Computability Final Test — 2013-06-10

Notes.

- Answer both theory questions, and choose and solve two exercises, only. Solving more exercises results in the failure of the test.
- To pass the exam you need to provide a reasonable contribution in *both* Theory and Exercises parts.
- Exercises with higher number award more points. To achieve a score ≥ 28 you have to solve the exercise marked with $\star$ below.
- Significantly wrong answers will result in negative scores.
- Always provide a justification for your answers.

Reminder: when equating the results of partial functions (as in $\phi_i(3) = \phi_i(5)$), we mean that either 1) both sides of the equation are defined, and evaluate to the same natural number, or 2) both sides are undefined.

Theory

Question 1. *Prove that* $\mathsf{K} \notin \mathcal{R}$ *and that* $\bar{\mathsf{K}} \notin \mathcal{RE}$*.*

Question 2. *Prove the Rice theorem (either its specialized version for the* λ*-calculus or its general form in the theory of recursive functions).*

Exercises

Exercise 3. *Let* $g, h \in \mathcal{R}$*,* $A \in \mathcal{R}$*,* $B \in \mathcal{RE}$*. Prove that the following* $f \in \mathcal{R}$*.*

$$f(x) = \begin{cases} g(x) & \textit{if } x \in A \\ h(x) & \textit{if } x \notin A \wedge x \in B \\ undefined & \textit{otherwise} \end{cases}$$

Solution (sketch). We can rewrite f in an equivalent way using an auxiliary function l as follows:

$$f(x) = \begin{cases} g(x) & \text{if } x \in A \\ l(x) & \text{otherwise} \end{cases} \qquad \text{where} \qquad l(x) = \begin{cases} h(x) & \text{if } x \in B \\ undefined & \text{otherwise} \end{cases}$$

The above definition is indeed equivalent because ... (check all three cases).

By the if-then-else lemma on $\mathcal{RE}$ sets, since $B \in \mathcal{RE}$, $h \in \mathcal{R}$, and l is undefined outside B, we have $l \in \mathcal{R}$. Hence, by the if-then-else lemma on $\mathcal{R}$ sets, since $A \in \mathcal{R}$, $g \in \mathcal{R}$, and $l \in \mathcal{R}$, we conclude $f \in \mathcal{R}$. □

Exercise 4. *Prove whether* $\{i \mid \phi_i(1) = 2\} \leq_m \{i \mid \phi_i(3) = undefined\}$

Solution (sketch). The statement is false.
Indeed, if it were true we would also have

$$A = \{i \mid \phi_i(1) \neq 2\} \leq_m \{i \mid \phi_i(3)\ defined\} = B$$

Note that $B \in \mathcal{RE}$ (it is easy to define a semi-verifier S_B). This implies $A \in \mathcal{RE}$, but one can also prove $A \notin \mathcal{RE}$ by Rice-Shapiro ($\Leftarrow$). This is a contradiction. □

Exercise 5. *Let f be an increasing recursive total function (that is, such that $\forall n, m \in \mathbb{N}.\ n < m \implies f(n) < f(m)$). Prove that $\mathsf{ran}(f) \in \mathcal{R}$.*

Solution (sketch). The following is a verifier for $\mathsf{ran}(f)$:

```
procedure V(x) :
    i := 0;
    while f(i) < x do
        i := i + 1;
    if f(i) = x then
        return 1;
    else
        return 0;
```

The algorithm is well-defined since $f \in \mathcal{R}$. Moreover, the whole algorithm halts for any x since the sequence $f(0), f(1), f(2), \ldots$ is strictly increasing, hence in at most x steps the `while` guard $f(i) < x$ will become false. Also, f being total ensures that the evaluation of $f(i)$ always halts.

Further, the algorithm is obviously correct.

- If $x \in \mathsf{ran}(f)$, there is an i such that $x = f(i)$. Hence, consider the minimum such i (satisfying $x = f(i)$). Then, in exactly i steps the while loop will find this i: the loop will be exited and 1 is returned.
- If $x \notin \mathsf{ran}(f)$, there is no i such that $x = f(i)$. When the while loop is exited, the `if` guard $f(i) = x$ will be false, hence 0 is returned.

□

Exercise 6. ⋆ *Let*

$$f(n) = \max(\{0\} \cup \{\phi_m(m) \mid m \in \mathsf{K}\ \wedge\ m \leq n\})$$

Prove by diagonalisation that $f \notin \mathcal{R}$.

Solution (sketch). Intentionally omitted.

□