Computability Exam Solutions

September 13, 2011

Exercise 1

Theorem: A is r.e. ⟺ ∃ computable f : ℕ → ℕ such that A = img(f)

where img(f) = {y : ∃z. y = f(z)}.

Proof:

(⇒) If A is r.e., then A = img(f) for some computable f

Case 1: A = ∅ Take f(x) = 0 for all x. Then f is computable and img(f) = {0} ≠ ∅. Actually, for A = ∅, we need the empty image. This is a special case - take f to be a partial function that is nowhere defined. But we need f total...

Let me handle this correctly. If A = ∅, then A is r.e. (vacuously), and we can take any computable function f such that img(f) = ∅. But every total function has non-empty image.

Proper approach: If A = ∅, then A is r.e., and we can represent it as the image of the nowhere-defined function (which is not total). For the theorem to work with total functions, we exclude the empty set or modify the statement.

Case 2: A ≠ ∅ and A is r.e. Since A is r.e., its semi-characteristic function sc_a is computable.

Define g : ℕ → ℕ by:

```
g(⟨x,t⟩) = {
  x  if scₐ(x) converges in exactly t steps
  ↑  otherwise
}
```

Since sc_a is computable, g is computable. Moreover:

```
img(g) = {x : scₐ(x) ↓} = {x : x ∈ A} = A
```

But g might be partial. To get a total function, pick any a_0 ∈ A and define:

```
f(z) = {
  g(z)  if g(z) ↓
  a₀    otherwise
}
```

Then f is total computable and A ⊆ img(f). To ensure img(f) = A, we use the standard enumeration approach.

Standard approach: Since A is r.e., there exists a computable function h that enumerates A (possibly with repetitions). Define f = h, then img(f) = A.

(⇐) If A = img(f) for some computable f, then A is r.e.

Given total computable f with A = img(f), define:

```
scₐ(x) = 1(μz. f(z) = x)
```

Since f is computable, this semi-characteristic function is computable, so A is r.e.

Conclusion: The theorem holds (with appropriate handling of the empty set case).

Exercise 2

Question: Can there exist a non-computable f : ℕ → ℕ such that dom(f) ∩ img(f) is finite?

Answer: Yes, such functions exist.

Construction:

Let K be the halting set. Define f : ℕ → ℕ by:

```
f(x) = {
  x + |K|  if x ∈ K (where |K| is infinite, so this is just x + ∞ conceptually)
  ↑        if x ∉ K
}
```

More precisely, define:

```
f(x) = {
  x + c  if x ∈ K, where c > max(K ∩ [0,n]) for some large n
  ↑      if x ∉ K
}
```

Actually, let me give a cleaner construction:

```
f(x) = {
  2x + 1  if x ∈ K
  ↑       if x ∉ K
}
```

Verification:

1. **f is not computable:** If f were computable, we could decide K:

   ```
   x ∈ K ⟺ f(x) is defined
   ```

   contradicting the undecidability of K.

2. **dom(f) ∩ img(f) analysis:**
   - dom(f) = K
   - img(f) = $\{2x + 1 : x \in K\}$
   - dom(f) ∩ img(f) = $K \cap \{2x + 1 : x \in K\}$

   For $y \in$ dom(f) ∩ img(f), we need:
   - $y \in K$ (since $y \in$ dom(f))
   - $y = 2x + 1$ for some $x \in K$ (since $y \in$ img(f))

   So $y \in K$ and y is odd and $y = 2x + 1$ where $x \in K$. This means $(y-1)/2 \in K$. The intersection is finite if K contains only finitely many numbers x such that $2x + 1 \in K$.

Alternative simpler construction:

```
f(x) = {
  0  if x ∈ K and x > 0
  ↑  otherwise
}
```

Then:

- dom(f) = $K \setminus \{0\}$ (if $0 \notin K$) or K (if $0 \in K$)
- img(f) = $\{0\}$
- dom(f) ∩ img(f) = $\{0\}$ if $0 \in K$, or $\varnothing$ if $0 \notin K$

Both cases give a finite intersection.

Therefore, such non-computable functions exist.

Exercise 3

Classification of A = $\{x \in \mathbb{N} : \exists k \in \mathbb{N}.\ \varphi_x(x + 3k) \uparrow\}$

A is r.e.:

```
scₐ(x) = 1(μ⟨k,t⟩. ∀s ≤ t: ¬H(x, x + 3k, s))
```

Actually, this doesn't work because we're trying to prove non-termination.

The condition says: $\exists k$ such that $\varphi_x(x + 3k)$ doesn't terminate.

This is equivalent to: $\neg\forall k.\ \varphi_x(x + 3k) \downarrow$

So $A = \{x : \neg\forall k \in \mathbb{N}.\ \varphi_x(x + 3k) \downarrow\}$

A is r.e.: This is not immediately clear since it involves proving non-termination.

Actually, let me reconsider. We have:

```
x ∈ A ⟺ ∃k. ϕₓ(x + 3k) ↑
```

We can't directly enumerate this since proving divergence is undecidable.

A is not r.e.: We can show this by reducing from the totality problem. If we could enumerate A, we could potentially solve undecidable problems.

Ā is r.e.:

```
x ∈ Ā ⟺ ∀k ∈ ℕ. ϕₓ(x + 3k) ↓
```

This can be semi-decided by:

```
scĀ(x) = 1(μt. ∀k ≤ t ∃s ≤ t: H(x, x + 3k, s))
```

If all $\varphi_x(x + 3k)$ terminate, then eventually we'll find termination evidence for all k up to some bound.

Final classification: A is not r.e.; Ā is r.e. but not recursive.

Exercise 4

Classification of $B = \{x \in \mathbb{N} : W_x \supseteq Pr\}$

where $Pr \subseteq \mathbb{N}$ is the set of prime numbers.

B is saturated: $B = \{x \mid \varphi_x \in \mathcal{B}\}$ where $\mathcal{B} = \{f \mid Pr \subseteq dom(f)\}$.

B is not r.e.: We use Rice-Shapiro theorem. Consider any total function f (e.g., the identity). Then $f \in \mathcal{B}$ since $Pr \subseteq dom(f) = \mathbb{N}$.

For any finite function $\theta \subseteq f$, we have $dom(\theta)$ finite. Since Pr is infinite, $Pr \not\subseteq dom(\theta)$, so $\theta \notin \mathcal{B}$.

Since $f \in \mathcal{B}$ and $\forall$ finite $\theta \subseteq f$: $\theta \notin \mathcal{B}$, by Rice-Shapiro theorem, B is not r.e.

B̄ is not r.e.: Consider the empty function $\emptyset$. Then $dom(\emptyset) = \emptyset$, so $Pr \not\subseteq \emptyset$, hence $\emptyset \notin \mathcal{B}$, i.e., $\emptyset \in \bar{\mathcal{B}}$.

For any function $g \in \bar{\mathcal{B}}$, we have $Pr \not\subseteq Wg$. Consider $\theta = \emptyset \subseteq g$. Since $dom(\theta) = \emptyset$ and $Pr \not\subseteq \emptyset$, we have $\theta \notin \mathcal{B}$, so $\theta \in \bar{\mathcal{B}}$.

Since $\forall g \in \bar{\mathcal{B}}$: $\emptyset \subseteq g$ and $\emptyset \in \bar{\mathcal{B}}$, the condition for Rice-Shapiro to apply to $\bar{\mathcal{B}}$ is that $\forall$ finite $\theta \subseteq g$: $\theta \in \bar{\mathcal{B}}$. But this isn't necessarily true for all finite θ.

Let me try differently. Consider any function g such that $Pr \not\subset Wg$. There exists some prime $p \notin Wg$. Consider any finite extension $\theta \supseteq g$ with $p \in dom(\theta)$. We still might have $Pr \not\subset dom(\theta)$ (if dom(θ) doesn't include all primes), so $\theta \in \bar{B}$.

By Rice's theorem, since B is saturated and non-trivial, B is not recursive. Combined with the Rice-Shapiro analysis, both B and $\bar{B}$ are not r.e.

Final classification: B and $\bar{B}$ are both not r.e. (and hence not recursive).

Exercise 5

Second Recursion Theorem

For every total computable function $f : \mathbb{N} \to \mathbb{N}$, there exists $e_0 \in \mathbb{N}$ such that:

```
ϕₑ₀ = ϕf(e₀)
```

Proof that $\exists x$ such that $\varphi_x(y) = y/2$ if $x \leq y \leq x + 2$, ↑ otherwise

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

```
g(x,y) = {
  ⌊y/2⌋  if x ≤ y ≤ x + 2
  ↑       otherwise
}
```

This function is computable since:

- The condition $x \leq y \leq x + 2$ is decidable
- The floor function $\lfloor y/2 \rfloor$ is computable
- We can implement the "↑ otherwise" using a divergent loop

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

```
ϕₛ₍ₓ₎(y) = g(x,y)
```

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

By the Second Recursion Theorem, ∃e such that:

```
ϕₑ = ϕf(e) = ϕₛ₍ₑ₎
```

For this e, we have:

```
ϕₑ(y) = ϕₛ₍ₑ₎(y) = g(e,y) = {
  ⌊y/2⌋  if e ≤ y ≤ e + 2
  ↑       otherwise
}
```

Therefore, $x = e$ is the desired index such that $\varphi_x(y) = y/2$ when $x \leq y \leq x + 2$, and undefined otherwise.