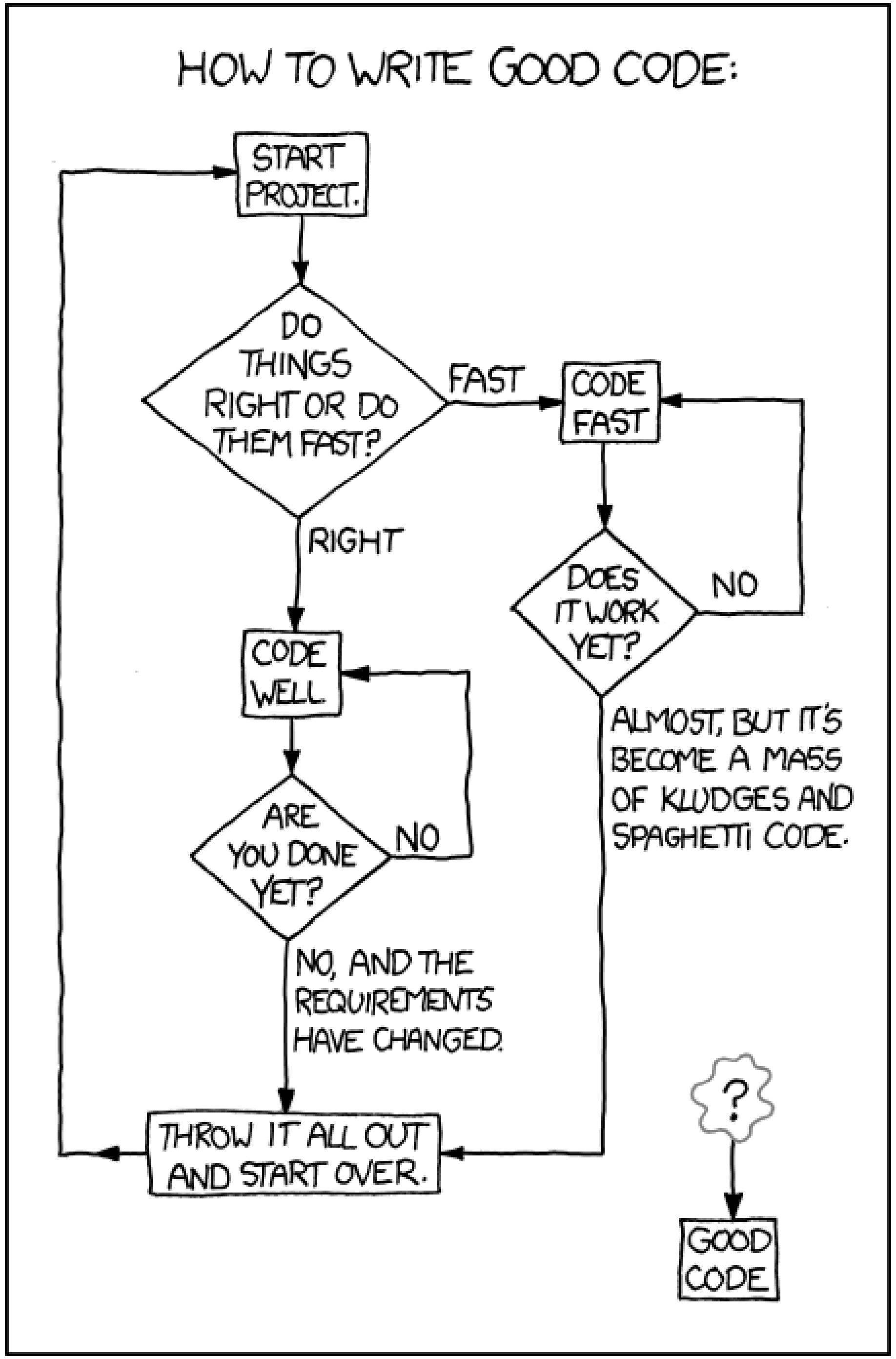


0

Intro

Analiza Algoritmilor

Source: https://xkcd.com/844/

Course structure

Fundamentals of characterizing and thinking about algorithms

Mathematical approach: formal definitions and rigorous proofs

Half of a complete algorithms course (+ PA on 2^{nd} semester)

Syllabus – 1

Computability theory:

- What problems can(not) be solved by computers?
- Formal model of computation: Turing Machines
- Unsolvable problems, the halting problem
- How to prove problems are (un)solvable

Syllabus – 2

Complexity theory:

- What does it mean for an algorithm to be “efficient”?
- What does it mean for a problem to be hard/easy?
- Bachmann-Landau notation ($O(n), O(n^2), O(n \log n)$)
- $P \ vs. NP$

Syllabus – 3

Recurrence relations:

- How to determine complexity of recursive algorithms

Amortized analysis:

- How to determine complexity of sequences of operations

Grading

Semester activities: **60 points**

Minimum 30 points required.

- Assignments: **20 points**
- Midterm exam: **30 points**
- Lab attendance: **10 points**

Exam: **40 points**

Minimum 20 points required.

Lecture

Very visual, few formal definitions and rigorous proofs

Slides and lecture notes available on moodle/wiki

Q&A strongly encouraged

Quick quizzes + programming challenges

Labs (10 points)

12 labs

1 point per lab

Physical presence + active engagement

Assignments (20 points)

Two assignments, **10 points each**

First assignment:

Turing Machines

~3rd week – 5th week

Second assignment (+ contest):

SAT Solver

~7th week –

Midterm exam (30 points)

15 multiple-choice questions**,** 2 points per questions

60 minutes

Either:

- **Saturday, 7 DEC, 11:00 AM**
- **Wednesday, 4 DEC, 20:00 PM**

Final exam (40 points)

You need **a minimum of 30 points** in semester activities to take the exam

You need **a minimum of 20 points** on the exam in order to pass

Task: try implementing a solution for one of the following three problems

Problem 1: Pairs

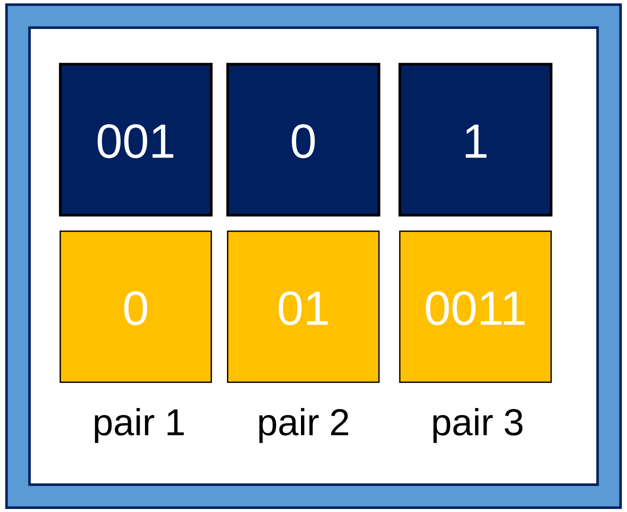
001	0	1
0	01	0011
pair 1	pair 2	pair 3

One pair = one “blue” word,
one “yellow” word

001	0	001	1
0	01	0	0011
pair 1	pair 2	pair 1	pair 3

Problem 1: Pairs (2)

Input:



Output: yes (we can find a sequence of matching pairs) / no (otherwise)

Problem 1: Pairs (3)

pair 1	pair 2	pair 3
100	0	1
1	100	00

Output: yes

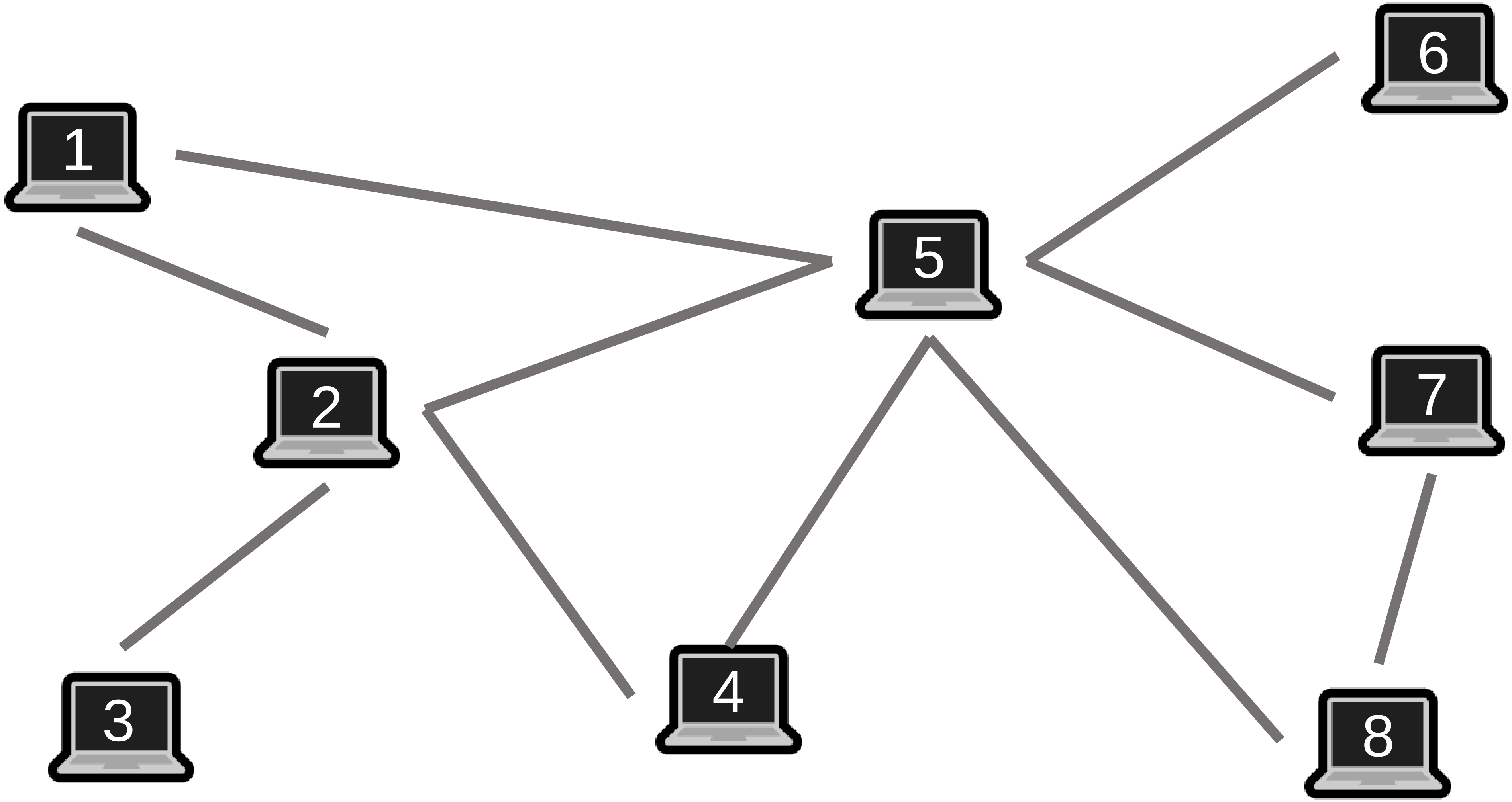
pair 1	pair 3	pair 1	pair 1	pair 3	pair 2	pair 2
100	1	100	100	1	0	0
1	00	1	1	00	100	100

Problem 1: Pairs (4)

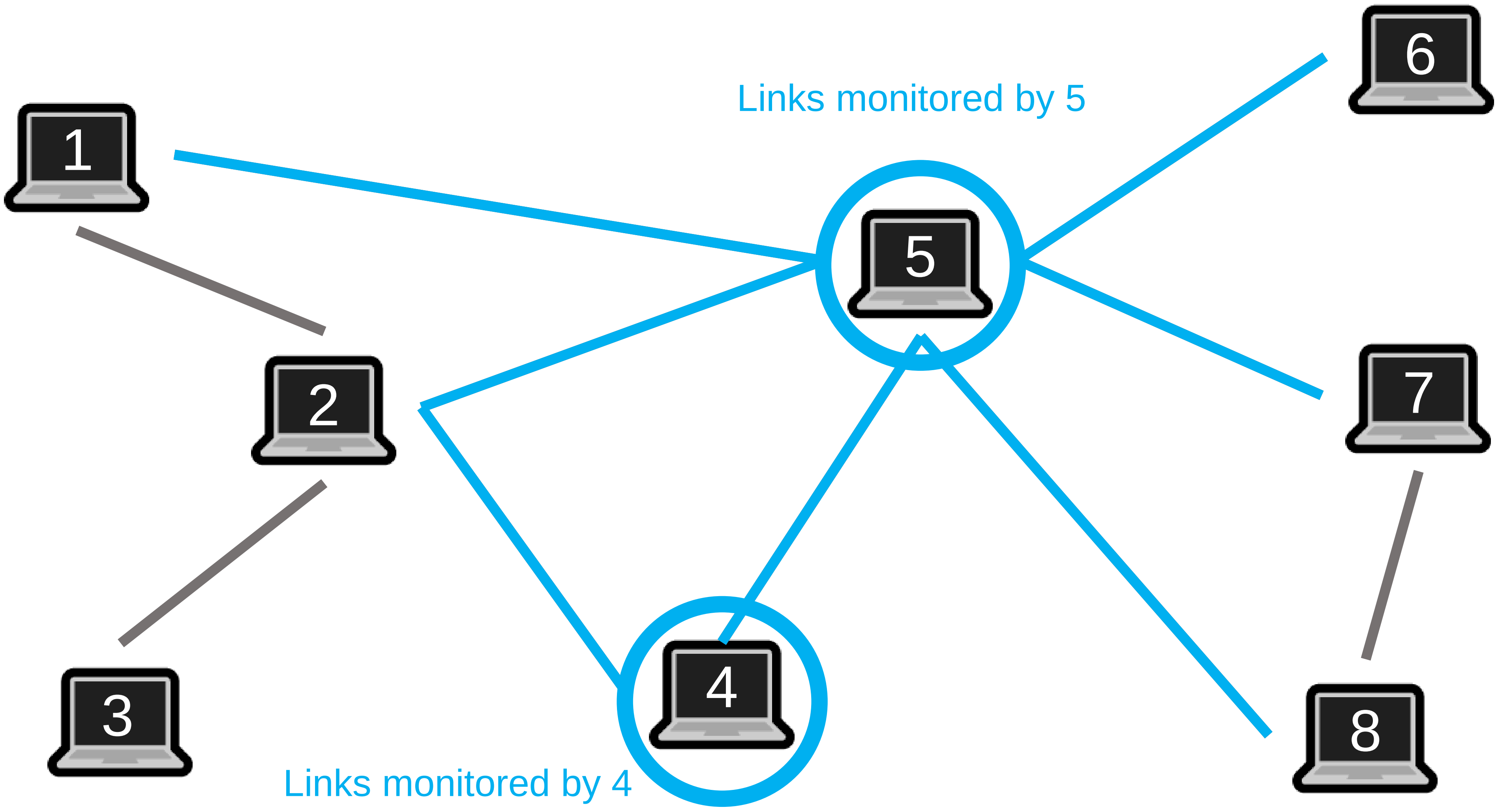
pair 1	pair 2	pair 3
1	0	101
0	001	1

Output: no

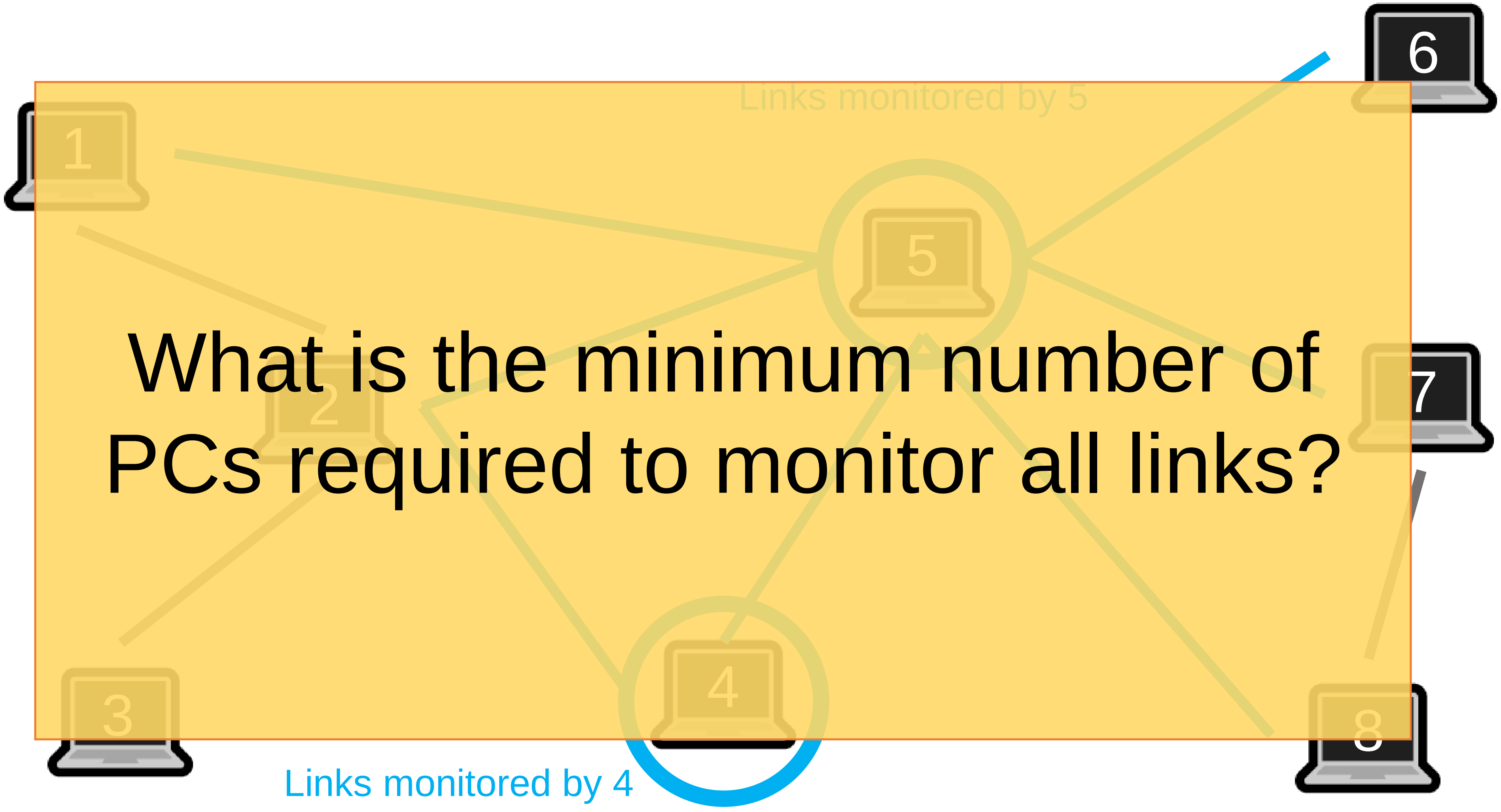
Problem 2: Monitor



Problem 2: Monitor

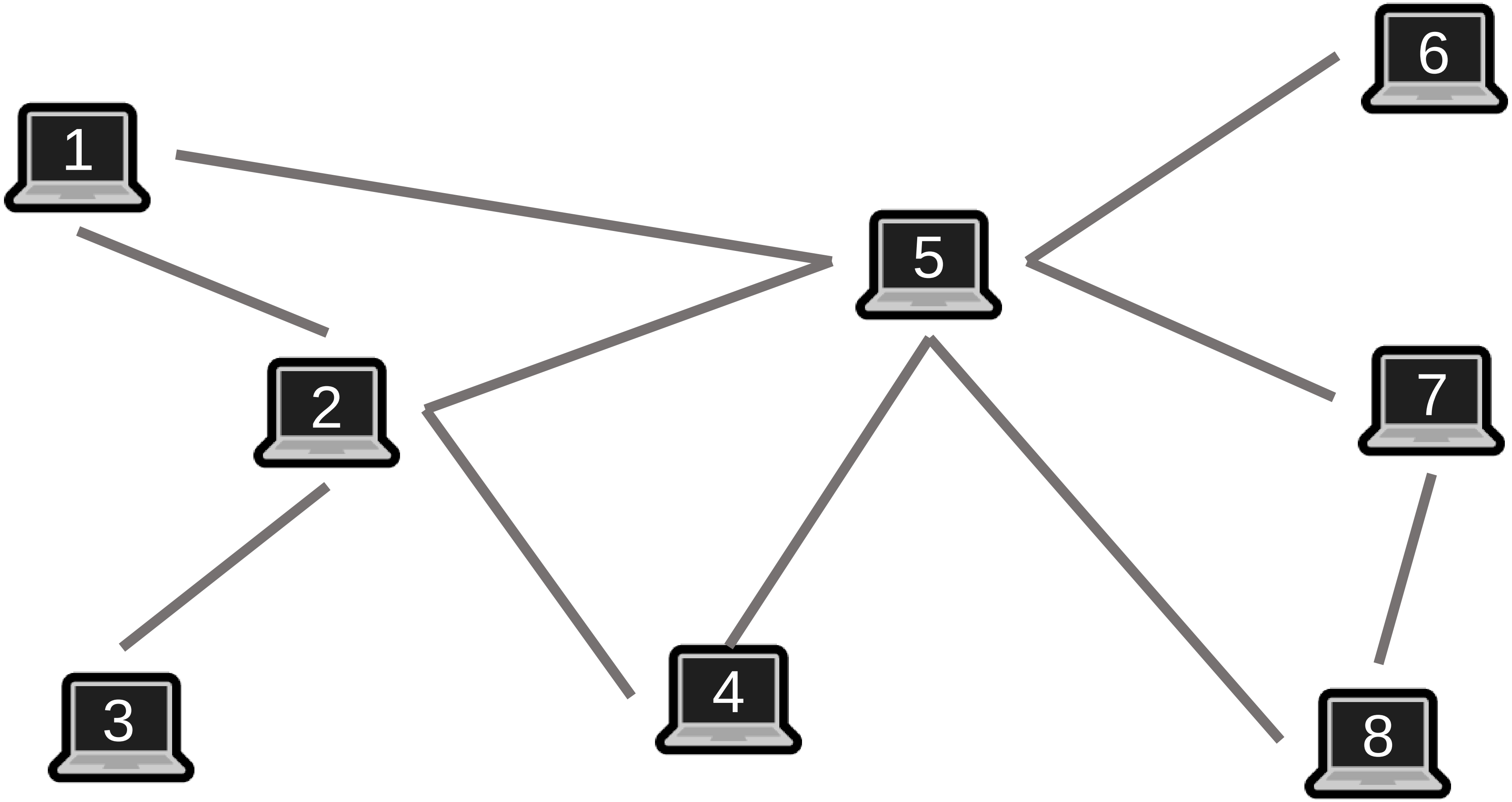


Problem 2: Monitor



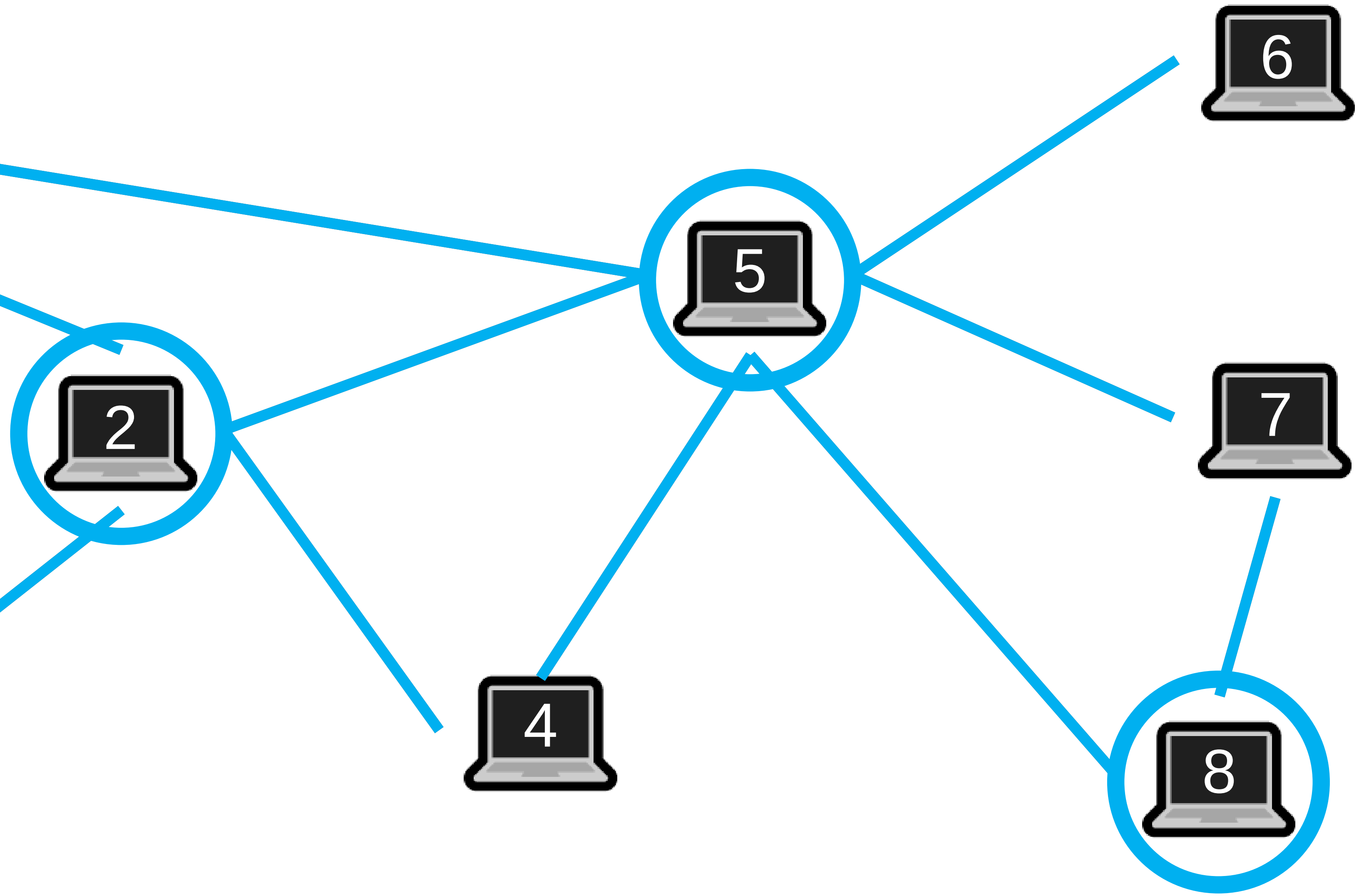
What is the minimum number of PCs required to monitor all links?

Problem 2: Monitor



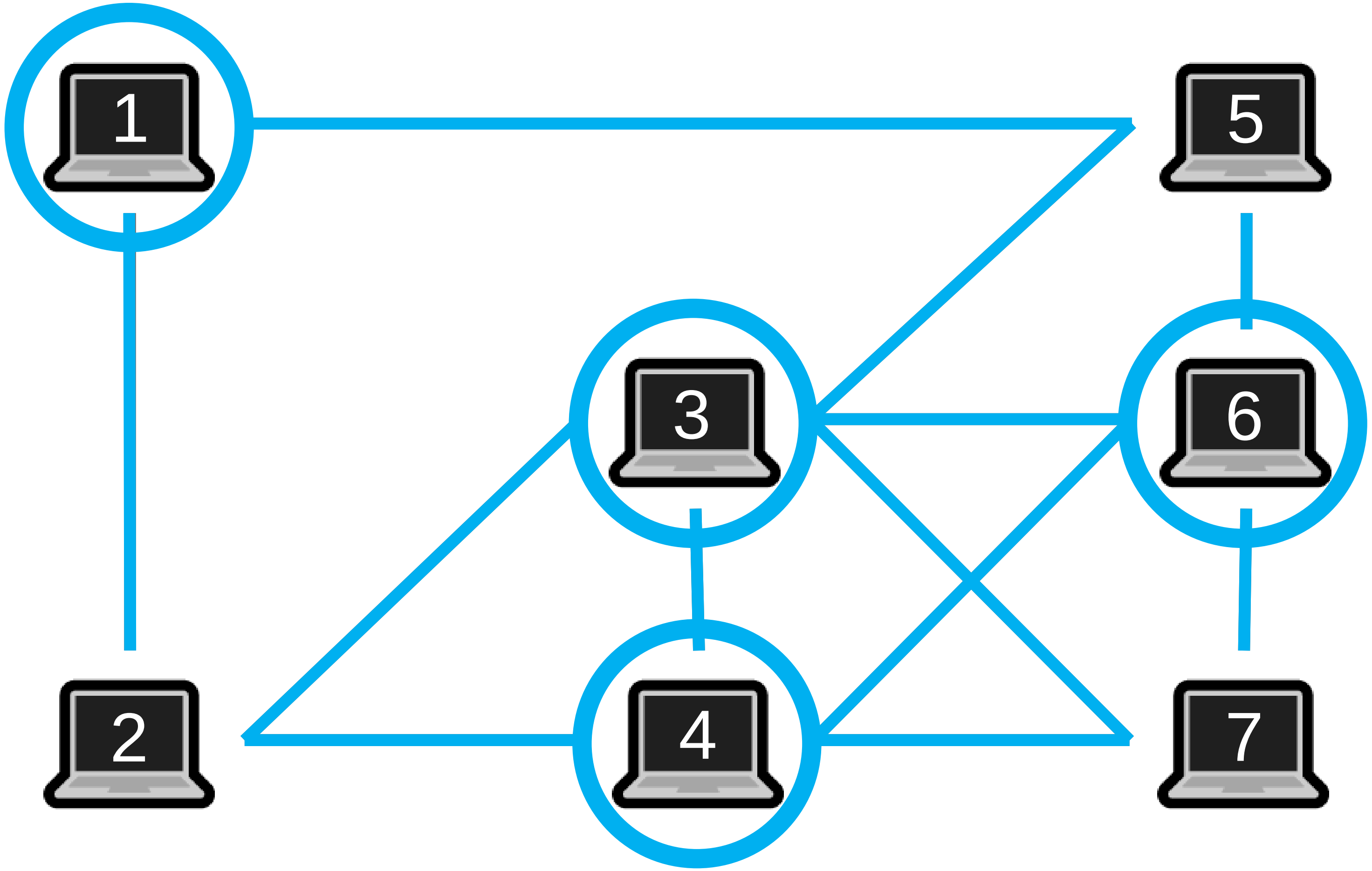
Problem 2: Monitor

Output: 3



Problem 2: Monitor (2)

Output: 4



Problem 3: Decoding

Input:

A	101
B	11
C	011
D	1

enc(**AAB**) = 10110111

Is it possible to decode 10110111 as something else?

dec(10110111) = **DCCD**

dec(10110111) = **AAB**

Output: yes

Output: is it possible to decode some sequence in (at least) two different ways?

Problem 3: Decoding (2)

A	00
B	01
C	10
D	11

Output: no

Summary

1

Pairs

Is there a matching sequence of pairs?

2

Monitor

What is the minimal number of PCs to monitor all links?

3

Decoding

Is there a sequence which can be decoded in different ways?

Optional Task

2 bonus points for exam (40 points total)

Choose **one problem** and solve it

All honest attempts will get the points

Upload the solution to moodle

Deadline: Tuesday, 08 OCT 2024, 14:00