T-79.5103 Computational Complexity Theory

Lecturer: Prof. Pekka Orponen

Aalto University
School of Science
Department of Computer Science

Spring 2016
For a computer, how hard is it ...

to sort an array of integers?

| 10 | 2  | 100 | -2 | ... | 5  | -2 | 11 |
For a computer, how hard is it ...

to verify that a C++ program can never segfault?

... 
for(unsigned int i = cell->length; i > 0; i--, ep++) {
    if((int)(*ep) > info.split_element &&
        *ep < next_split_element &&
        best_path_orbits.is_minimal_representative(*ep) &&
        (!opt_use_long_prune ||
            info.long_prune_redundant.find(*ep) ==
            info.long_prune_redundant.end())) {
        next_split_element = *ep;
        next_split_element_pos = ep;
    }
}
...
For a computer, how hard is it ...

to solve sudokus?

to find shortest Travelling Salesperson tours?

http://www.math.uwaterloo.ca/tsp/world/countries.html
For a computer, how hard is it ... to efficiently solve sudokus?

to efficiently find shortest Travelling Salesperson tours?

http://www.math.uwaterloo.ca/tsp/world/countries.html
For a computer, how hard is it ... to determine whether the white player has a winning strategy in a given Go configuration? to deduce whether an output line of a sequential circuit can ever be active?
Weekly Sessions and Course Personnel

- **Lectures:** Mondays & Wednesdays 10–12, T3
  First lecture Mon 11 January

- **Lecturer:** Prof. Pekka Orponen
  e-mail: pekka.orponen@aalto.fi

- **Homework assistants:**
  M.Sc. Tuomo Lempiäinen and M.Sc. Abdulmelik Mohammed

- **Course home page:**
  - https://mycourses.aalto.fi/course/search.php?
    search=T-79.5103
  - Lecture slides, homework assignments, current info etc. uploaded as the course progresses.

- **Old home page:**
  - https://noppa.aalto.fi/noppa/kurssi/t-79.5103/
  - Lecture slides & homework assignments from 2015, archives for older course materials.
General Themes

- Concepts and phenomena of computational complexity
- Identification of computationally hard problems
- Classification of problems according to their complexity
- Structure of the universe of complexity classes
Topics

- Models of computation: Turing machines, RAM machines, Boolean circuits
- Modes of computation: deterministic, nondeterministic, parallel, randomised, alternating
- Basic notions of computational complexity: complexity measures and complexity classes, hierarchy theorems, reductions between problems, completeness
- Central complexity classes: P, NP, PSPACE, NC, polynomial time hierarchy, etc.
- Design of completeness proofs, especially for NP-completeness
- Applications and extensions: approximability, counting, games, cryptographic protocols


Prerequisites: T-79.1001 Introduction to Theoretical Computer Science
Learning Outcomes

- Once you have taken the course, you master the key complexity classes, their underlying models of computation, and relationships.

- You are able to formalise and abstract from a given computational task relevant computational problems and argue that they belong to appropriate complexity classes.

- You understand the concept of reductions and how it can be used to order problems by their computational complexity. You are able to show using reductions that a problem is complete for a central complexity class (such as NP) and you understand the importance and implications of such a result.

- You are familiar with the concepts of randomised, approximation, and parallel computation and aware of related complexity classes and their relation to other complexity classes and their models of computation.
Course Requirements

In order to pass the course you need to:

1. pass the first midterm exam (Mon 1 Feb 2016),
2. pass the home assignments (seven rounds; start after the first midterm),
3. pass the second midterm exam (Tue 5 Apr 2016).

- You must register for the course in Oodi.
- You must pass all three parts.
- The grade of the course (0–5) is determined by the respective grades of (i) the first midterm exam (20%), (ii) homework (40%) and (iii) the second midterm exam (40%).
- By submitting the course feedback form you earn 1 extra homework bonus point.
Homework Assignments

- There are seven home assignments on the course, with two problems each. The home assignments are published, submitted and graded weekly, following the first midterm exam. Further instructions on the process will be provided on the problem sheets.
- The first set of problems will be published by 11 February, with a return deadline of 21 February.
- Each assignment problem is graded on a scale of 0–4. If you get fewer than 3 points for some problem, you then have the possibility to revise your solution.
- The deadline for submitting revised solutions is 13 April 2016. Revised solutions will be graded on a scale of 0–3.
- The credit points from the assignments (AP = 0–56) are translated into course grade points (CP = 0–24) by the formula: \[ CP = 24 \times \left( \frac{AP}{56} \right), \] rounded up.