CS-251  Theory of computation

Svensson Ola Nils Anders

<table>
<thead>
<tr>
<th>Cursus</th>
<th>Sem.</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>HES - IN</td>
<td>E</td>
<td>Obl.</td>
</tr>
<tr>
<td>Informatique</td>
<td>BA4</td>
<td>Obl.</td>
</tr>
<tr>
<td>Mineur en Informatique</td>
<td>E</td>
<td>Opt.</td>
</tr>
<tr>
<td>Systèmes de communication</td>
<td>BA4</td>
<td>Obl.</td>
</tr>
</tbody>
</table>

Summary
This course constitutes an introduction to theory of computation. It discusses the basic theoretical models of computing (finite automata, Turing machine), as well as, provides a solid and mathematically precise understanding of their fundamental capabilities and limitations.

Content
- Basic models of computation (finite automata, Turing machine)
- Elements of computability theory (undecidability, reducibility)
- Introduction to time complexity theory (P, NP and theory of NP-completeness)

Keywords
theory of computation, Turing machines, P vs. NP problem, complexity theory, computability theory, finite automata, NP-completeness

Learning Prerequisites
Required courses
CS-101 Advanced information, computation, communication I
CS-250 Algorithms

Learning Outcomes
By the end of the course, the student must be able to:
- Perform a rigorous study of performance of an algorithm or a protocol
- Classify computational difficulty of a decision problem
- Define the notion of NP-completeness
- Analyze various computation models
- Design a reduction between two computational problems
- Characterize different complexity classes
- Explain P vs. NP problem

Transversal skills
Teaching methods
Ex cathedra with exercises

Assessment methods
Written exam and continuous control