PART II AUTOMATA AND FORMAL LANGUAGES
MICHAELMAS 2020-21
EXAMPLE SHEET 2

Unless explicitly asked to, you need not *prove* that any machine you construct defines the language you say it does.
You may appeal to Church's thesis at any time, provided you clearly say so.
$*$ denotes a harder problem.

(1) Let A be a recursive set, and define the set

$$B = \{2n \mid n \in A\} \cup \{2n+1 \mid n \in \mathbb{K}\}.$$

(a) Is B recursive? If not, which of B and $\mathbb{N} \setminus B$ are r.e. (if any), and why?

(b) By replacing A in the construction of B with a suitably chosen set, construct a set $C \subseteq \mathbb{N}$ such that neither C nor $\mathbb{N} \setminus C$ are r.e..

(2) Give examples, with proofs, of infinite collections of recursive sets whose union:

(a) Is recursive.

(b) Is r.e. but not recursive.

(c) Is not r.e., and its complement is not r.e. either.

(3) Show that there is an indexing set $I \subseteq \mathbb{N}$ which is r.e. and such that W_n is recursive for each $n \in I$ but $\bigcap_{n \in I} W_n$ is not r.e..

(4) Prove that the set $\{n \in \mathbb{N} \mid n$ codes a program and the set W_n has more than 5 elements$\}$ is r.e., but not recursive.

(5) We define the following sets:
$\mathbf{Tot} := \{n \in \mathbb{N} \mid n$ codes a program and $f_{n,1}$ is total$\}$, the indices of ***tot***al P.R. functions.
$\mathbf{Inf} := \{n \in \mathbb{N} \mid n$ codes a program and W_n is infinite$\}$, the indices of ***inf***inite r.e. sets.
$\mathbf{Fin} := \{n \in \mathbb{N} \mid n$ codes a program and W_n is finite$\}$, the indices of ***fin***ite r.e. sets.

(a) Show that $\mathbf{Tot} = \{n \in \mathbb{N} \mid n$ codes a program and $W_n = \mathbb{N}\}$.

(b) Show that $\mathbb{K} \leq_m \mathbf{Tot}$, and that $\mathbb{K} \leq_m \mathbf{Inf}$.

(c) Are any of $\mathbf{Tot}, \mathbf{Inf}, \mathbf{Fin}$ recursive? Give reasons.

(d) Show that $\mathbf{Inf} \leq_m \mathbf{Tot}$.
Hint: Try adapting the proof of Theorem 1.27 *from the lecture notes.*

Date: October 26, 2020.

(e*) Show that $\mathbb{K} \leq_m$ **Fin**.
Hint: consider the partial function $h : \mathbb{N} \to \mathbb{N}$ *given by*

$$h(n,x) = \begin{cases} 0 & \text{if } f_{n,1}(n) \text{ does not halt after } x \text{ steps} \\ \uparrow & \text{otherwise.} \end{cases}$$

(f) Using the above results, show that **Tot** is not r.e., and neither is $\mathbb{N} \setminus$ **Tot**.

(6) Given an explicit code m for a register machine P_m, consider the set

$$T_m := \{n \in \mathbb{N} \mid n \text{ codes a program and } W_m \subseteq W_n\}$$

Construct two explicit codes m, m' such that T_m is recursive, and $T_{m'}$ is not r.e.
Hint: Use the results of the previous question.

(7) Let g be a total recursive function on k variables. Show that, with **Tot** as in (10), we have **Tot** $\leq_m \{n \in \mathbb{N} \mid n \text{ codes a program and } f_{n,k} = g \text{ as functions }\}$. Hence show there is no partial algorithm to verify if answers to question (1) of Example Sheet 1 are correct, nor a partial algorithm to verify if they are incorrect. (That is, the set of correct answers is not r.e., and neither is the set of incorrect answers.)

(8) Construct DFAs, via transition diagrams, which accept the following languages:

(a) $\{w \in \{0,1\}^* \mid |w| > 2\}$.

(b) $\{w \in \{0,1\}^* \mid w$ is an alternating sequence of 1's and 0's $\}$.

(c) $\{w \in \{0,1\}^* \mid w$ is a multiple of 3 when interpreted in binary $\}$.

(d) $\{w \in \{a,\ldots,z\}^* \mid w$ contains *ababa* as a substring $\}$.

(9) Construct NFAs, via transition diagrams, which accept the following languages:

(a) $\{w \in \{a,\ldots,z\}^* \mid w$ contains *ababa* as a substring $\}$.

(b) $\{w \in \{a,\ldots,z\}^* \mid w$ contains *dpmms* and/or *damtp* as a substring $\}$.

(c) $\{w \in \{a,\ldots,z\}^* \mid w \in \{what, where, when\}\ \}$.

Use the subset construction to convert (9a) to a DFA.

(10) Construct an ϵ-NFA which accepts the *union* of the three languages from question (9), and has only *one* accept state.

(11) Let Σ be a finite alphabet, and L be a regular language over Σ. Taking the shortlex ordering on Σ^* with word ordering $\{w_1, w_2, \ldots\}$, show that $\{n \in \mathbb{N} \mid w_n \in L\}$ is a recursive set.

(a) Fix an ordering of Σ, and use this to describe an ordering (bijection with $\mathbb{N}$) $\{w_1, w_2, \ldots\}$ of Σ^* by extending the idea of the shortlex ordering of $\mathbb{N}^m$.

(b) Using this ordering of Σ^*, show that $\{n \in \mathbb{N} \mid w_n \in L\}$ is a recursive set.
(So there is no regular language which is r.e. but not recursive: DFAs are weaker than register machines.)