Introduction to Computation and Complexity

Final Exam

December 2019

Intitulé	EPITA_2020_ING3_S9
Majeure	RDI
Code	COMP
Teacher	Adrien Pommellet
Duration	2 hours
Documents	All non-electronic documents are allowed.

This exam is graded on a scale from 0 to 15 points. Read each one of the four exercises carefully. When asked to prove something, make sure that your answer is detailed and rigorous.

Exercise 1 (3 points)

Consider the language:

$$\mathcal{N} = \{\langle M \rangle \# w \mid M(w) \text{ writes a non-blank symbol on its second tape.}\}$$

where $\langle M \rangle$ stands for the code of the Turing Machine M.

Question 1.

Find a reduction f from the halting problem $\mathcal{H}$ to $\mathcal{N}$.

Hint: given a Turing machine M and an input x, design a Turing machine N and an input y such that $M(x)$ halts if and only if $N(y)$ writes a non-blank symbol on its tape.

Question 2.

Is $\mathcal{N}$ decidable? Why?

Exercise 2 (4 points)

The radix order $<_{rad}$ is a binary relation on the set of input words Σ^* such that $x \leq_{rad} y$ if and only if $|x| < |y|$ or $|x| = |y|$ and $x \leq_{lex} y$, where $\leq_{lex}$ stands for the lexicographic order.

We want to prove that a language L is decidable if and only if there exists a Turing machine enumerating all the words of L in radix order.

Question 1.

Let M be a Turing machine accepting a decidable language L. Design a Turing machine N enumerating L in radix order.

Hint: N can use multiple tapes.

Question 2.

Let M be a Turing machine enumerating a sequence $(w_i)_{i \geq 0}$ sorted according to the radix order. Prove that the language $L = \{w_i \mid i \geq 0\}$ is decidable.

Hint: design a Turing machine N with multiple tapes accepting L. Obviously, N should rely on M.

Exercise 3 (4 points)

Let `DOUBLE-SAT` $= \{\varphi \mid \varphi$ is a Boolean formula satisfiable at least twice.$\}$.

Question 1.

Prove that `DOUBLE-SAT` is in `NP`.

Hint: find a non-deterministic polynomial algorithm, or use the certification theorem.

Question 2.

Find a polynomial reduction f from `SAT` to `DOUBLE-SAT`.

Hint: given a formula φ, design in polynomial time a formula ψ such that φ admits at least one solution if and only ψ admits at least two solutions.

Question 3.

Is `DOUBLE-SAT` `NP`-complete? Why?

Exercise 4 (4 points)

We want to prove that P is closed under Kleene star. To this end, we consider a language $L \subseteq \Sigma^*$ recognized by an algorithm A running in polynomial time.

There is no need to write proofs featuring Turing machines in this exercise.

Question 1.

Let $x = x_1 \dots x_n$ be a non-empty word in Σ^*. $\forall i, j \in \{1, \dots, n\}$, we define:

$$l_{i,j} = \begin{cases} 1 \text{ if } i \leq j \text{ and } x_i \dots x_j \in L \\ 0 \text{ otherwise.} \end{cases}$$

Design an algorithm computing the matrix $(l_{i,j})_{i,j \in \{1,\dots,n\}}$ in polynomial time.

Question 2.

Let G be a directed graph with n vertices $X_1, \dots, X_n$ such that $X_i \to X_j$ if and only if $l_{i,j} = 1$. Prove that $x \in L^*$ if and only if there exists a path from X_1 to X_n in G.

Hint: prove both directions of the equivalence.

Question 3.

Prove that P is closed under Kleene star.

Hint: design an algorithm B recognizing L^* in polynomial time.