CS-524: Computational complexity

Mika Göös

EPFL School of Computer and Communication Sciences

Intro Lecture, 11.09.2025

Who am I (the course responsible)?

- Mika Göös
  - *mika.goos@epfl.ch*
  - *https://theory.epfl.ch/mika/*
- Faculty of computer science
  - Research in *Complexity Theory*

CS-524 course overview

What is CS-524?

- Master level course
- Mathematical — problem solving, ability to write proofs. Must attend exercise sessions and solve homework problems...
- Challenging — abstract thinking, how to reduce one problem to another?

CS-524 course overview

What is CS-524?

- Master level course
- Mathematical — problem solving, ability to write proofs. Must attend exercise sessions and solve homework problems...
- Challenging — abstract thinking, how to reduce one problem to another?

> *What are the fundamental **limitations** of efficient algorithms?*

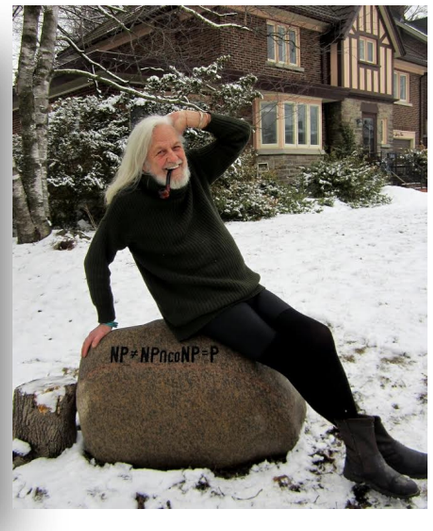
This question goes back to the 1930s...

1930s — Computability

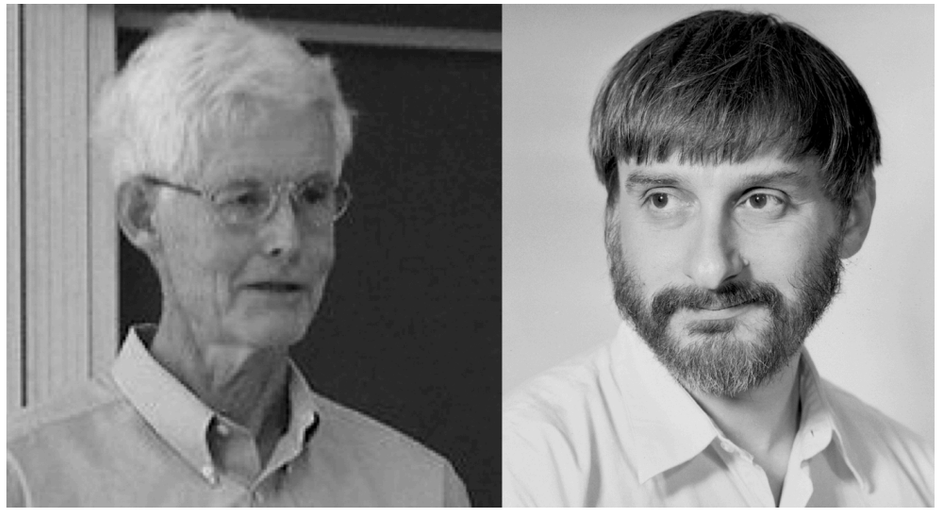
Gödel: What can be mathematically proved?

Turing: What can be computed?

?


NP≠NP∩coNP=P

?



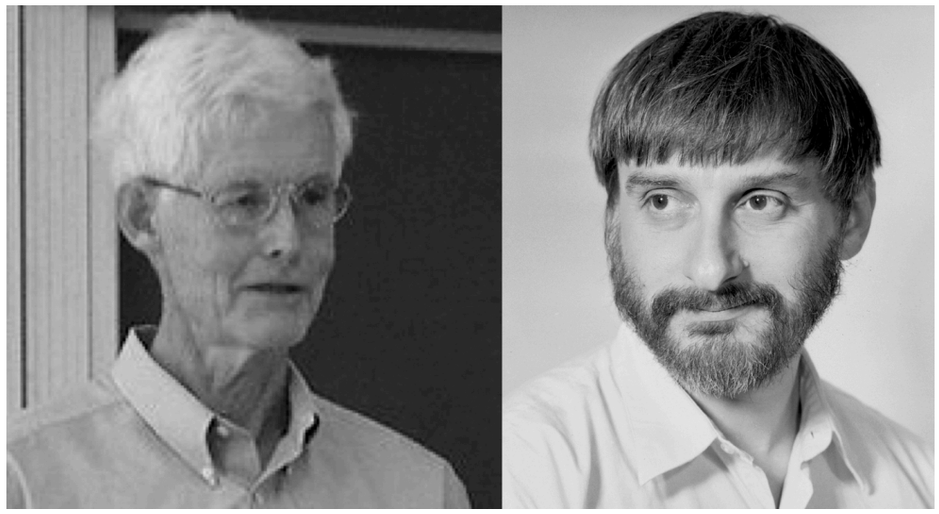
?


SIMONS INSTITUTE
FOR THE THEORY OF COMPUTING
THE NORTH FACE

?

Alan Turing

?



?

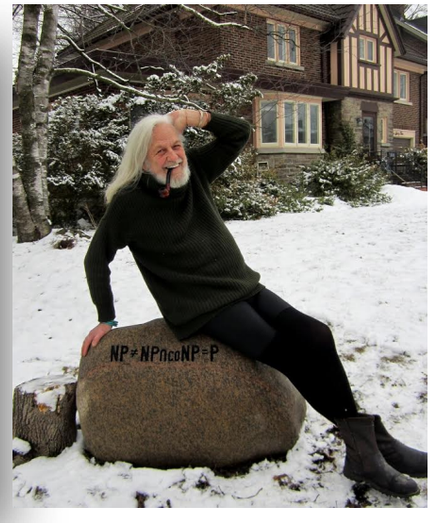
?

Seen as the father of computer science

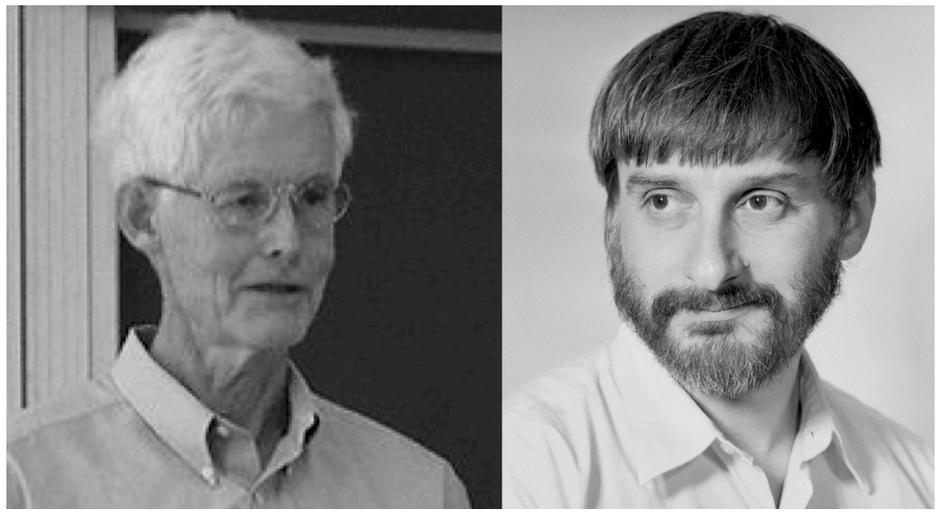
"On Computable Numbers, with an Application to the Entscheidungsproblem"

- Introduced "Universal machine" that is capable of computing anything that is computable
- "Central concept of modern computer" was due to this theoretical paper published in 1936

Alan Turing



Jack Edmonds



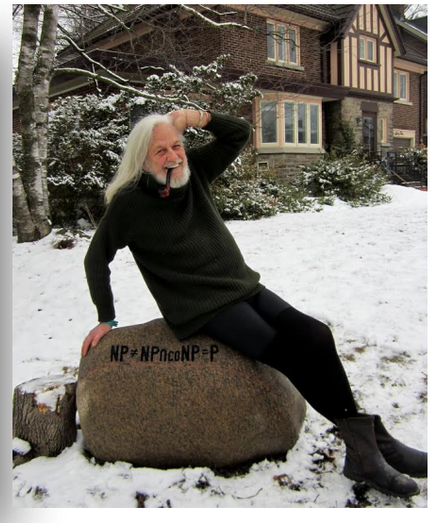
?

?

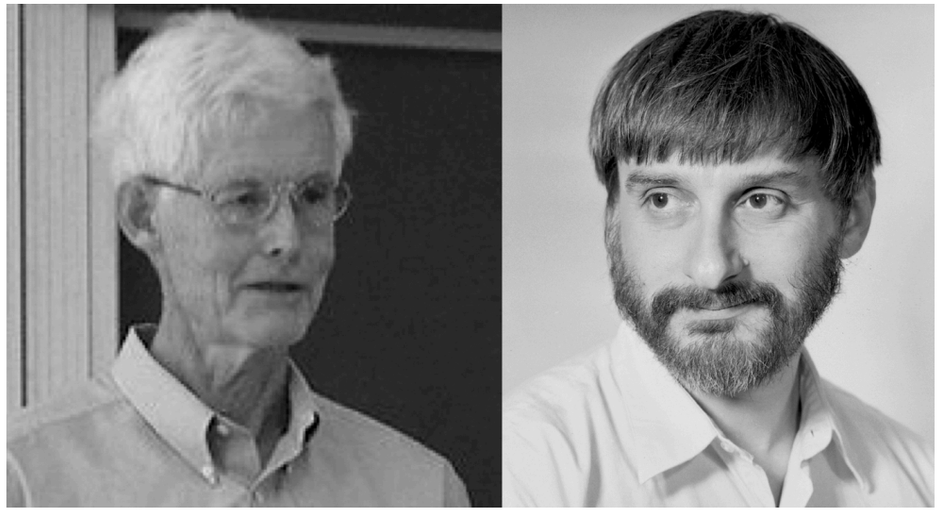
Introduced the class **P** — the idea that practical computation is polynomial time computation

In the first fast algorithm for the matching problem published in 1965

Alan Turing



Jack Edmonds



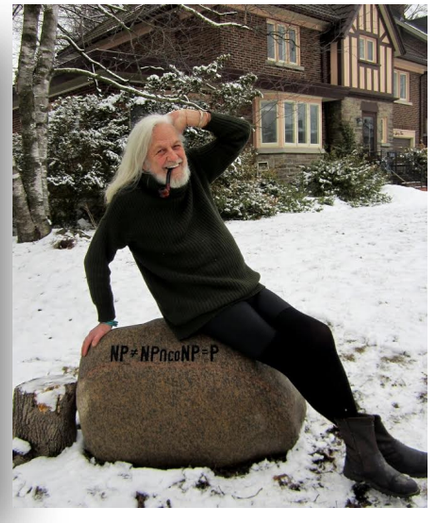
Stephen Cook & Leonid Levin

?

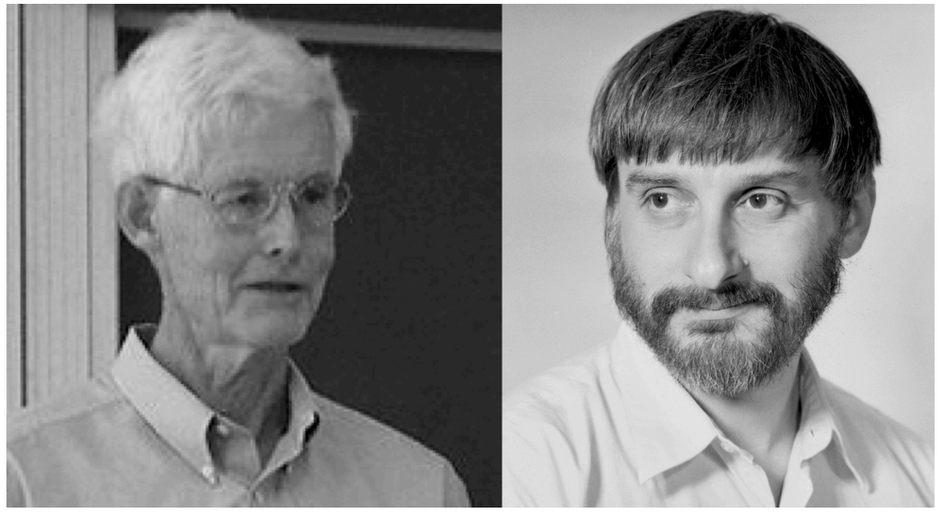
Introduced the concept of **NP**-complete problems

- Stephen Cook was denied tenure at Berkeley (1970);
  Invented **NP**-completeness (1971); Won Turing Award (1982)
- Leonid Levin's seminal paper obscured by Russian and Cold War

Alan Turing



Jack Edmonds



Stephen Cook & Leonid Levin

Avi Wigderson

Modern era

- Won the Abel/Turing prizes in 2021/23
- Karchmer–Wigderson games

The Computational Universe

Intro Lecture, 11.09.2025

The Computational Universe

What computational resources do we need to solve this problem?

How much time?

How much space?

How much communication?

The Computational Universe

A

B

What are the relations between different problems and between different computational models?

Does randomness help? Quantum?

If I can solve problem A, can I solve problem B?

The Computational Universe

Test whether a computer program finishes running or continues forever

Undecidable

Satisfiability

Graph Coloring

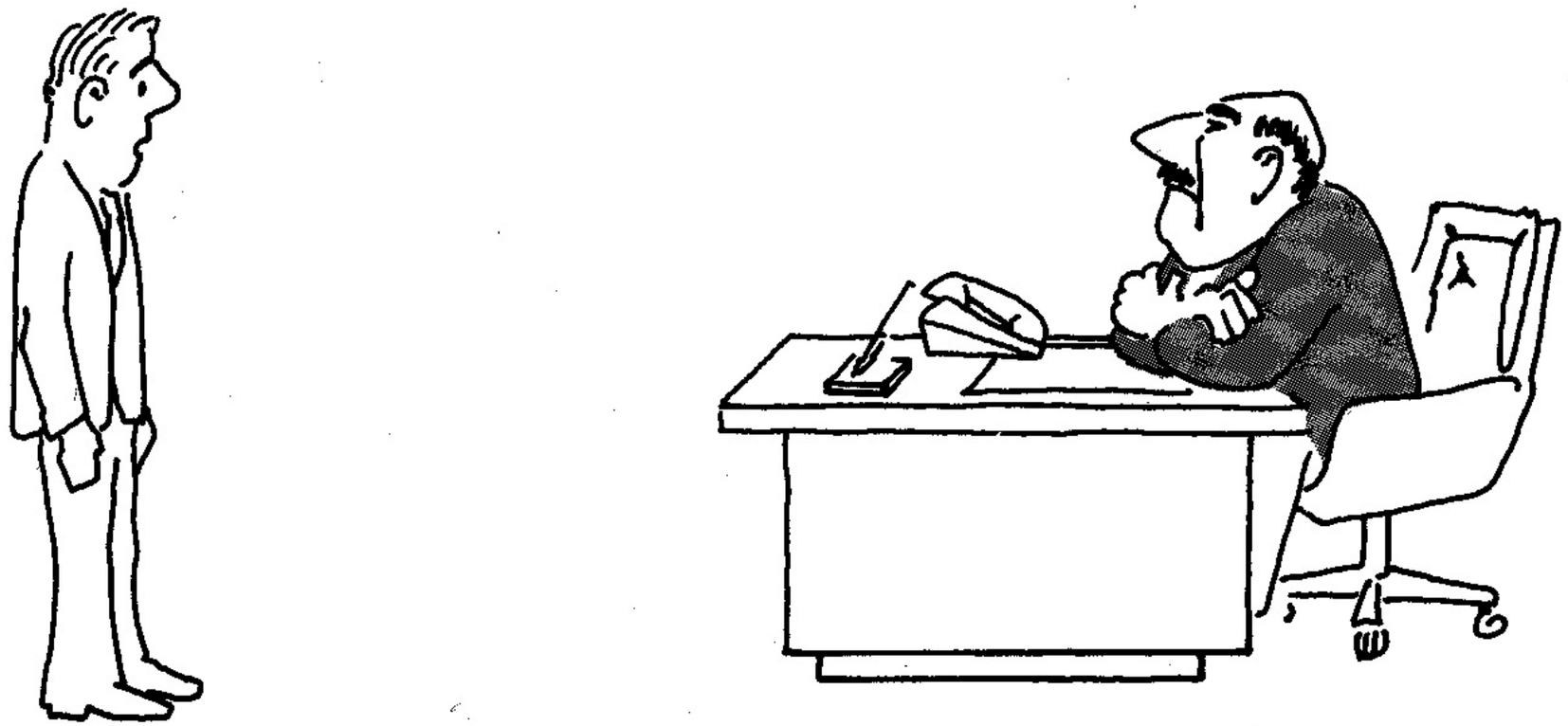
Traveling salesman problem

NP
(efficient verifiable)

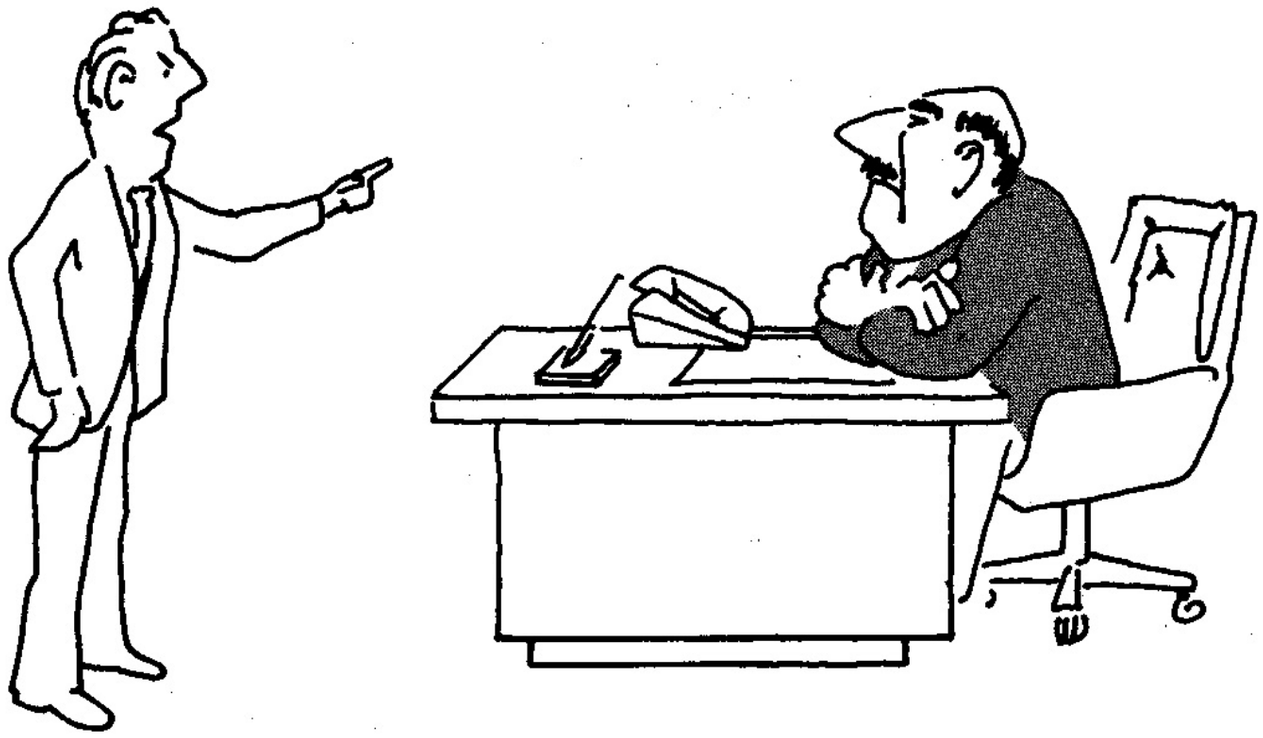
P
(efficient solvable)

Sorting

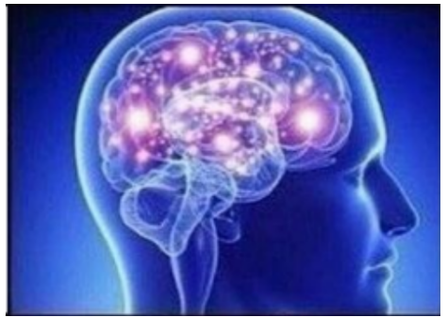
Shortest path

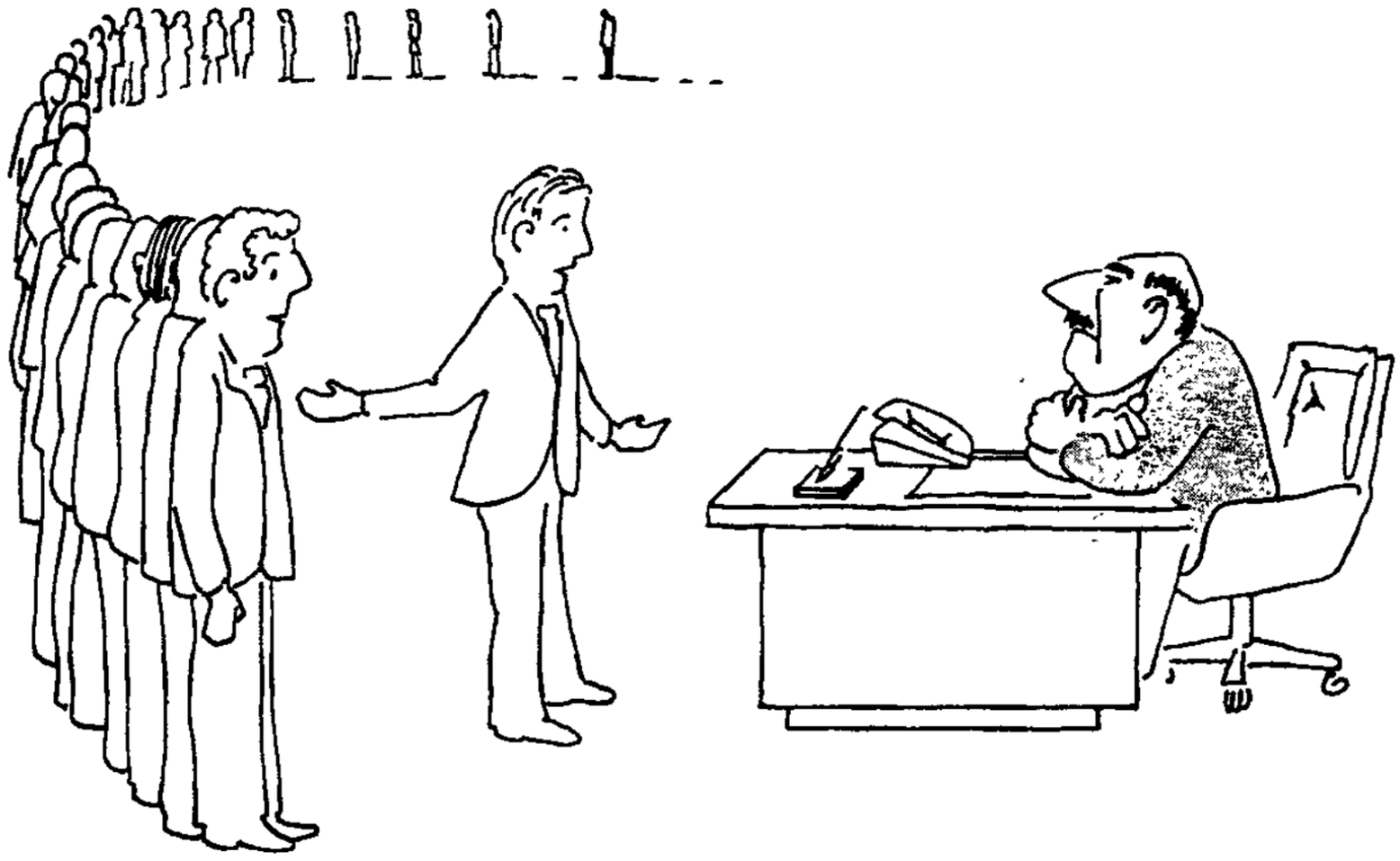


"I can't find an efficient algorithm, I guess I'm just too dumb."



"I can't find an efficient algorithm, because no such algorithm is possible!"





"I can't find an efficient algorithm, but neither can all these famous people."

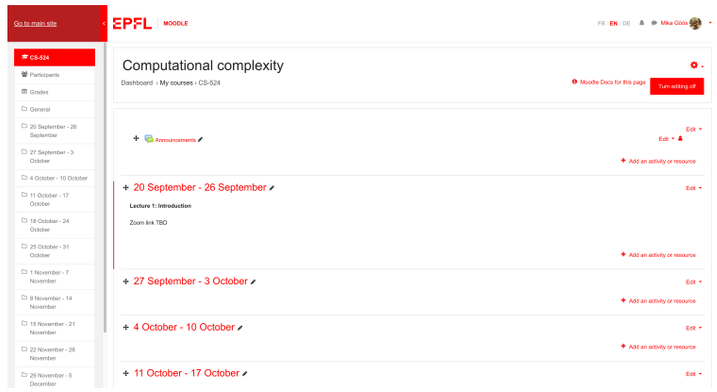
THE COURSE ESSENTIALS

Course staff

TAs: Valentin Imbach, 2th year PhD
Artur Riazanov, *n*-th year PhD

Resources

Moodle



Dynamic: course material (these slides will be there), exercises+solutions, etc.

Ed Forum for questions!

Recommended textbooks

- *Computational Complexity: A Modern Approach* by Arora & Barack (draft available online)
- *Nature of computation* by Mertens & Moore

Office hours: On demand (via email)

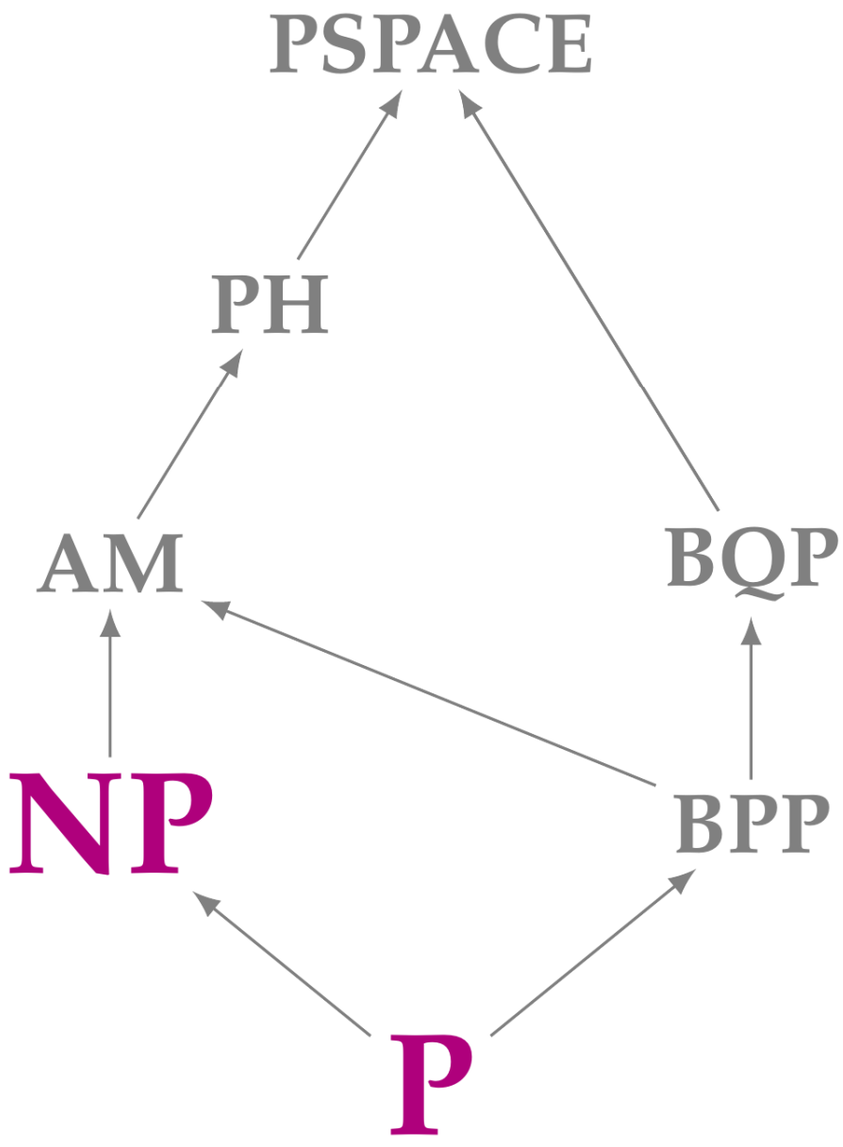
Grading

30% — 3 sets of homeworks in groups of 1–2 students

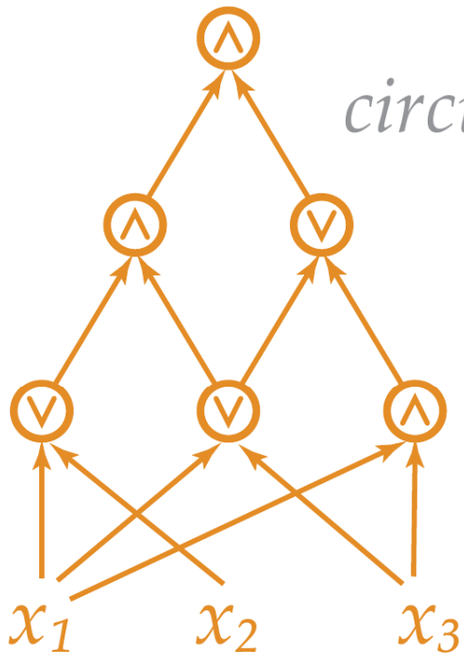
70% — Final exam in December (date TBD)

Course content

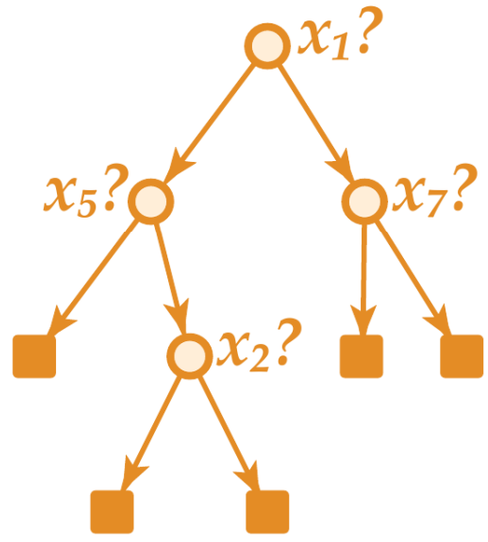
Structural



Concrete



circuits



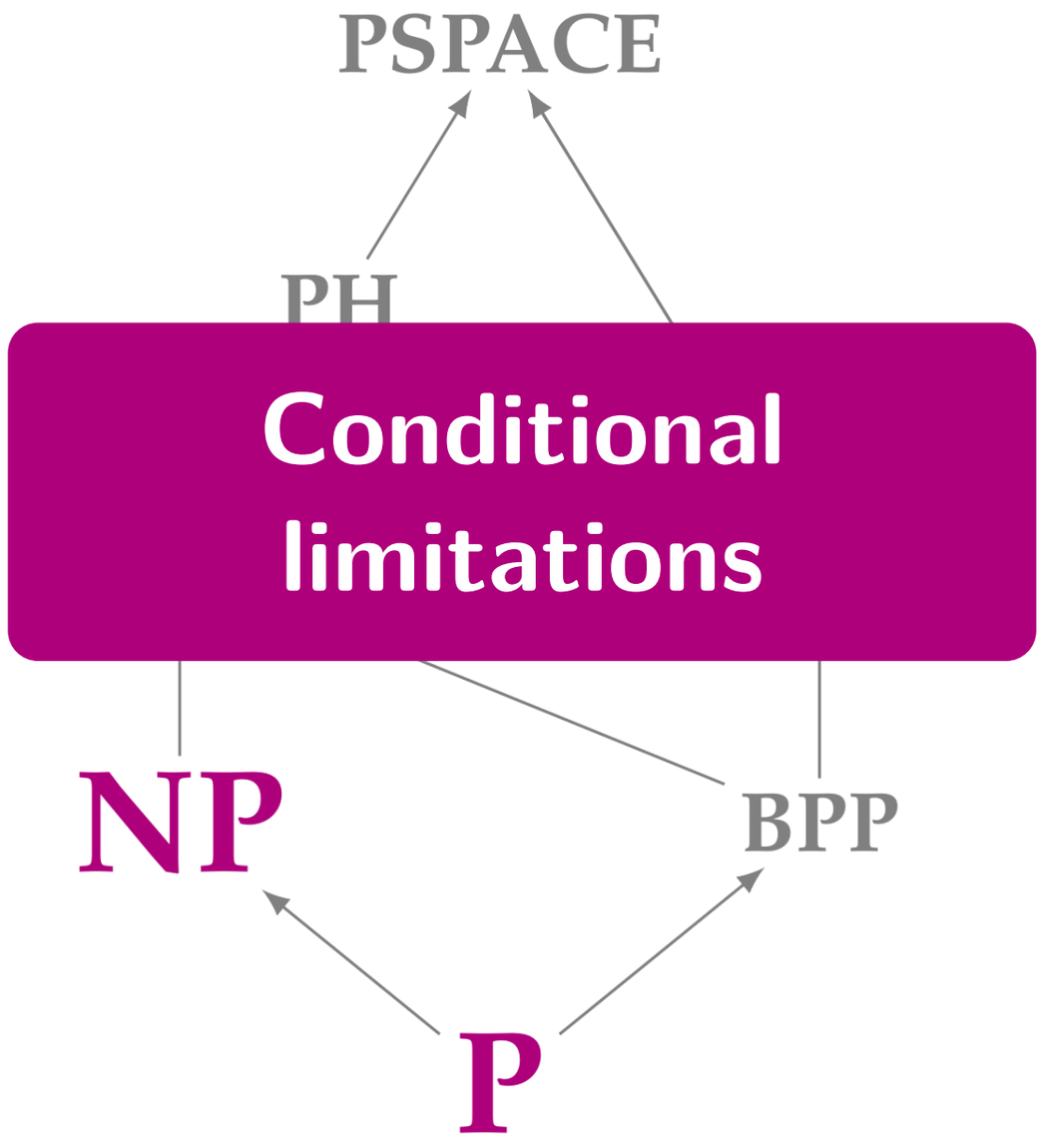
decision trees

proof systems

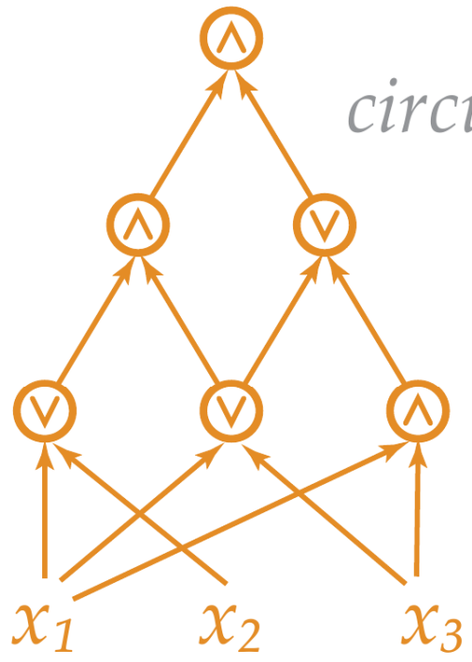
$$\frac{x_1 \qquad \overline{x_1} \vee x_2}{x_2}$$

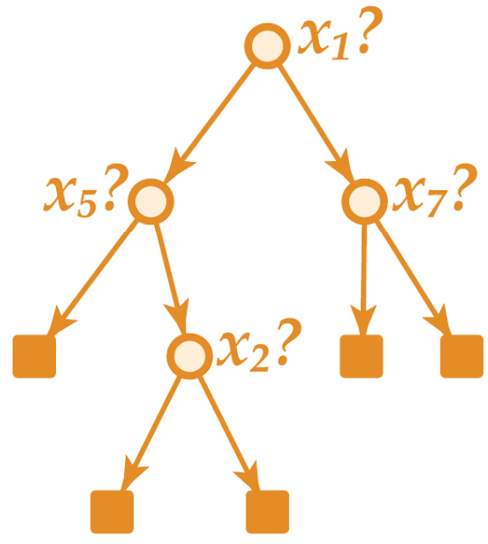
Course content

Structural



Concrete





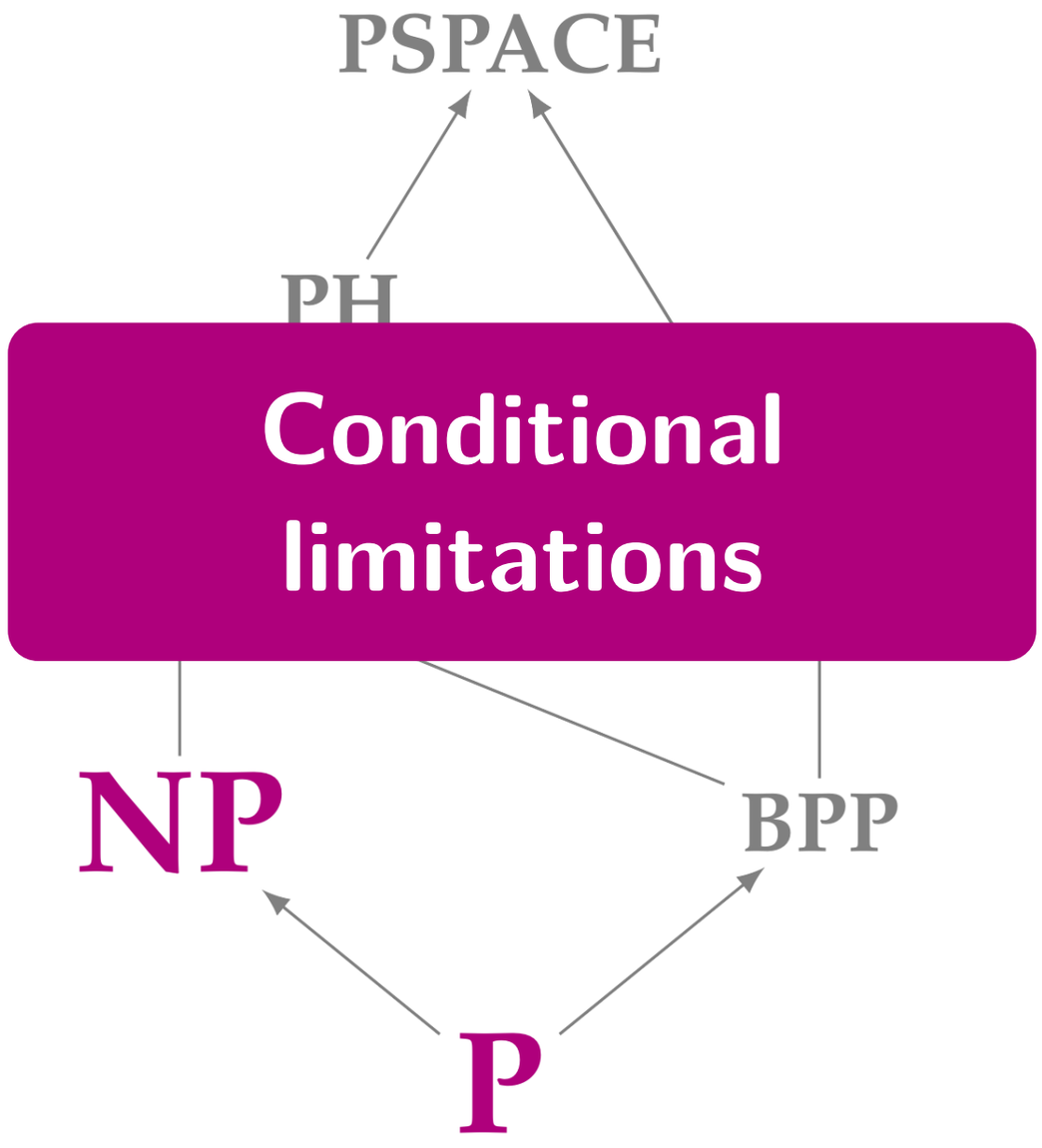
decision trees

proof systems

$$\frac{x_1 \qquad \overline{x_1} \vee x_2}{x_2}$$

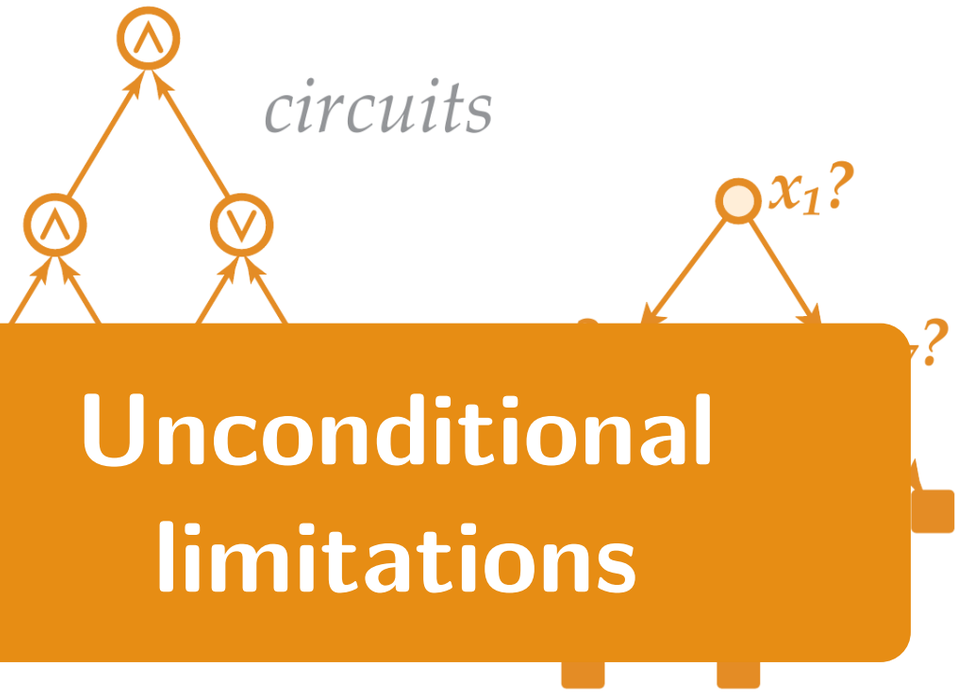
Course content

Structural



Conditional limitations

Concrete



Unconditional limitations

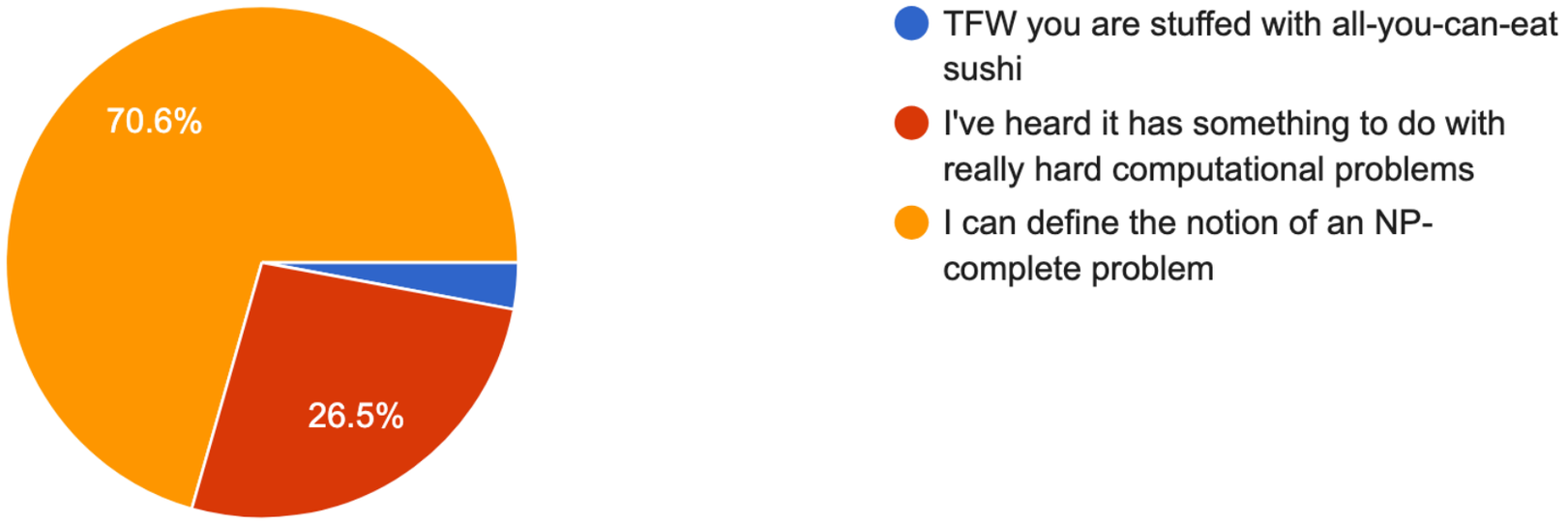
proof systems

$$\frac{x_1 \qquad \overline{x}_1 \vee x_2}{x_2}$$

decision trees

What is **NP**-completeness?

34 responses



Cook–Levin theorem?

34 responses

Let's get started!