MSc in Mathematics and Foundations of Computer Science

LAMBDA CALCULUS AND TYPES

Hilary Term 2025

Submission deadline 12 noon, Wednesday 16th April 2025, via Inspera.

There is a total of 100 marks available for this paper, you should attempt all parts of the paper.

NB: You must not discuss this examination paper with anyone.

Question 1

We have seen that leftmost reductions are β-normalising, i.e. if a term t is β-normalisable and we keep applying the leftmost reduction, we will eventually reach its unique β-normal form t'. In other words, if $t \twoheadrightarrow_\beta t'$ and t' is a normal form, then $t \twoheadrightarrow_l t'$.

Let

$$t = t_0 \to_\beta t_1 \to_\beta \ldots \to_\beta t_n = t'$$

be a sequence of β-reductions from t to its normal form t'. The length of this sequence is the number of reductions in it, i.e. n. We say a sequence is optimal if it has the shortest possible length.

(a) For every natural number k, find a β-normalisable term t such that the leftmost reduction sequence starting at t is exactly k steps longer than the optimal reduction sequence.

(5 marks)

(b) For every natural number k, find a β-normalisable term t such that the length of the leftmost reduction sequence starting at t is at least k times the length of the optimal reduction sequence. (5 marks)

(c) Let $|t|$ be the number of nodes in the construction tree of t. Find a family T of β-normalisable terms such that:

- For every $n \in \mathbb{N}$, there is a term $t \in T$ with $|t| \geq n$;
- There is a constant $c \in \mathbb{N}$, such that for every term $t \in T$, an optimal reduction sequence from t has a length of at most $c \cdot |t|$;
- There is a constant $d \in \mathbb{N}$, such that for every term $t \in T$, the leftmost reduction sequence from t has a length of at least $d \cdot |t|^2$.

(5 marks)

Question 2

Recall that for any natural number $n \in \mathbb{N}$, we denote its Church encoding by $\ulcorner n \urcorner$ and that a function $\phi : \mathbb{N}^m \to \mathbb{N}$ is called *definable* by a term t if for all $n_1, \ldots, n_m \in \mathbb{N}$, we have

$$\lambda\beta \vdash t \ulcorner n_1 \urcorner \ \ldots \ \ulcorner n_m \urcorner \quad = \quad \ulcorner \phi(n_1, \ldots, n_m) \urcorner.$$

In each part (a)–(f) below, prove that the provided function is definable and find a term that defines it.

(a) $\phi_1 : \mathbb{N}^2 \to \mathbb{N} \quad \phi_1(n_1, n_2) = \gcd(n_1, n_2)$, where gcd denotes the greatest common divisor (5 marks)

(b) $\phi_2 : \mathbb{N}^5 \to \mathbb{N} \quad \phi_2(n_1, \ldots, n_5) = \gcd(n_1, \ldots, n_5)$ (5 marks)

(c) $\phi_3 : \mathbb{N} \to \mathbb{N} \quad \phi_3(n) = \begin{cases} 1 & n \text{ is prime} \\ 0 & \text{otherwise} \end{cases}$ (5 marks)

(d) $\phi_4 : \mathbb{N}^2 \to \mathbb{N} \quad \phi_4(n_1, n_2) =$ number of common prime factors of n_1 and n_2
For example, $\phi_4(12, 9) = 1$ since 3 is the only common prime factor of 12 and 9. (5 marks)

The Collatz conjecture is one of the most classical open problems in number theory. Consider the following operation on a natural number n:

$$f(n) = \begin{cases} n/2 & n \text{ is even} \\ 3 \cdot n + 1 & n \text{ is odd} \end{cases}$$

We can form a sequence of numbers by repeated application of f to n, i.e. the *Collatz sequence* starting at n is

$$n, f(n), f(f(n)), f(f(f(n))), \ldots$$

Here are two example Collatz sequences:

$$1, 4, 2, 1, 4, 2, \ldots$$

$$52, 26, 13, 40, 20, 10, 5, 16, 8, 4, 2, 1, 4, 2, 1, \ldots$$

Collatz conjectured that for every starting number n, the sequence will eventually reach 1 and thus keep repeating 1, 4, 2. It is known that the conjecture holds when the starting number n is less than 10^{20}. We define the *length* of a Collatz sequence as the number of entries before reaching the first 1.

(e) $\phi_5 : \mathbb{N} \to \mathbb{N} \quad \phi_5(n) = \begin{cases} \text{Length of the Collatz sequence starting at } n & n < 10^{20} \\ 0 & n \geq 10^{20} \end{cases}$
For example, $\phi_5(1) = 0, \phi_5(2) = 1, \phi_5(4) = 2, \phi_5(52) = 11$. (5 marks)

(f) $\phi_6 : \mathbb{N}^2 \to \mathbb{N} \quad \phi_6(n_1, n_2) = |\{n \in \mathbb{N} \mid 0 < n \leq n_1 \ \wedge \ \phi_5(n) = n_2\}|$. (5 marks)

TURN OVER

Question 3

A *Deterministic Finite Automaton* over the alphabet $\{0,1\}$ is a tuple $A = (Q, \delta, q_0, F)$ where:

- Q is a finite set of *states*;
- $q_0 \in Q$ is the *initial state*;
- $F \subseteq Q$ is the set of *accepting states*; and
- $\delta : Q \times \{0,1\} \to Q$ is the *transition function.*

Let $s = [s_1, \ldots, s_n]$ be a string, i.e. a sequence over the alphabet $\{0,1\}$. The *run* of A over s is a sequence of states $q_0, q_1, \ldots, q_n$, where q_0 is the initial state and for every $i \geq 1$, we have $q_i = \delta(q_{i-1}, s_i)$. Intuitively, we start at q_0 and, in each step, if we are at q_{i-1}, we read one character s_i from the input string and move according to the transition function δ to the state $q_i = \delta(q_{i-1}, s_i)$. Specifically, if our string s is empty, then the corresponding run is simply q_0. The automaton A is said to accept s if $q_n \in F$.

(a) Suppose $Q \subseteq \mathbb{N}$, i.e. every state is identified by a natural number. Devise an encoding for strings and DFAs in λ-calculus. Note that although this part has few marks, a clever design would significantly facilitate the next parts. So, it is expected that you come back and redesign your encoding based on the challenges faced in the remainder of this question. (5 marks)

(b) Find a term t such that

$$\lambda\beta \vdash t \ \ulcorner A \urcorner \ \ulcorner s \urcorner \ = \ \mathbf{true} \quad \text{if } A \text{ accepts } s;$$

$$\lambda\beta \vdash t \ \ulcorner A \urcorner \ \ulcorner s \urcorner \ = \ \mathbf{false} \quad \text{if } A \text{ does not accept } s.$$

Here, $\ulcorner A \urcorner$ refers to your encoding of the DFA A. Similarly, $\ulcorner s \urcorner$ is your encoding of s as in part (a) above. Following the lectures, we have $\mathbf{true} \equiv \lambda xy.x$ and $\mathbf{false} \equiv \lambda xy.y$.

(20 marks)

(c) Find a term t_{empty} such that

$$\lambda\beta \vdash t_{\text{empty}} \ \ulcorner A \urcorner \ = \ \mathbf{true} \quad \text{if } A \text{ does not accept any string;}$$

$$\lambda\beta \vdash t_{\text{empty}} \ \ulcorner A \urcorner \ = \ \mathbf{false} \quad \text{if there is at least one string accepted by } A.$$

(10 marks)

(d) Find a term t_{prefix} such that

$$\lambda\beta \vdash t_{\text{prefix}} \ \ulcorner A \urcorner \ \ulcorner s \urcorner \ = \ \mathbf{true} \quad \text{if there is a string } s' \text{ for which } A \text{ accepts } ss';$$

$$\lambda\beta \vdash t_{\text{prefix}} \ \ulcorner A \urcorner \ \ulcorner s \urcorner = \mathbf{false} \quad \text{if there is no string } s' \text{ for which } A \text{ accepts } ss'.$$

Here, ss' is the concatenation of s and s'. (10 marks)

(e) Find two terms p and q such that for all DFAs A and strings s, we have the equalities

$$\mathbf{not}\ (t_{\text{empty}} \ulcorner A \urcorner) \ = \ t_{\text{prefix}} \ulcorner A \urcorner \ p$$

$$\mathbf{not}\ (t_{\text{prefix}} \ulcorner A \urcorner \ulcorner s \urcorner) \ = \ t_{\text{empty}}\ (q \ulcorner A \urcorner \ulcorner s \urcorner)$$

in $\lambda\beta$. (10 marks)