Aalto University

Distributed Algorithms 2024

0 About this course

Algorithms for networks

*How can individual nodes
in a large computer network
work together towards
a common goal?*

Learning objectives

- Understand models of distributed computing
- Design and analyze efficient distributed algorithms
- Prove impossibility results
- Use standard graph-theoretic concepts

Practiced in exercises

Tested in the exam

Zero to research-level

- No prior knowledge on distributed systems expected
- We will reach topics close to current research by the end of the course

Good start for a Master’s thesis or PhD studies

Intensive course

5 credits in 6 weeks

≈

22 working hours/week

This is a theory course

- **100% mathematics**
  - definitions
  - theorems
  - proofs …
- **0% practice**
  - programming
  - hardware
  - protocols …

Prerequisite:

Introduction to Mathematical Reasoning for Computer Scientists

Grading

- **To pass the course:**
  - you need to pass the ***exam***
- **For a good grade:**
  - you need to ***solve exercises***
  - quiz + exercises = max 96 points in total
  - 80 points = grade 5/5

			Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8
Tue	16.15	Exercise deadline		1 + 2	3 + 4	5 + 6	7 + 8	9 + 10	11 + 12	
	16.15	Lecture	1 + 2	–	5 + 6	7 + 8	–	11 + 12		
Wed	10.15	Quiz deadline	1	3	5	7	9	11		
	10.15	Exercise session	1 + 2	3	5	7	9	11		
Thu	9.00	Exam								1–12
	14.15	Lecture	3 + 4	–	recap	9 + 10	–	recap		
Fri	12.15	Quiz deadline	2	4	6	8	10	12		
	12.15	Exercise session	–	4	6	8	10	12		

			Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8
Tue	16.15	Exercise deadline		**1 + 2**	3 + 4	5 + 6	7 + 8	9 + 10	11 + 12	
	16.15	Lecture	**1 + 2**	–	5 + 6	7 + 8	–	11 + 12		
Wed	10.15	Quiz deadline	**1**	3	5	7	9	11		
	10.15	Exercise session	**1 + 2**	3	5	7	9	11		
Thu	9.00	Exam								1–12
	14.15	Lecture	3 + 4	–	recap	9 + 10	–	recap		
Fri	12.15	Quiz deadline	**2**	4	6	8	10	12		
	12.15	Exercise session	–	4	6	8	10	12		

			Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8
Tue	16.15	Exercise deadline		1 + 2	**3 + 4**	5 + 6	7 + 8	9 + 10	11 + 12	
	16.15	Lecture	1 + 2	–	5 + 6	7 + 8	–	11 + 12		
Wed	10.15	Quiz deadline	1	**3**	5	7	9	11		
	10.15	Exercise session	1 + 2	**3**	5	7	9	11		
Thu	9.00	Exam								1–12
	14.15	Lecture	**3 + 4**	–	recap	9 + 10	–	recap		
Fri	12.15	Quiz deadline	2	**4**	6	8	10	12		
	12.15	Exercise session	–	**4**	6	8	10	12		

			Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8
Tue	16.15	Exercise deadline		1 + 2	3 + 4	**5 + 6**	7 + 8	9 + 10	11 + 12	
	16.15	Lecture	1 + 2	–	**5 + 6**	7 + 8	–	11 + 12		
Wed	10.15	Quiz deadline	1	3	**5**	7	9	11		
	10.15	Exercise session	1 + 2	3	**5**	7	9	11		
Thu	9.00	Exam								1–12
	14.15	Lecture	3 + 4	–	recap	9 + 10	–	recap		
Fri	12.15	Quiz deadline	2	4	**6**	8	10	12		
	12.15	Exercise session	–	4	**6**	8	10	12		

			Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8
Tue	16.15	Exercise deadline		1 + 2	3 + 4	5 + 6	**7 + 8**	9 + 10	11 + 12	
	16.15	Lecture	1 + 2	–	5 + 6	**7 + 8**	–	11 + 12		
Wed	10.15	Quiz deadline	1	3	5	**7**	9	11		
	10.15	Exercise session	1 + 2	3	5	**7**	9	11		
Thu	9.00	Exam								1–12
	14.15	Lecture	3 + 4	–	recap	9 + 10	–	recap		
Fri	12.15	Quiz deadline	2	4	6	**8**	10	12		
	12.15	Exercise session	–	4	6	**8**	10	12		

			Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8
Tue	16.15	Exercise deadline		1 + 2	3 + 4	5 + 6	7 + 8	**9 + 10**	11 + 12	
	16.15	Lecture	1 + 2	–	5 + 6	7 + 8	–	11 + 12		
Wed	10.15	Quiz deadline	1	3	5	7	**9**	11		
	10.15	Exercise session	1 + 2	3	5	7	**9**	11		
Thu	9.00	Exam								1–12
	14.15	Lecture	3 + 4	–	recap	**9 + 10**	–	recap		
Fri	12.15	Quiz deadline	2	4	6	8	**10**	12		
	12.15	Exercise session	–	4	6	8	**10**	12		

			Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8
Tue	16.15	Exercise deadline		1 + 2	3 + 4	5 + 6	7 + 8	9 + 10	**11 + 12**	
	16.15	Lecture	1 + 2	–	5 + 6	7 + 8	–	**11 + 12**		
Wed	10.15	Quiz deadline	1	3	5	7	9	**11**		
	10.15	Exercise session	1 + 2	3	5	7	9	**11**		
Thu	9.00	Exam								1–12
	14.15	Lecture	3 + 4	–	recap	9 + 10	–	recap		
Fri	12.15	Quiz deadline	2	4	6	8	10	**12**		
	12.15	Exercise session	–	4	6	8	10	**12**		

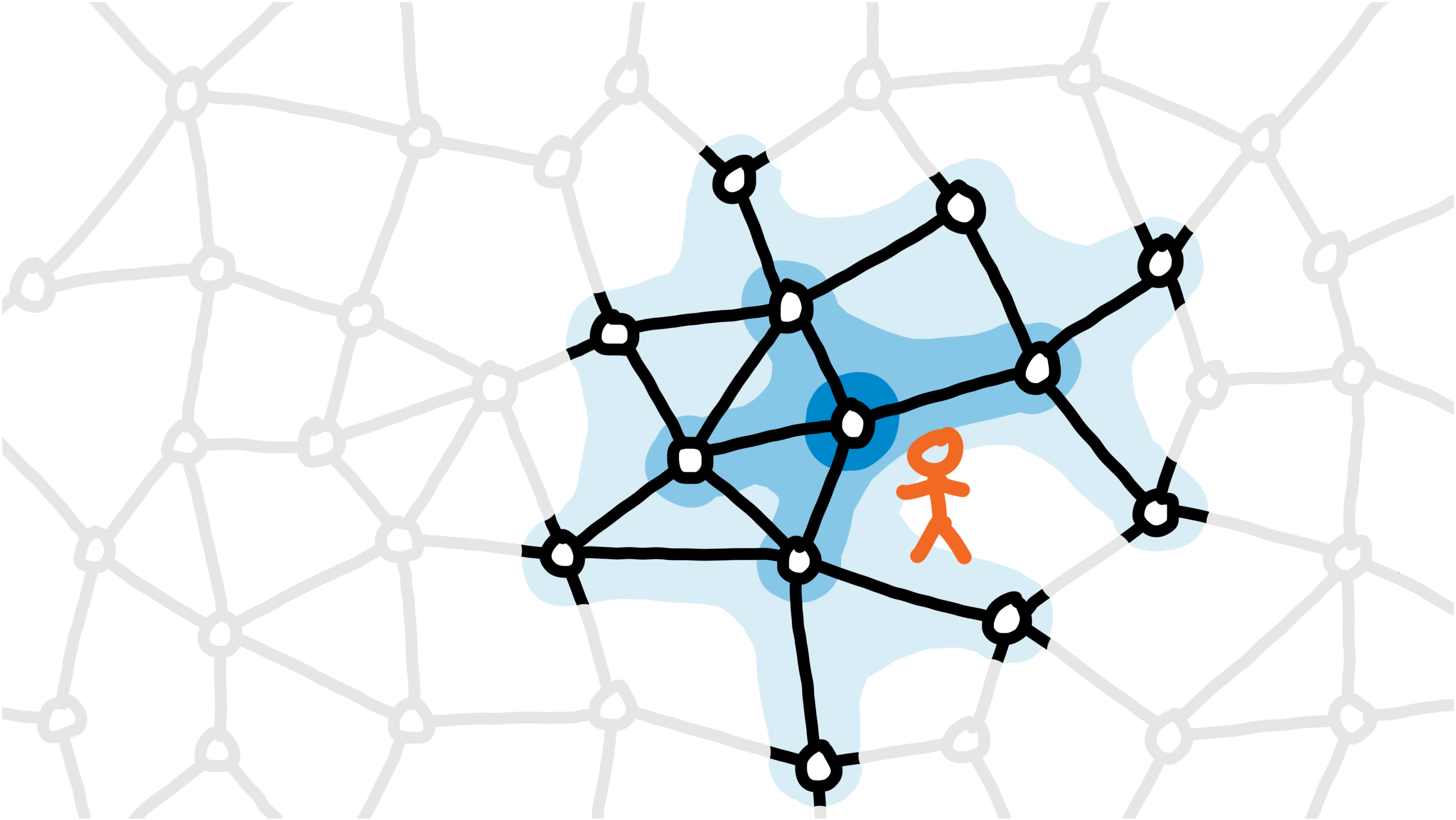
			Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8
Tue	16.15	Exercise deadline		1 + 2	3 + 4	5 + 6	7 + 8	9 + 10	11 + 12	
	16.15	Lecture	1 + 2	–	5 + 6	7 + 8	–	11 + 12		
Wed	10.15	Quiz deadline	1	3	5	7	9	11		
	10.15	Exercise session	1 + 2	3	5	7	9	11		
Thu	9.00	Exam								1–12
	14.15	Lecture	3 + 4	–	recap	9 + 10	–	recap		
Fri	12.15	Quiz deadline	2	4	6	8	10	12		
	12.15	Exercise session	–	4	6	8	10	12		

Aalto University

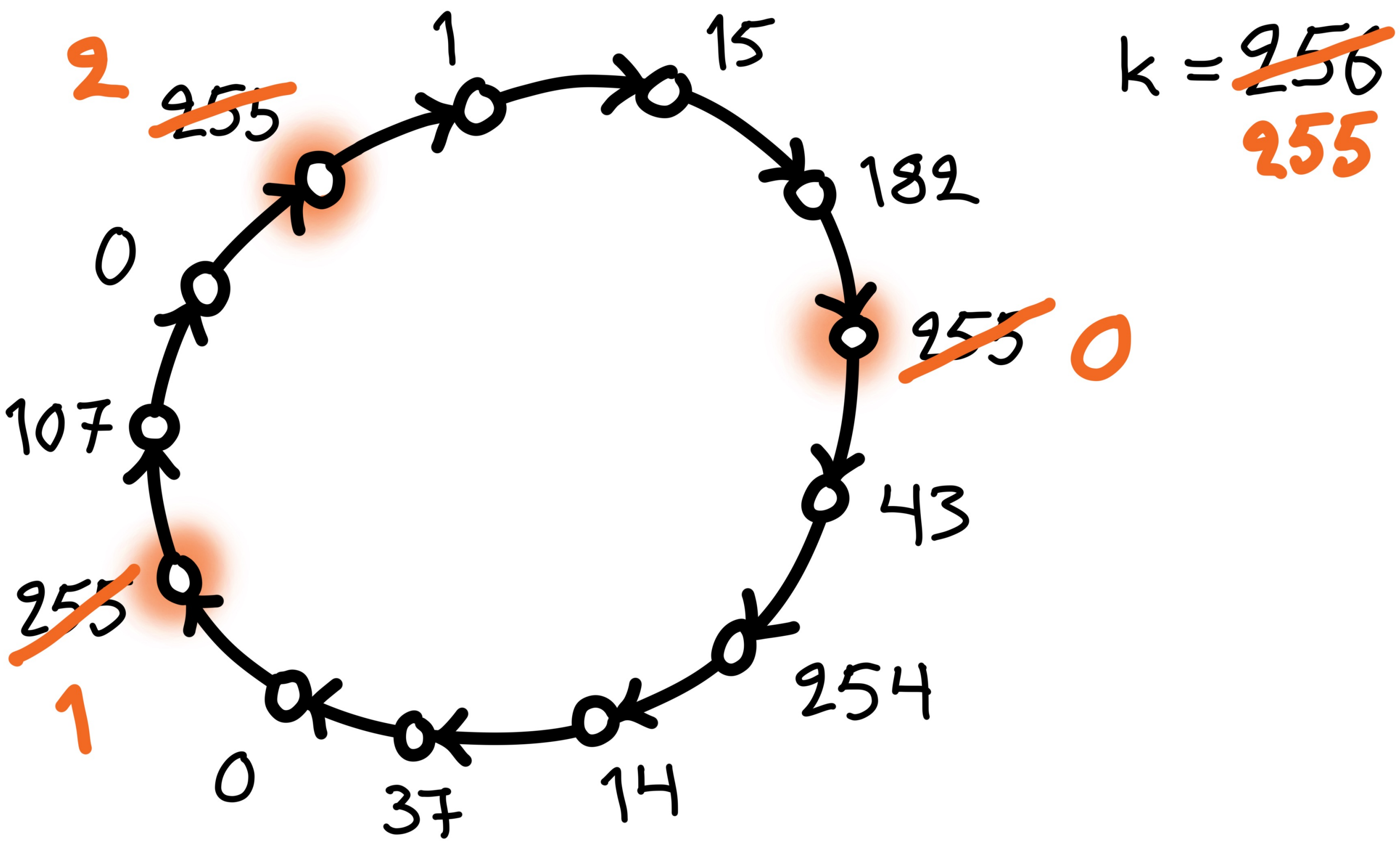
Distributed Algorithms 2024

1 Warm-up

Video 1a: introduction

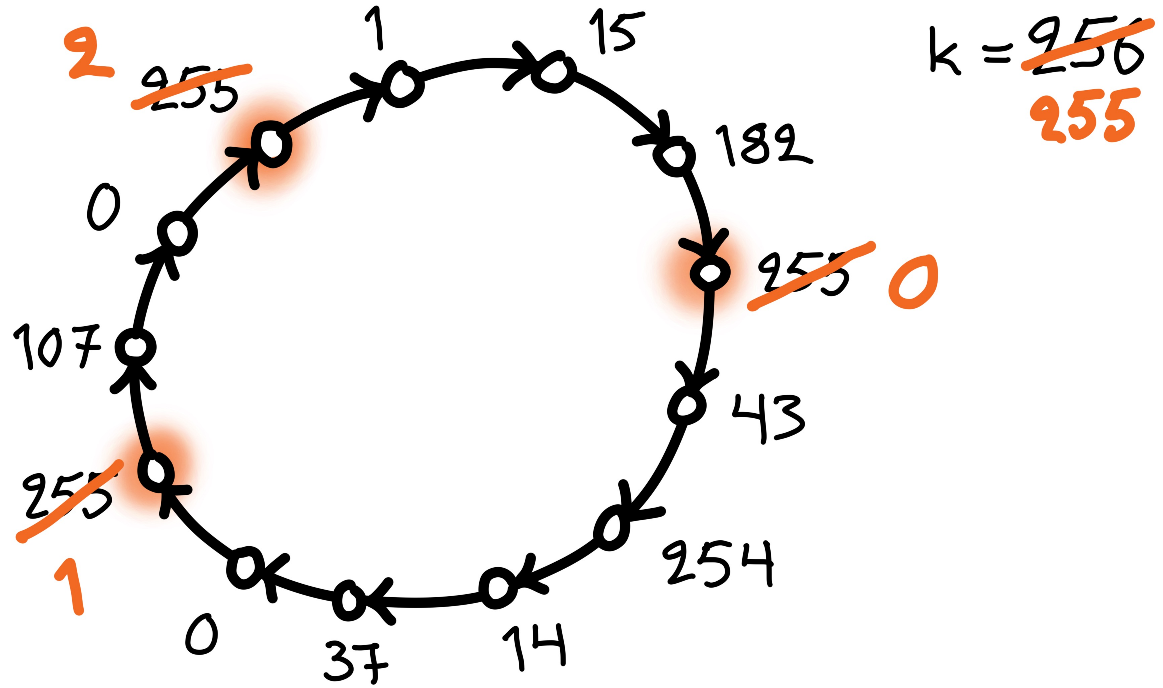


Video 1b: coloring



Slow color reduction

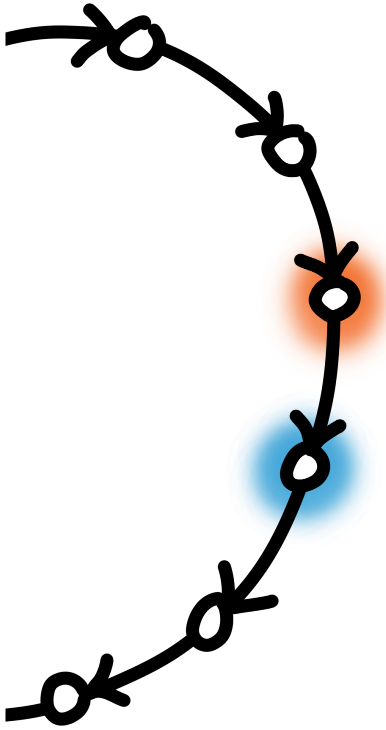
- Algorithm idea:
  - all nodes with the **largest color** are active
  - active nodes pick the ***smallest color that is not used by their neighbors***



Puzzle 1

- Consider a simpler algorithm idea:
  - ***all nodes*** pick the smallest color that is not used by their neighbors
- **What would go wrong?**
  - *construct an example in which this algorithm fails!*

Video 1b: coloring fast



0-7   0-1   3 bits   1 bit

$(2, 0) = \underbrace{010}_{\text{3 bits}}0$

$k = 256$

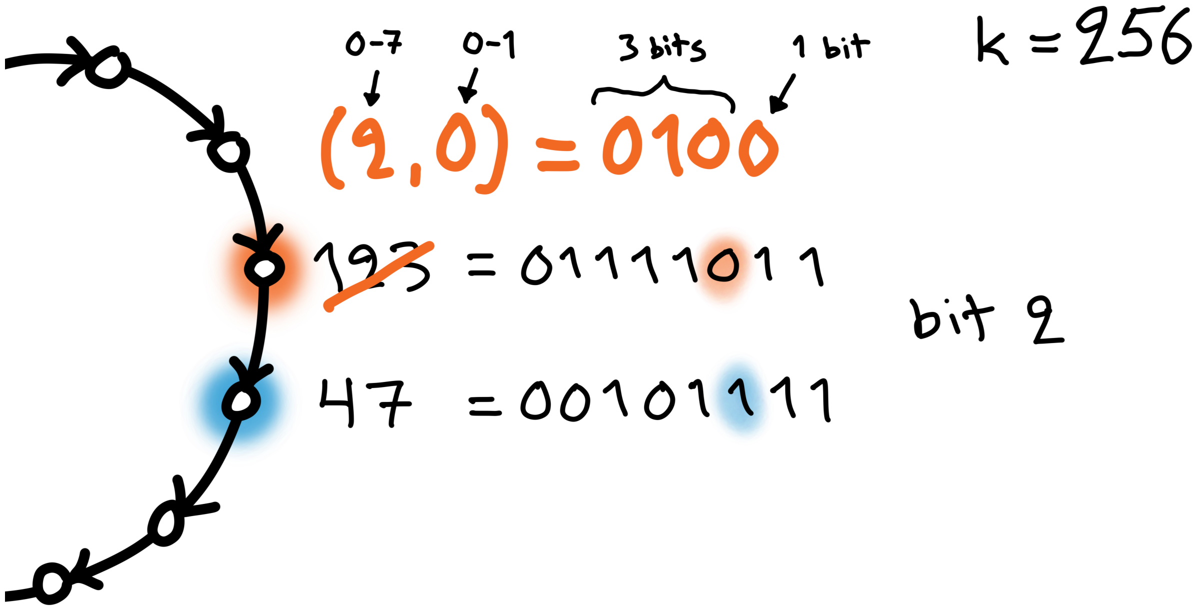
~~123~~ $= 01111011$

bit 2

$47 = 00101111$

Fast color reduction

- Algorithm idea:
  - find the first bit that differs in successor
  - index ***i***, bit value ***b***
  - new color is (***i***, ***b***)



Puzzle 2

- Algorithm idea:
  - find the first bit that differs in successor
  - index i, bit value b
  - new color is (i, b)
- **What would go wrong if the new color was just b?**
  - *construct an example in which it fails!*

Puzzle 3

- Algorithm idea:
  - find the first bit that differs in successor
  - index ***i***, bit value ***b***
  - new color is (***i***, ***b***)
- **What would go wrong if the new color was just *i*?**
  - *construct an example in which it fails!*

Puzzle 4

- Algorithm idea:
  - find the first bit that differs in successor
  - index ***i***, bit value ***b***
  - new color is (***i***, ***b***)
- **Why does the algorithm work correctly?**
  - *why is my new color always different from the new colors of my successor and my predecessor?*

Aalto University

Distributed Algorithms 2024

2 Graph-theoretic foundations

Graphs in this course

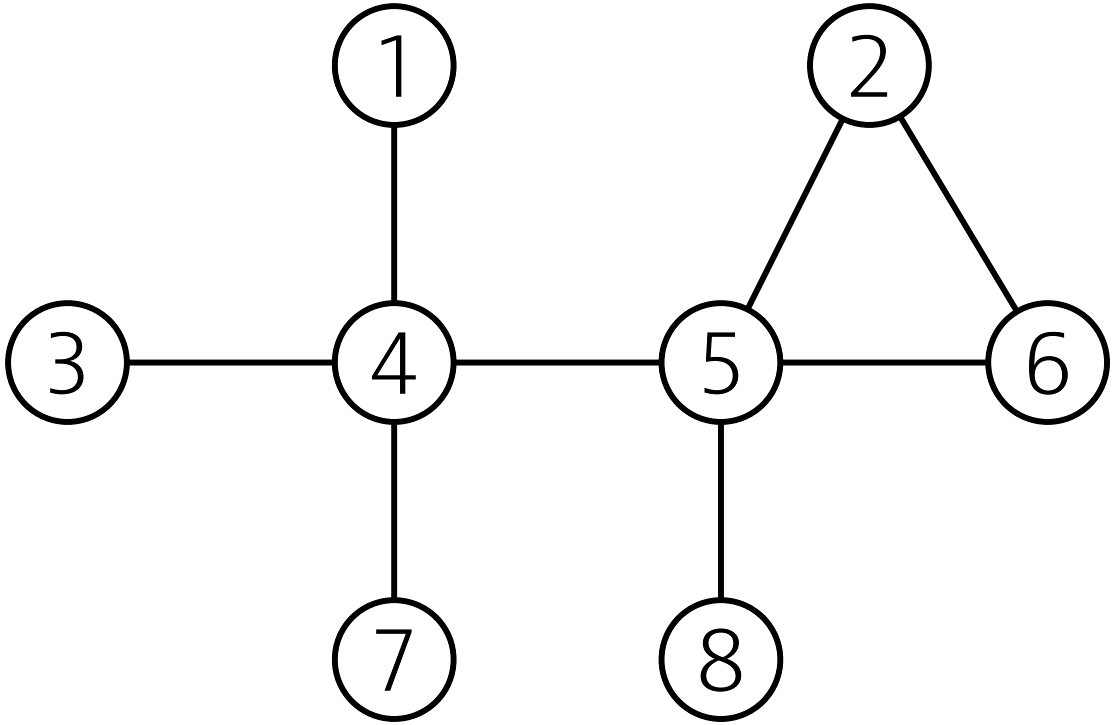
- Defining:
  - models of distributed computing
  - what we want to solve
  - what are the assumptions
- Designing & analyzing algorithms
- Proving impossibility results
- Often: ***graph ≈ network, node ≈ computer***

Please do not confuse

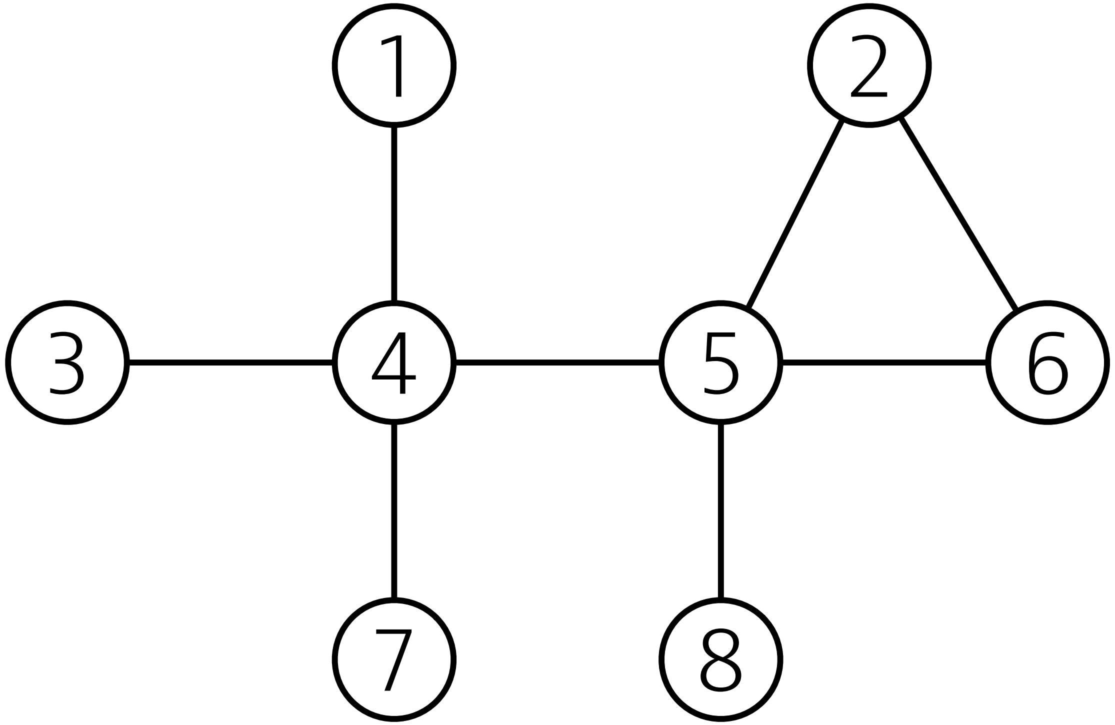
- **Maximal**
  - not a subset of another solution
  - very easy to find: add greedily
- **Maximum**
  - largest possible solution
  - often hard to find

Please do not confuse

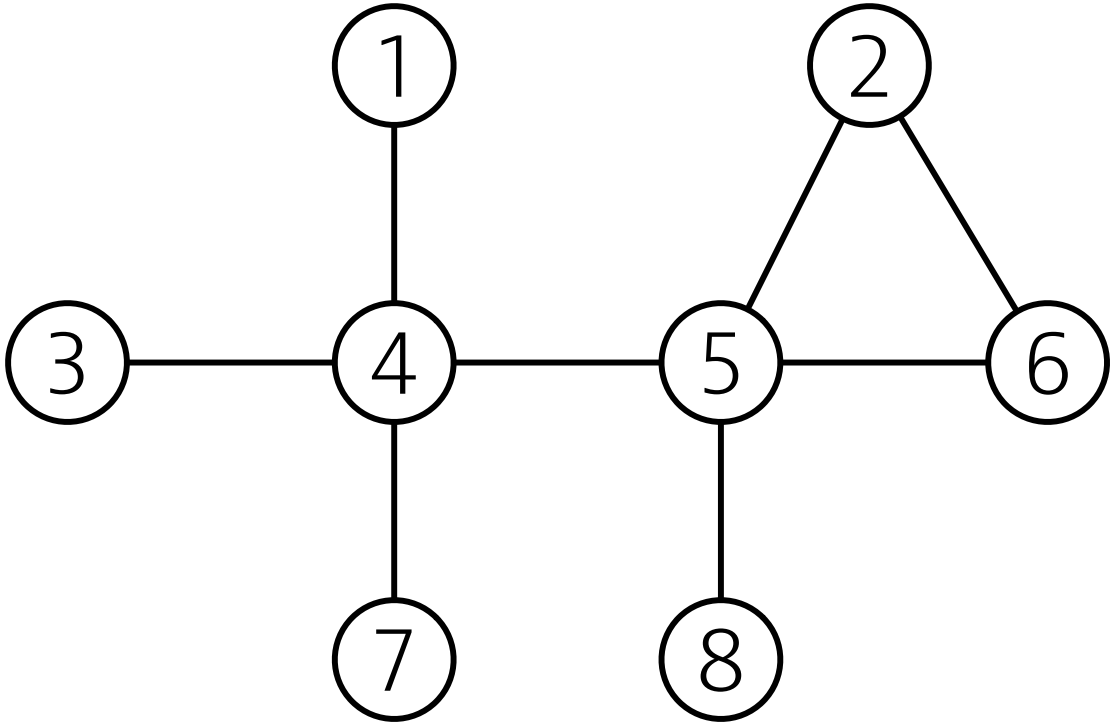
- **Minimal**
  - not a superset of another solution
  - very easy to find: remove greedily
- **Minimum**
  - smallest possible solution
  - often hard to find



Minimum vertex cover

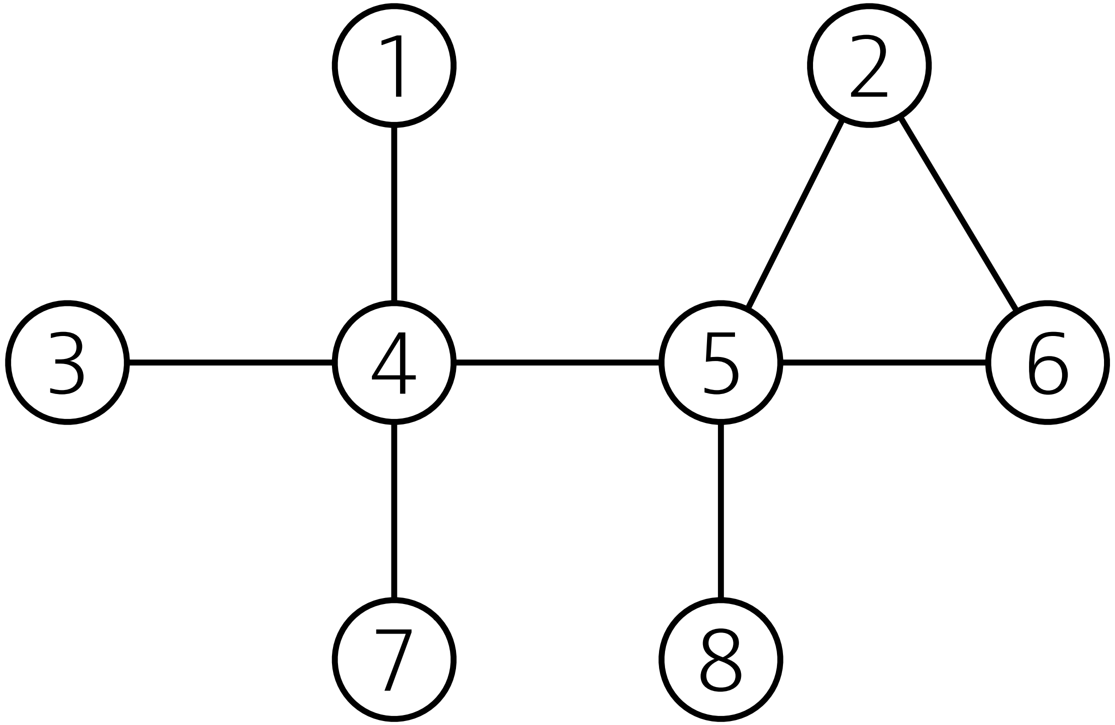


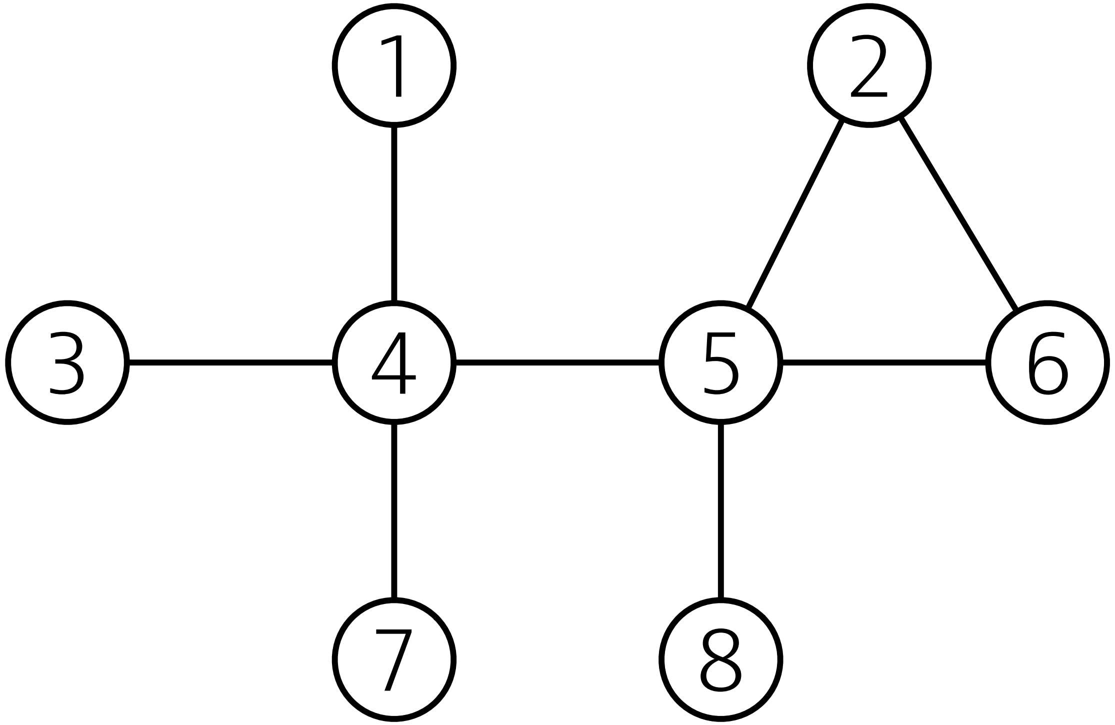
Minimum dominating set



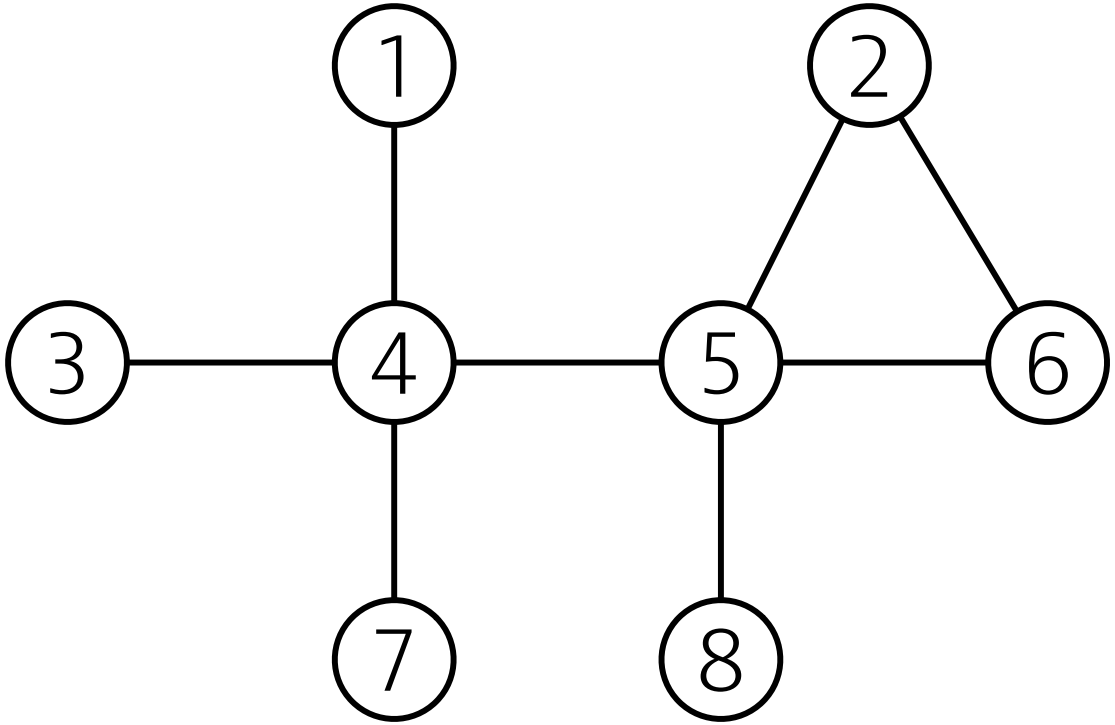
Maximum independent set

Smallest
set of nodes
that is both
an independent set
and a dominating set

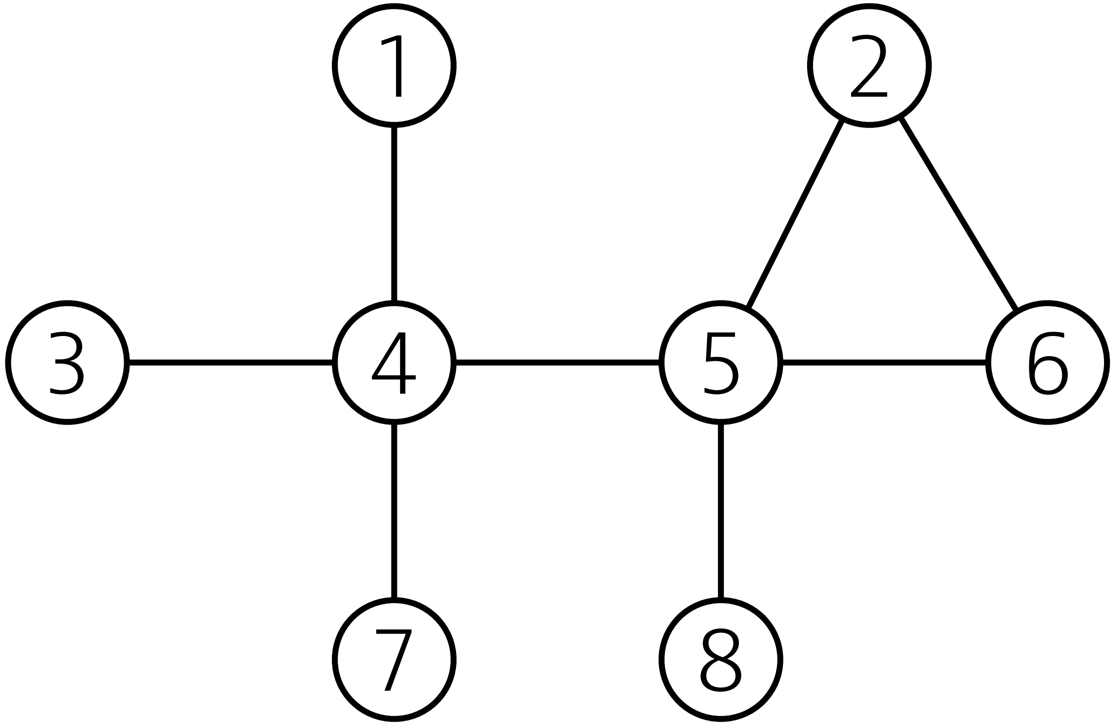




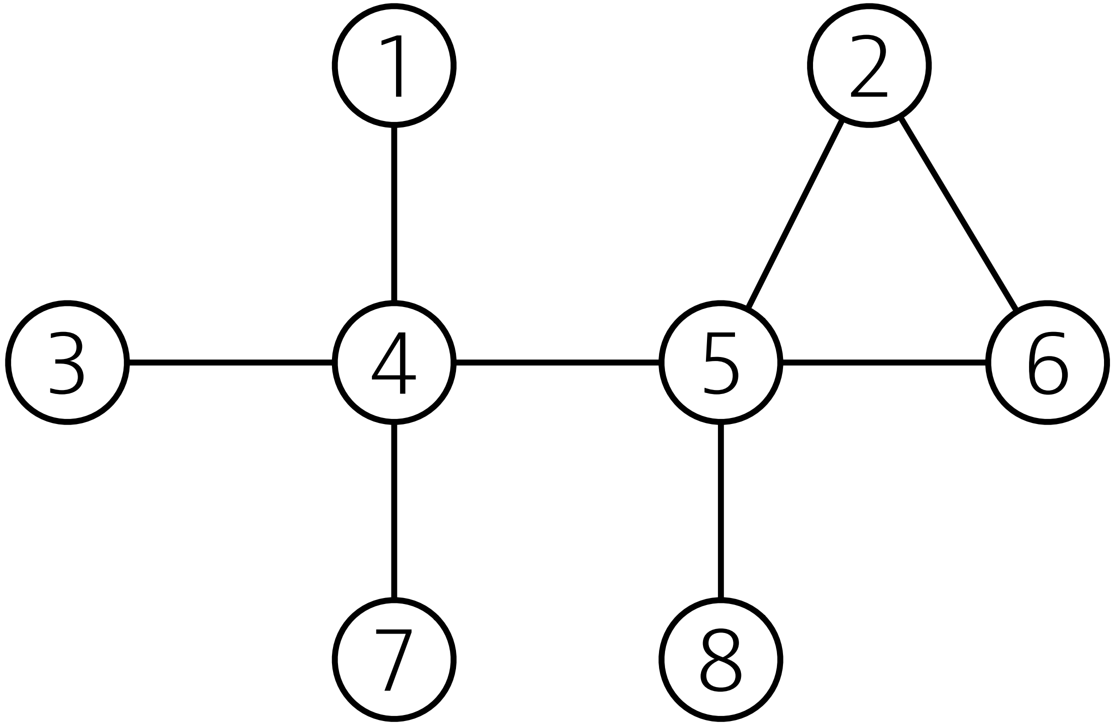
Largest
set of nodes
that is both
an independent set
and a dominating set



Maximum matching



Minimum edge cover



Minimum edge dominating set