Complexité avancée - TD 2

Guillaume Scerri

September 30, 2022

Exercise 1 (Dyck's language)**.**

- Let A be the language of balanced parentheses – that is the language generated by the grammar $S \to (S)|SS|\epsilon$. Show that $A \in \mathsf{L}$.
- What about the language B of balanced parentheses of two types? that is the language generated by the grammar $S \to (S)|[S]|SS|\epsilon$

Exercise 2 (A few NL-complete problems)**.** Show that the following problems are NL-complete.

1. Deciding if a non-deterministic automaton $\mathcal{A}$ accepts a word w.
2. Deciding if a directed graph is strongly connected.
3. Deciding if a directed graph has a cycle.

Exercise 3 (Restrictions of the `SAT` problem)**.**

1. Let 3-`SAT` be the restriction of `SAT` to clauses consisting of at most three literals (called 3-clauses). In other words, the input is a finite set S of 3-clauses, and the question is whether S is satisfiable. Show that 3-`SAT` is NP-complete for logspace reductions (assuming `SAT` is).
2. Let 2-`SAT` be the restriction of `SAT` to clauses consisting of at most two literals (called 2-clauses). Show that 2-`SAT` is in P, using proofs by resolution.
3. Show that 2-`UNSAT` (i.e, the unsatisfiability of a set of 2-clauses) is NL-complete.
4. Conclude that 2-`SAT` is NL-complete. You may use the fact that $\mathsf{co}-\mathsf{NL} = \mathsf{NL}$.

Exercise 4 (Space hierarchy theorem)**.** Consider two space-constructible functions f and g such that $f = o(g)$. Prove that $\mathsf{DSPACE}(f) \subsetneq \mathsf{DSPACE}(g)$.

Hint: You may consider a language $L = \{(M, w') \mid$ the simulation (by a universal TM) of M on (M, w') rejects $\}$ with an appropriate restriction on the simulation of M.

Exercise 5 (NL alternative definition)**.** A Turing machine with *certificate tape*, called a verifier, is a <u>deterministic</u> Turing machine with an extra read-only input tape called *the certificate tape*, which moreover is *read once* (*i.e.* the head on that tape can either remain on the same cell or move right, but never move left). A verifier takes as input a word x in the alphabet, along with word u written in the certificate tape.

Define NL_{certif} to be the class of languages L such that there exists a polynomial $p : \mathbb{N} \to \mathbb{N}$ and a verifier M running in logarithmic space such that:

$$x \in L \text{ iff } \exists u, |u| \leq p(|x|) \text{ and } M \text{ accepts on input } (x, u)$$

1. Show that $\mathsf{NL}_{certif} = \mathsf{NL}$
2. What complexity class do you obtain if you remove the read-once constraint in the definition of a machine with certification tape ? Justify your answer. You may use the fact that SAT is NP-complete for logspace reduction.