Computability Exam Solutions

August 30, 2010

Exercise 1

Rice's Theorem

Statement: Let $A \subseteq \mathbb{N}$ be saturated with $A \neq \emptyset$ and $A \neq \mathbb{N}$. Then A is not recursive.

Definition: A set $A \subseteq \mathbb{N}$ is saturated (or extensional) if:

```
∀x,y ∈ ℕ: (x ∈ A ∧ ϕₓ = ϕᵧ) ⟹ y ∈ A
```

Proof:

We prove that $K \leq_m A$, which implies A is not recursive since K is not recursive.

Since $A \neq \emptyset$ and $A \neq \mathbb{N}$, there exist indices $e_0 \notin A$ and $e_1 \in A$.

Define $g : \mathbb{N}^2 \to \mathbb{N}$ by:

```
g(x,y) = {
  ϕₑ₁(y)  if ϕₓ(x) ↓
  ϕₑ₀(y)  if ϕₓ(x) ↑
}
```

This can be implemented as:

```
g(x,y) = {
  ϕₑ₁(y)  if ∃t. H(x,x,t)
  ϕₑ₀(y)  otherwise
}
```

Since φ_{e1} and φ_{e0} are computable, and H is decidable, g is computable.

By the s-m-n theorem, there exists a total computable function $s : \mathbb{N} \to \mathbb{N}$ such that:

```
ϕₛ₍ₓ₎(y) = g(x,y)
```

Verification of the reduction:

- **If $x \in K$:** Then $\varphi_x(x) \downarrow$, so $\forall y$: $\varphi_{s(x)}(y) = \varphi_{e1}(y)$, hence $\varphi_{s(x)} = \varphi_{e1}$. Since A is saturated and $e_1 \in A$, we get $s(x) \in A$.
- **If $x \notin K$:** Then $\varphi_x(x) \uparrow$, so $\forall y$: $\varphi_{s(x)}(y) = \varphi_{e0}(y)$, hence $\varphi_{s(x)} = \varphi_{e0}$. Since A is saturated and $e_0 \notin A$, we get $s(x) \notin A$.

Therefore, $x \in K \iff s(x) \in A$, which means $K \leq_m A$ via s.

Since K is not recursive, A cannot be recursive.

Exercise 2

Analysis of $f(x) = \varphi_x(x+1) + 1$ if $\varphi_x(x+1) \downarrow$, $\uparrow$ otherwise

Answer: The function f is computable.

Proof:

The function f can be computed as follows:

```
f(x) = ϕₓ(x+1) + 1
```

with the understanding that if $\varphi_x(x+1) \uparrow$, then $f(x) \uparrow$.

Algorithm to compute f(x):

1. Simulate the computation of $\varphi_x(x+1)$
2. If $\varphi_x(x+1)$ converges to value v, return v + 1
3. If $\varphi_x(x+1)$ diverges, then f(x) diverges

Formal implementation:

```
f(x) = Ψᵤ(x, x+1) + 1
```

where Ψ_u is the universal function.

Since the universal function Ψ_u is computable and addition is computable, f is computable.

Key insight: The function f is simply the composition of:

- The universal function $(x,y) \mapsto \varphi_x(y)$ applied to (x, x+1)
- The successor function

Both operations preserve computability, so f is computable.

Exercise 3

Classification of $A = \{x \mid \varphi_x \text{ strictly increasing}\}$

A function f is strictly increasing if $\forall y,z \in dom(f)$: $y < z \implies f(y) < f(z)$.

A is saturated: $A = \{x \mid \varphi_x \in \mathcal{A}\}$ where $\mathcal{A} = \{f \mid f \text{ is strictly increasing}\}$.

A is not r.e.: We use Rice-Shapiro theorem. Consider the identity function id, which is strictly increasing, so $id \in \mathcal{A}$.

Consider any finite function $\theta \subseteq id$. While θ might be strictly increasing on its finite domain, the key insight is that we can find finite functions that are not strictly increasing.

Actually, let me be more careful. Consider $f(x) = x^2$ (strictly increasing). Then $f \in A$.

Consider the finite function $\theta = \{(0,0), (1,0)\} \subseteq f$. This θ is not strictly increasing since $\theta(0) = \theta(1) = 0$, violating the strict inequality requirement.

Since $f \in A$ and $\exists$ finite $\theta \subseteq f$ with $\theta \notin A$, by Rice-Shapiro theorem, A is not r.e.

Ā is not r.e.: Consider the constant function $g(x) = 0$. This function is not strictly increasing since it's constant, so $g \notin A$.

For any finite $\theta \subseteq g$, we have $\theta : dom(\theta) \to \{0\}$. Such a function cannot be strictly increasing unless $|dom(\theta)| \leq 1$.

The empty function $\varnothing$ is vacuously strictly increasing, so $\varnothing \in A$.
Any function $\theta = \{(k,0)\}$ for $k \in \mathbb{N}$ is also strictly increasing on its single-element domain.

Since $g \notin A$ and $\exists$ finite $\theta \subseteq g$ with $\theta \in A$, by Rice-Shapiro theorem, Ā is not r.e.

Final classification: A and Ā are both not r.e. (and hence not recursive).

Exercise 4

Classification of $B = \{x \mid \forall m \in \mathbb{N}.\ m \cdot x \in W_x\}$

B is not r.e.: The condition requires that ALL multiples of x are in W_x. This universal quantification over an infinite set typically leads to non-r.e. sets.

We can show $\bar{K} \leq_m B$. Define $g : \mathbb{N}^2 \to \mathbb{N}$ by:

```
g(x,y) = {
  y       if x ∉ K and y = k·x for some k ∈ ℕ
  ↑       otherwise
}
```

By s-m-n theorem, $\exists s$ such that $\varphi_{s(x)}(y) = g(x,y)$.

- If $x \notin K$: $\varphi_x(x) \uparrow$, so $W_{s(x)} = \{k \cdot x : k \in \mathbb{N}\}$, hence $\forall m$: $m \cdot x \in W_{s(x)}$, so $s(x) \in B$
- If $x \in K$: $\varphi_x(x) \downarrow$, so $W_{s(x)} = \varnothing$, hence $\exists m$: $m \cdot x \notin W_{s(x)}$ (any m), so $s(x) \notin B$

Wait, I need to be more careful about s(x) vs x in the multiples.

Let me redefine: if $x = 0$, then $m \cdot x = 0$ for all m, which is manageable. For $x > 0$:

```
g(x,y) = {
  y      if x ∉ K and ∃k ∈ ℕ: y = k·s(x)
  ↑      otherwise
}
```

This becomes circular. Let me use a different approach.

Alternative reduction: Show that the totality problem reduces to B.

B̄ is r.e.:

```
x ∈ B̄ ⟺ ∃m ∈ ℕ. m·x ∉ Wₓ
```

This can be semi-decided by searching for some multiple of x that never appears in W_x, though proving non-membership is difficult.

Final classification: B is not r.e.; B̄ might be r.e. depending on detailed analysis.

Exercise 5

Second Recursion Theorem and proof that ∃x such that $\varphi_x(y) = x \dot{-} y$

Second Recursion Theorem: For every total computable function $f : \mathbb{N} \to \mathbb{N}$, there exists $e_0 \in \mathbb{N}$ such that $\varphi_{e0} = \varphi f(e_0)$.

Proof of existence:

Define $g : \mathbb{N}^2 \to \mathbb{N}$ by:

```
g(x,y) = x ∸ y
```

This function is computable since proper subtraction is primitive recursive.

By the s-m-n theorem, there exists a total computable function $s : \mathbb{N} \to \mathbb{N}$ such that:

```
ϕₛ₍ₓ₎(y) = g(x,y) = x ∸ y
```

Define f(x) = s(x). Then f is total and computable.

By the Second Recursion Theorem, there exists e such that:

```
ϕₑ = ϕf(e) = ϕₛ₍ₑ₎
```

For this e, we have:

```
ϕₑ(y) = ϕₛ₍ₑ₎(y) = g(e,y) = e ∸ y
```

Therefore, $x = e$ is the desired index such that $\varphi_x(y) = x \dot{-} y$.