Infinite Automata 2025/26

Course Introduction

Henry Sinclair-Banks

Practical details.

Lecturer	Henry Sinclair-Banks
Email	`hsb` *at* `mimuw` *dot* `edu` *dot* `pl` – *Please use "Infinite Automata" in the subject line.*
Tutorials	Wojciech Czerwiński
Webpage	https://mimuw.edu.pl/∼hsb/teaching.html
Starred exercises	Series 1 deadline: T.B.C. (late November 2025) Series 2 deadline: T.B.C. (January 2026)
Oral examination	Date: T.B.C. (shortly after the course in February 2026).

The course webpage is the main information source. Lecture notes, tutorial exercises, starred exercises, examination questions, dates, and deadlines will all be published on the course webpage. Important details will also be announced during the lectures.

General vision. This course will study a wide variety of models of computation most, if not all, of which will have infinite-state space. These models include Turing machines (which have an infinite tape), pushdown automata (which have an unbounded stack), counter machines (which have unbounded counters), and Petri nets (which can contain unboundedly many tokens). We will also study a wide variety of decision problems that one can use to determine properties of these models. These decision problems include reachability, coverability, and language equivalence. One of my core missions, as a researcher who studies infinite automata, is to find and analyse the most complex models of computation which are (for the most part) decidable. On the other hand, unfortunately, it often the case that simple models of computation are undecidable. Turing machines are a prime example: it is not even possible to decide whether a given Turing machine halts on a given input! When analysing (simpler) infinite-state systems, like pushdown automata and Petri nets, it becomes clear that one must develop strong mathematical tools when attempting to answer even some of the simplest questions. When decidability is given, we are then tasked with finding (the best possible) algorithms that can indeed solve decision problems of interest. Our mathematical tools will help us both in the design and analysis of our algorithms. I hope that by taking this course, you will be exposed to various fundamental models of computation, learn about a range of interesting mathematical tools, and develop an ability for analysing these models.

Examination format. The assessment methods for this course will be an oral assessment and (optional) starred exercises. The main examination will be an oral assessment. The length of the oral assessment will be between 40 and 60 minutes, depending on the number of enrolled students. The format of the oral assessment will adhere to the following design. There will be a set of *easier* questions (approximately 20) and a set of *harder* questions (approximately 5) that are published in advance of the end of the course. During the exam, a student will be asked a small *unknown* subset of the easier questions and the student will be asked to present a solution to a harder question of their choosing. Students will be scored on the quality of their solutions.

Starred exercises. There will be two series of starred exercises each containing between 3 and 5 starred exercises. Each starred exercise will have one score associated to it. Every student that submits a correct solution for a starred exercise will (uniformly) share that exercise's score. Starred exercises will be designed to be challenging, and hence, the starred exercises are optional. A student's final score for this course is the sum of the oral assessment score and the score earned from starred exercises.

2nd October 2025