Léonard Brice, Mathieu Sassolas
leonard.brice@ulb.be, mathieu.sassolas@ulb.be

Session 2:
Countability

Reminders

To compare the sizes of infinite sets, we use bijections, injections and surjections: two infinite sets A and B have "the same size" if there exists a bijection from A to B — or equivalently, from B to A. The set A is "greater" than B if there is a surjection and no injection from A to B — or equivalently, if there is an injection and no surjection from B to A.

A *countable* set is a set that is finite, or in bijection with $\mathbb{N}$. Equivalently, it is a set from which there exists an injection into $\mathbb{N}$; or a set onto which there is a surjection from $\mathbb{N}$. More intuitively, a countable set is a set that can be enumerated.

1 Countable sets

1. Show that the set of odd integers $\{2n + 1 \mid n \in \mathbb{N}\}$ is countable.
2. Show that the set $\mathbb{Z}$ is countable.
3. Show that the set of finite words Σ^* over a finite alphabet Σ is countable.

2 Uncountable sets

Let Σ be a finite alphabet with at least two symbols. We write Σ^ω the set of infinite words $w_0w_1\dots$ over Σ. We want to prove that Σ^ω is uncountable.

1. Consider a mapping $f : \mathbb{N} \to \Sigma^\omega$, and write $w^{(n)} = w_0^{(n)}w_1^{(n)}\dots = f(n)$ for each $n \geq 0$. Consider a word $w = w_0w_1\dots$ where for every k, we have $w_k \neq w_k^{(k)}$. Is there an integer n such that $w = f(n)$?
2. Conclude.
3. Use that result to prove that the set $[0, 1]$ is uncountable.
4. Let A be an infinite countable set. Use similar techniques to prove that 2^A, the set of subsets of A, is uncountable.

3 Counting Turing machines

Let $\Sigma \subseteq \Gamma$ be two finite alphabets. In all this exercise, by *language*, we mean a language of finite words over Σ, and by *Turing machine*, we mean a Turing machine with input alphabet Σ and tape alphabet Γ.

1. Explain briefly why a Turing machine can be encoded as a finite word over a finite alphabet.
2. Which ones of these sets are countable?
   (a) The set of Turing machines;
   (b) the set of decidable languages ($\mathbf{R}$);
   (c) the set of recursively enumerable languages ($\mathbf{RE}$);
   (d) the set of languages (2^{Σ^*});

(e) the set of undecidable languages ($2^{\Sigma^*} \setminus \mathbf{R}$);

(f) the set of languages that are neither recursively enumerable nor co-recursively enumerable ($2^{\Sigma^*} \setminus \mathbf{RE} \setminus \mathbf{coRE}$).

3. Assume one finds a computation model that is strictly more expressive than Turing machines (challenging therefore the Church-Turing thesis). Is there a hope that such a computation model "decides" all the languages?