PART II AUTOMATA AND FORMAL LANGUAGES
MICHAELMAS 2015-16
EXAMPLE SHEET 1

$*$ denotes a harder problem.

(1) Give an example of a register machine, either via a program diagram or a sequence of instructions, for computing each of the following functions.

(a) $f(n) = n + 3$

(b) $f(n) = 3n$

(c) $f(m, n) = mn$

(d) $f(m, n) = m \mod n$

(e) $f(m, n) = \begin{cases} 1 \text{ if } m = n \\ 0 \text{ if } m \neq n \end{cases}$

(2) Draw a program diagram for each of the following sequences of instructions, and identify the upper register index of each program. Also, for the specified n, write down the function on n variables that the program computes.

(a) $(1, +, 2), (1, +, 0)$. $n = 1$.

(b) $(1, -, 2, 5), (2, +, 3), (3, +, 4), (4, +, 1), (3, -, 6, 0), (2, -, 7, 8), (1, +, 6), (4, -, 9, 11), (5, +, 10), (2, +, 8), (5, -, 12, 5), (4, +, 11)$. $n = 1$.

(c) $(2, +, 2), (4, +, 3), (3, -, 5, 7), (1, -, 9, 10), (5, +, 6), (8, +, 3), (3, +, 0), (1, +, 8)$. $n = 4$.

(3) Show that the following functions can be built up as total recursive functions from the basic functions via composition, recursion and minimisation.

(a) $f(a, b, c, x) = ax^2 + bx + c$

(b) $f(n) = n!$

(c) $f(m, n) = m \mod n$

(4) Use Church's thesis to show there is an algorithm that, on input of a register machine program P, halts iff P halts on *some* input in *some* number of variables.

(5) Let E be an infinite subset of $\mathbb{N}$. Use Church's thesis to show that E is recursive iff there is a strictly increasing total recursive function $f : \mathbb{N} \to \mathbb{N}$ whose image is precisely E.

(6) Use Church's thesis to show that the set of prime numbers is recursive.

(7) Use Church's thesis to show there is an algorithm that, on input of an integer n, outputs the prime factorisation of n.

(8) Let $f : \mathbb{N} \to \mathbb{N}$ be a total bijective function. Show that f is total recursive iff f^{-1} is total recursive.

(9*) We define *Cantor's pairing function* $\langle \cdot, \cdot \rangle : \mathbb{N}^2 \to \mathbb{N}$ by

$$\langle x, y \rangle := \frac{1}{2}(x+y)(x+y+1) + y$$

and its natural extension, $\langle \cdot, \dots, \cdot \rangle_k : \mathbb{N}^k \to \mathbb{N}$, inductively via

$$\langle x_1, \dots, x_k \rangle_k := \big\langle \langle x_1, \dots, x_{k-1} \rangle_{k-1}, x_k \big\rangle$$

(a) Show that $\langle \cdot, \cdot \rangle$ is a total bijection from $\mathbb{N}^2 \to \mathbb{N}$.

(b) Show that $\langle \cdot, \cdot \rangle$ is a total computable function.

(c) Show that $\langle \cdot, \dots, \cdot \rangle_k : \mathbb{N}^k \to \mathbb{N}$ is a total computable bijection.

(d) Give an explicit total computable bijection $g : \mathbb{N} \to \bigcup_{i=1}^{\infty} \mathbb{N}^i$, in terms of $\langle \cdot, \dots, \cdot \rangle_k^{-1}$.

(e) Show that there is a total computable function $h : \mathbb{N}^2 \to \mathbb{N}$ such that, for each m, k, the partial computable function $f_{m,k}$ satisfies:

$$f_{m,k}(x_1, \dots, x_k) = f_{h(m,k),1}(\langle x_1, \dots, x_k \rangle_k)$$

(f) Show that with $\langle \cdot, \dots, \cdot \rangle_k : \mathbb{N}^k \to \mathbb{N}$ we can produce all multi-variable partial computable functions from just the one-variable ones.