Introduction to the theory of automata and computation

MORI, Takuo

1. Course Description

Recently, computers get to win each world champion in Chess, Japanese Chess or the game of Go. It has been common to use computers to solve problems which seem insolvable to human beings, by using huge data called the Big-data, and it has been common to use the Artificial Intelligence (AI) in our daily life.

In this course, students learn the meaning of computations by computers, and whether computations by computers have any bounds or not. To deal with computations by computer mathematically, we model a computer itself in simple forms, which is called a computational models or an automaton. Given input data and a program, it becomes possible for a computer to compute the program, thus, we also model input data and programs in simple forms, which are called languages in this field. A language is a set of a series of symbols which obey a rule. The rule is called a grammar. It is known that an automaton corresponds to a grammar and the pair of an automaton and a grammar makes a hierarchical structure.

For each pair of an automaton and a grammar, we will show its equivalency or non-equivalency by showing whether a grammar can generate a language, where the language is accepted by an automaton. In addition, as for an automaton called Turing machine, which belongs to the highest class of the hierarchy, we will show that there are problems which cannot be solved essentially by computers by showing its bound.

Moreover, by using the concept of complexity and Turing machines, we learn that problems dealt in the computer science can be classified into computational ones and non-computational ones. We also deal the relationship between computational ones and non-computational ones and these structures.

We also learn the relationship between Turing machines and logical circuits and their equivalency.

Students acquire skills related to the diplomatic policy 2 of Department of Information Science Correspondence Course.

2. Course Objectives

The goal of this course is that students master the following abilities;

Students can trace actions of simple automata giving an input series of symbols and state transition functions of automata.
Students understand the equivalency between deterministic and (generalized) non-deterministic finite automata and regular expressions or regular grammar, and can utilize transform algorithm for each of them.
Students can express the features of regular languages as an algebraic system.
Students understand the meaning of the Chomsky normal form and given production rules of an arbitral context-free grammar, students can transform it to the corresponding Chomsky normal form.
Students can explain the Chomsky Hierarchy.
Students can explain the non-equivalency between deterministic and non-deterministic pushdown automata.
Students can explain the equivalency between pushdown automata and context-free grammars and can utilize algorithms which translate one to another.
Students can explain the equivalency of deterministic and non-deterministic Turing machines, any other variations of Turing machines and logical circuits, the universality and the bound of Turing machines.
Students understand the concept of computational complexity and can express the complexity of each problem, algorithm with Big-O notation.
Students can explain the computational complexity and the relation of P, NP, NP complete and NP hard as classes of complexity, and can express the definition of NP.

3. Grading Policy
Grading policy:
Midterm report(50%), Examination(50%).

The way of feedback;
Answers for questions or feedback for the contents of classes and examination will be given in a class, through LMS or during office hours.

4. Textbook and Reference
Textbook
丸岡　章著　計算理論とオートマトン言語理論　サイエンス社、ISBN-13: 978-4781911045

5. Requirements(Assignments)

Students can hardly earn credits not submitting the mid-term report. Thus, it is expected for students to observe the deadline.
As for the self-learning support, students are expected to utilize materials, such as slides, handouts and quizzes on the LMS.

Before taking this course, students should take the following courses;
Mathematical Logic, Logic Circuits, Data Structure and Algorithms and Programming Language Theory

At the same semester with this course, students should take Graph Theory.

After taking this course, students should take computer architecture, information security.

If a student has a question on quizzes or mid-term report or examinations, ask the question in the class or in office hours or through LMS.

This course is a required course, and relates to the mid term 4-1 of the attaining targets for learning and educating, in the JABEE program.

6. Note

7. Schedule

[1]	Introduction Everything begins from computations
[2]	Concepts and technical terms on the theory of automata and computation
[3]	Deterministic finite automata
[4]	Nondeterministic finite automata, the equivalency between deterministic finite automata and nondeterministic finite automata
[5]	Regular expressions
[6]	Regular languages/Context-free languages, Pumping lemma for regular languages
[7]	Context-free grammars, generation and acceptance, Chomsky normal form
[8]	Pushdown automata1 A decision problem for context-free grammars, CYK algorithm, pushdown automata
[9]	Pushdown automata2 The equivalency between pushdown automata and context-free grammars
[10]	Turing machines
[11]	Nondeterministic Turing machines
[12]	Computational universality of Turing machines, reduction, Post correspondence problem and its undecidability
[13]	Quantification of computational complexity based on Turing machines
[14]	Logic circuits and those complexities, complexity classes
[15]	Summary and examination