MAST31213 Complexity theory
Master's Programme in Mathematics and Statistics
Fall 2019
Exercise set 5

Read chapters 2.1–2.2 of the book.

Exercise 1. A *partial* function from $\{0,1\}^*$ to $\{0,1\}^*$ is a function that is not necessarily defined on all its inputs. We say that a Turing machine M computes a partial function f if for every x on which f is defined, $M(x) = f(x)$ and for every x on which f is not defined M gets into an infinite loop when executed on input x. If S is a set of partial functions, we define f_S to be the Boolean function that on input α outputs 1 iff M_α computes a partial function in S. *Rice's Theorem* says that for every nontrivial S (i.e., neither empty nor the set of all partial functions computable by some Turing machine), the function f_s is not computable.

(a) Show that Rice's Theorem yields an alternative proof that the function $\texttt{HALT}$ is not computable.
(b) Prove Rice's Theorem.

Exercise 2. Prove that allowing the certificate to be of size *at most* $p(|x|)$ rather than equal to $p(|x|)$ in the definition of **NP** makes no difference. That is, show that for every polynomial-time Turing machine M and polynomial $p : \mathbb{N} \to \mathbb{N}$, the language

$$\{x \in \{0,1\}^* : \exists u \text{ s.t. } |u| \leq p(|x|) \text{ and } M(x,y) = 1\}$$

is in **NP**.

Exercise 3. Prove that the following languages are in **NP**:

(a) Three colouring $\texttt{3COL} = \{\llcorner G \lrcorner : \text{ the graph } G \text{ has a colouring with three colours}\}$, where a colouring of G with c colours is an assignment of a number in $\{1,\dots,c\}$ to each vertex such that no adjacent vertices get the same number.
(b) Graph isomorphism $\texttt{GRAPH\_ISOM} = \{(\llcorner G \lrcorner, \llcorner H \lrcorner) : \text{ the graphs } G \text{ and } H \text{ are isomorphic}\}$.

Exercise 4. Suppose $L_1, L_2 \in \mathbf{NP}$. Then is $L_1 \cup L_2$ in **NP**? What about $L_1 \cap L_2$?

Exercise 5. Let $\texttt{HALT}$ be the Halting language (i.e., the language, whose characteristic function the halting function is). Show that $\texttt{HALT}$ is **NP**-hard. Is it **NP**-complete?