Computability theory

Spring 2018

Basic info

- Optional course (MAST31214) of the Master's Programme in Mathematics and Statistics
- Prerequisites: Introduction to logic I&II recommended
- Lectures: Monday 14-16 (B121) and Thursday 10-12 (B121)
- Exercise class: Tuesday 12-14 (C129)
- Assessment: final exam + exercises
- Teacher: Juha Kontinen (C333), juha.kontinen@helsinki.fi

Basic info cont.

Material:

- The main text book for the course is Computability: An Introduction to Recursive Function Theory by Nigel Cutland.
- There is a lot of material available (books and papers) on the topic of the course, e.g., the article The History and Concept of Computability by Robert I. Soare.
- Also excellent supplementary reading is the autobiography of Alan Turing by Andrew Hodges.

Examples algorithms

- The Sum, product, and greatest common divisor of two natural numbers:

1. Input natural numbers n, m
2. Output: n+m, nm, and the largest k that divides both n and m

- Decide whether n is a prime:

1. Input n
2. Output: “yes” or “no”

Examples cont.

- Compute the decimal expansion of π:

1. Input n
2. Output: n first digits of π

- Decide whether a propositional formula F is a tautology:

1. Input: a propositional formula
2. Output: “yes” or “no”

Contents (tentative):

- We study computable (effectively calculable) functions over the natural numbers.
- The borderline between computable and non-computable (undecidable) sets and functions is the main object of study unlike in the course Complexity theory which classifies computable problems according to resources (e.g., time, space) needed in their computation.
- Computability is formalized in terms of so-called URM register machines but also other models of computation will be discussed.
- The tentative plan is to cover chapters 1-7 of Cutland's book.

Historial remarks

- The informal notion of an algorithm or effective procedure is very old (e.g., Euclid's algorithm for finding the greatest common divisor of two numbers).
- During the first half of the 20th century several researchers aimed at formalizing for the notion of effective calculability: Church/lambda-calculus, Herbrand-Gödel/recursive functions, and Turing/Turing machine.
- Their work was in part motivated by Hilbert's Entscheidungsproblem, i.e., the question whether an effective procedure to decide first-order validity exists.

Historical remarks cont.

- Church-Turing Thesis was generally accepted only after Turing's introduction of Turing machines and the corresponding notion of computability.
- In his seminal article, Turing (1936) also showed the undecidability of Hilbert's Entscheidungsproblem. This required a mathematically precise notion of an algorithm.