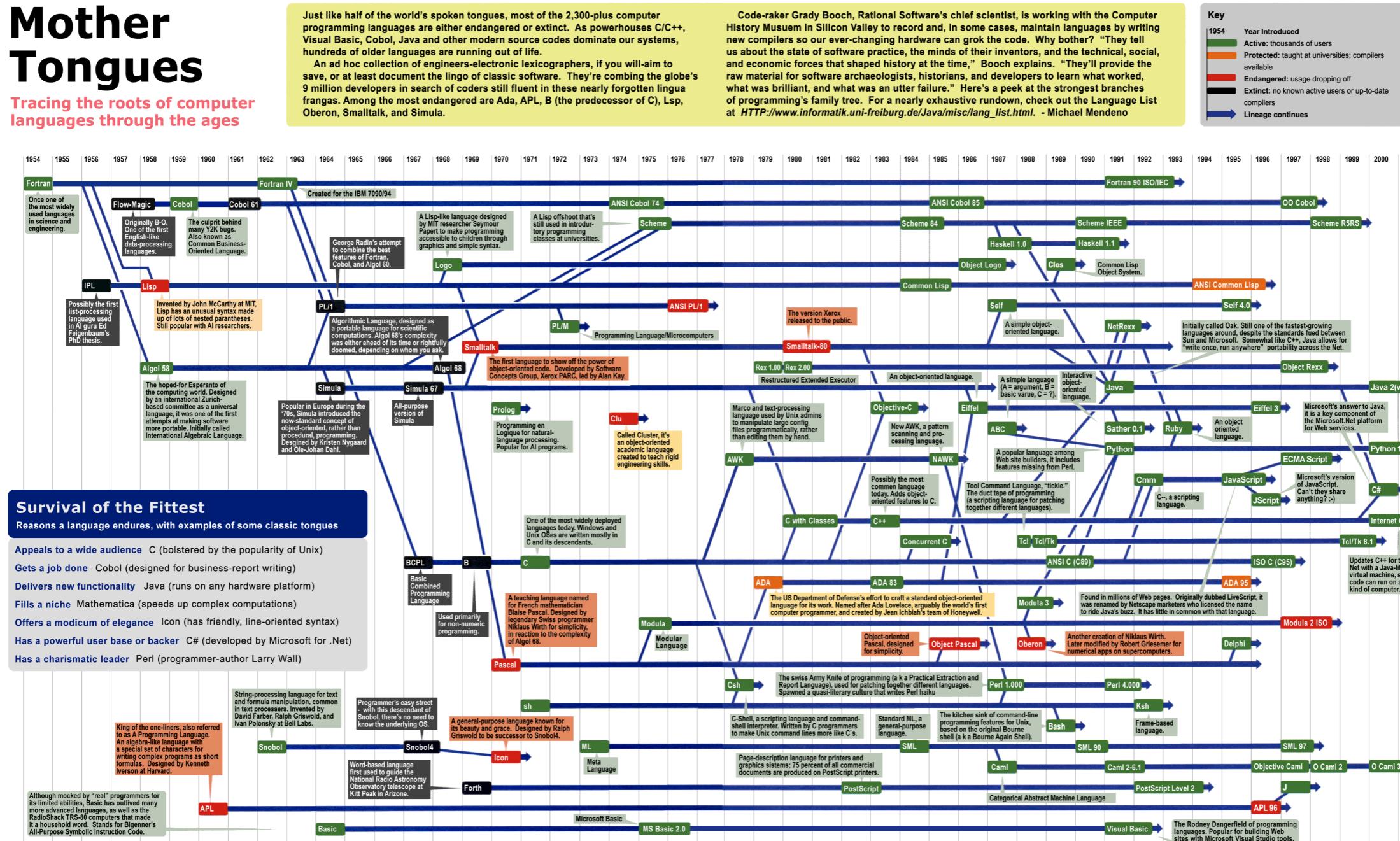
2000년 이후로 개발된 주요 언어들은 다음과 같다:

- 2000년 - [액션스크립트](#)
- 2001년 - [C#](#)
- 2001년 - [비주얼 베이직 닷넷](#)
- 2002년 - [F#](#)
- 2003년 - [그루비](#)
- 2003년 - [스칼라](#)
- 2003년 - [팩터](#)
- 2007년 - [Clojure](#)
- 2009년 - [고](#)
- 2011년 - [다트](#)
- 2014년 - [스위프트](#)
- 2015년 - [러스트](#)

# Programming Language History

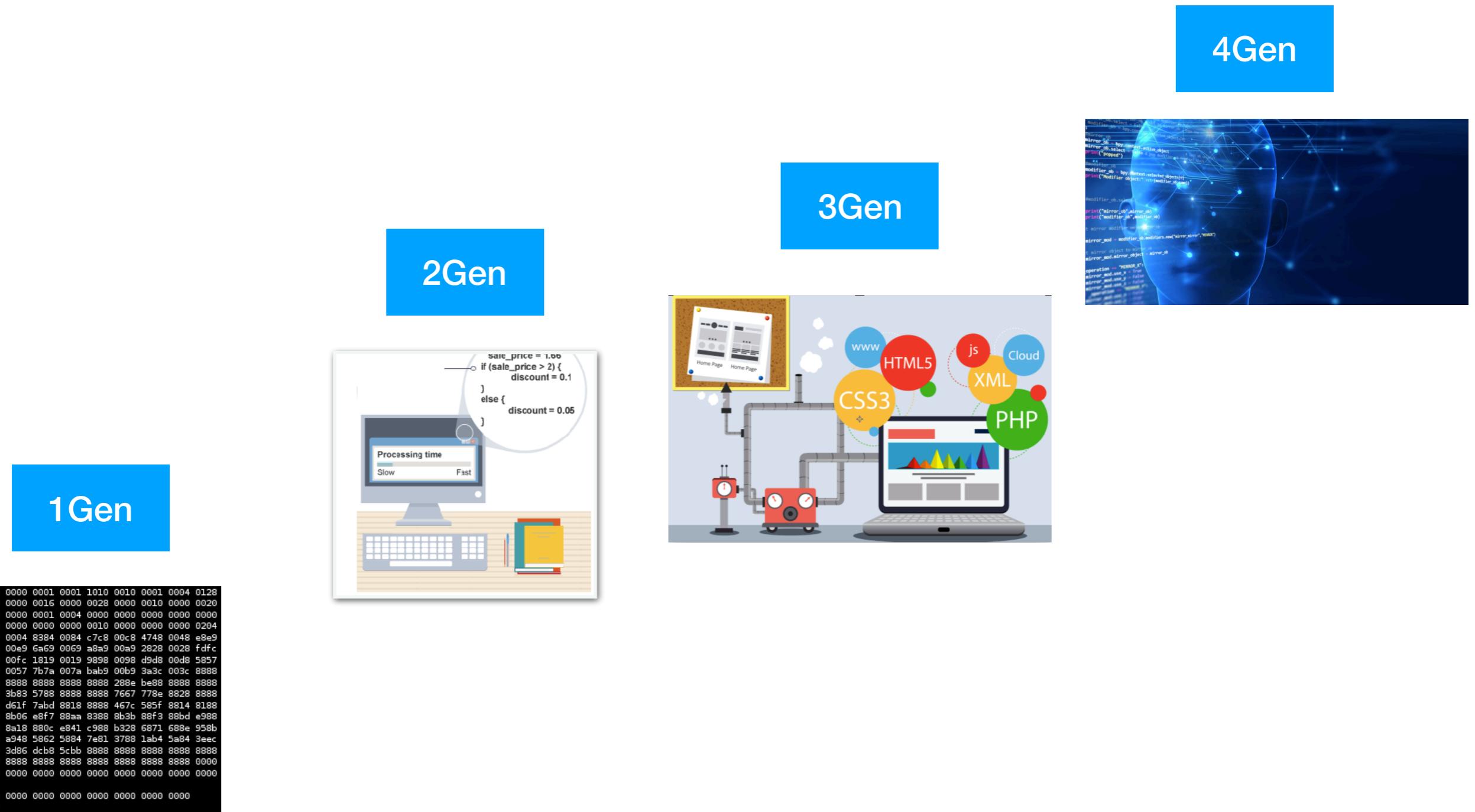
## Mother Tongues

Tracing the roots of computer languages through the ages

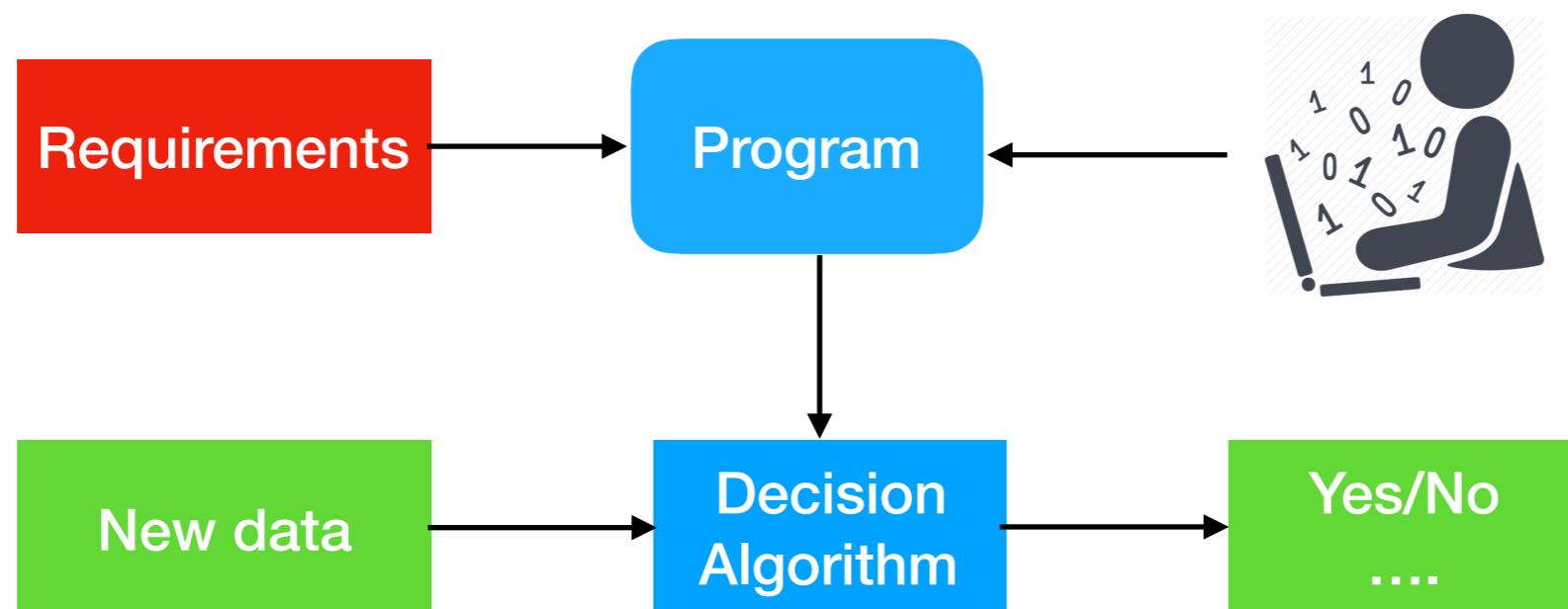


Sources: Paul Boutin; Brent Hailpern, associate director of computer science at IBM Research; The Retrocomputing Museum; Todd Proebsting, senior researcher at Microsoft; Gio Wiederhold, computer scientist, Stanford University

# Programming Paradigm Evolution

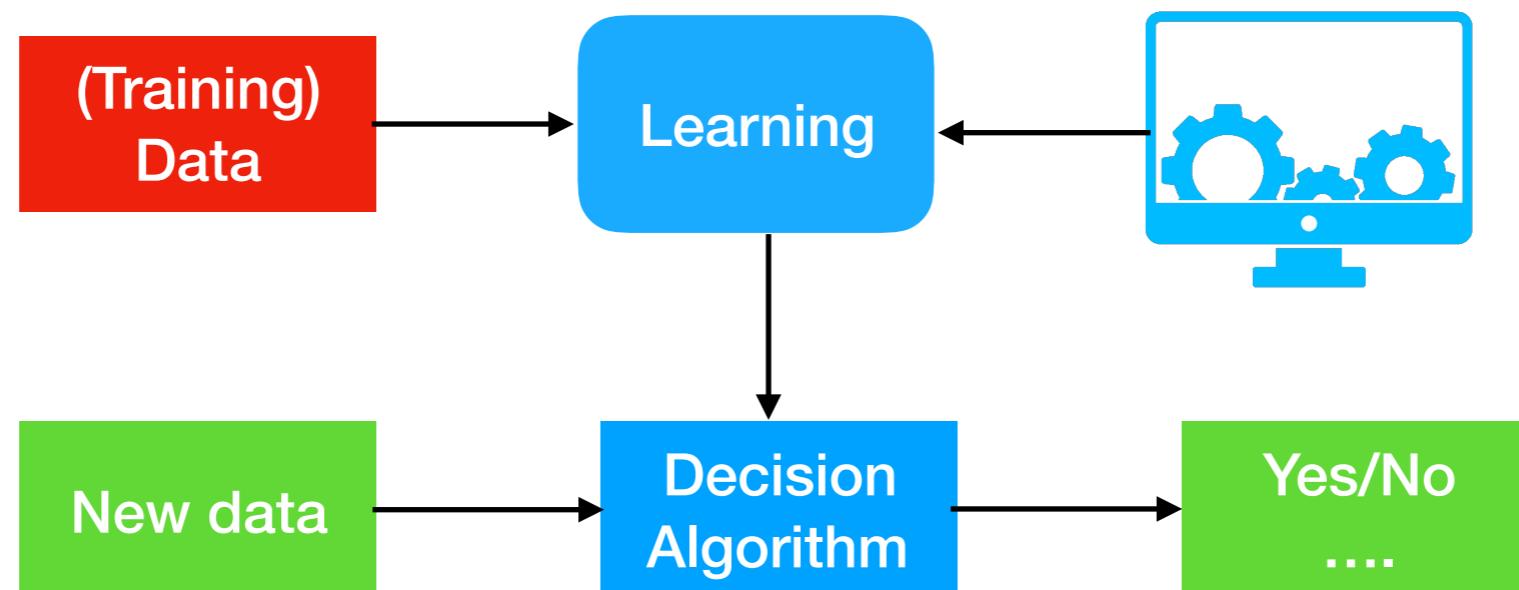


# Programming



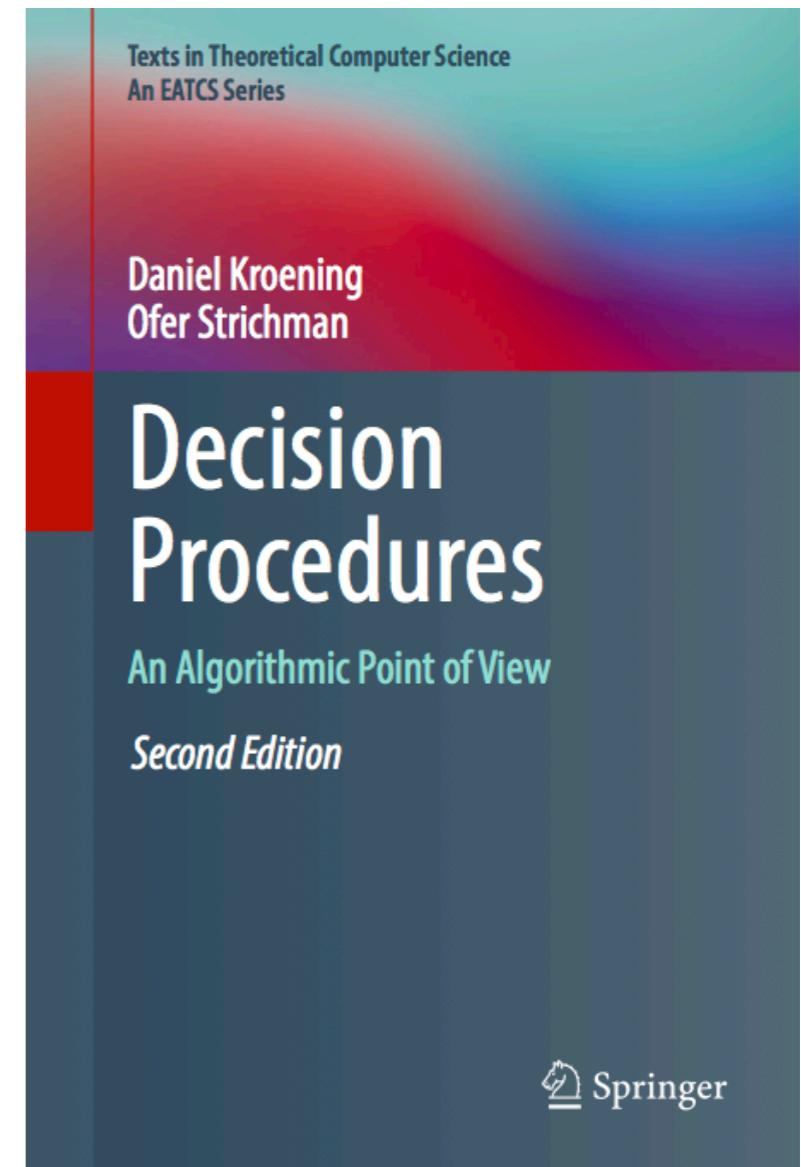
# Machine Learning

- Learning to decision



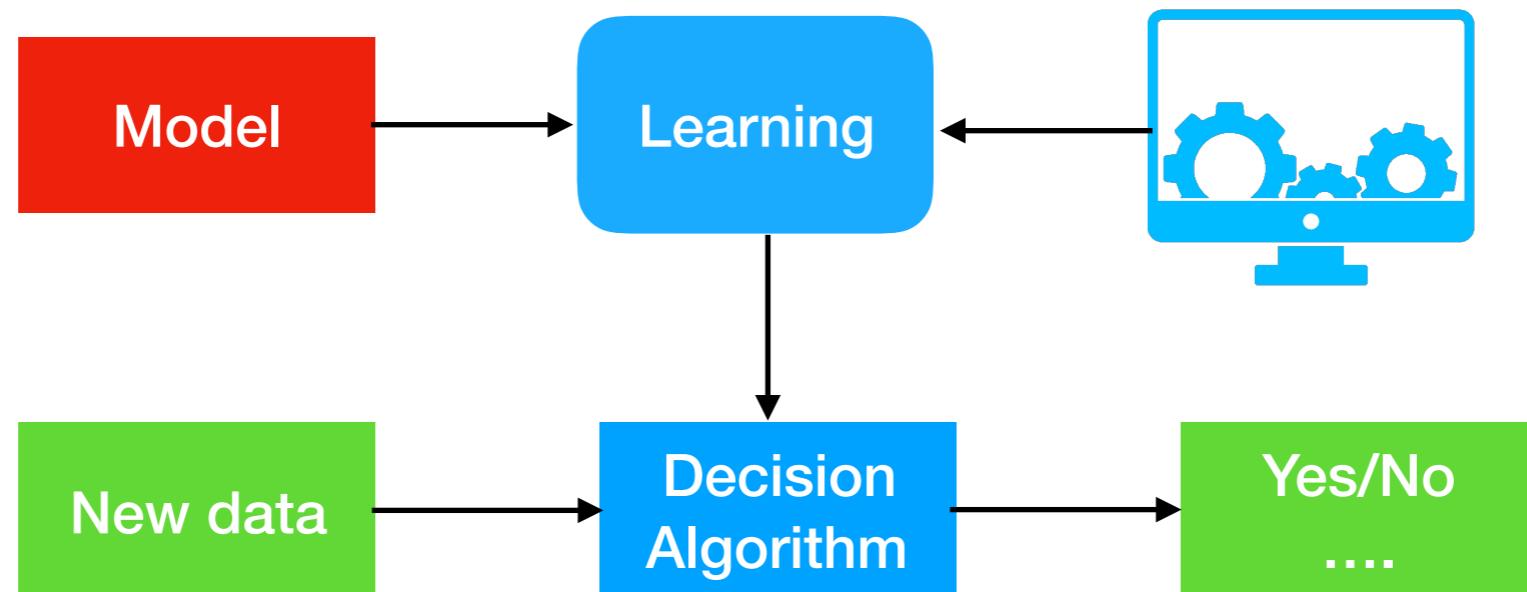
# Decision Procedures

- Satisfiability (SAT) - Propositional Logic
- Satisfiability Modulo Theory (SMT) - First Ordered Logic ...
- Integer Linear Programming (ILP) - Mathematical Optimization for integer variables
- Constrained Programming (CP) - No objective
- Machine Learning (ML)
- Model-based (Machine) Learning (MML)



# Model-based Machine Learning (MML)

- Instead of training data, a model is given



# This Class

- Discuss 3 subjects:
  - Concept of Optimization - Network, Real-time systems, ...
  - Time-Sensitive Network
  - Solver
    - Integer Linear Program
    - Constraint Programming
    - Uppaal Stratego

# Lecture Plan

Week	Lecture Contents	Textbook	Presenter
1	Introduction	Lecture slides	
2	Concept of Network Optimization	Lecture slides	Jin Hyun Kim
3	Scheduling Time Sensitive Network	Lecture slides	Jin Hyun Kim
4	- UPPAAL STRATEGO: A Model-based Learning for Optimization Problems. - HW 1 Out	Lecture slides	
5	UPPAAL STRATEGO Project Presentation	Lecture slides	
6	Integer Linear Programming 1	Lecture slides	
7	- Integer Linear Programming 2 - HW 2 Out		
8	A Network Optimization Problem Solving with ILP - Term Project Description Out		
9	Mid Exam - Term Project Proposal	Lecture slides	Jin Hyun Kim
10	Satisfiability (SAT)	Lecture slides	Jin Hyun Kim
11	Satisfiability Modulo Theory (SMT)	Lecture slides	Jin Hyun Kim
12	Satisfiability Modulo Theory (SMT)		
13	Constraint Programming 1 - HW 3 Out		
14	Constraint Programming 1		
15	A Network Optimization Problems Solving 2 with CP		
16	Final Exam - Term Project Presentations		

# Grading

- Class Participation: 10%
- Term: 40%
- Paper Submission (50%) to KIISE (Conference or Journal)

# Homepage

- Class information, Lecture slides, Readings, Assignment descriptions, Schedules etc
  - <http://jinkimh.github.io/netopt>
- Instant announcements, Questions, Assignment & Quiz submission:
  - Google Classroom: **I2y8sh4**
  - Using this Google Classroom, the grade will be evaluated, so make sure that you subscribe Google Classroom using above code

# Next Class

- Network Optimization Problems